

## СЕКЦИЯ 1

### ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ

Среда, 27 апреля 2011 г., читальный зал преподавателей (ауд.232), гл. корпус МГТУ им.Н.Э.Баумана.

**Начало в 10.00.**

**Председатель: профессор, д.т.н. Норенков И.П.**

**Руководитель экспертной комиссии: к.т.н., доцент Федорук В.Г.**

**Ученый секретарь: к.т.н., доцент Власов А.И.**

#### **Экспертная комиссия:**

**БОЖКО А.Н., ВОЛОСАТОВА Т. М., ЖУК Д. М., МАНИЧЕВ В. Б., МАРТЫНЮК В. А., ТРУДОНОШИН В. А., ГРОШЕВ С. В., КНЯЗЕВА С. РОДИОНОВ С. В., ФЕДУРУК Е.**

В рамках работы секции будет осуществлен первый этап конкурсного отбора проектов для участия в конкурсе инновационных проектов МГТУ им.Н.Э.Баумана. Критериями конкурсного отбора участников молодежного научно-инновационного конкурса являются:

1. Уровень инновационности идеи (предложения, метода, способа ...).

1.1. Идея должна быть новой, впервые сформулированной именно самим номинантом. В этом смысле все номинанты равны. Остальные просто не могут быть номинированы на участие в этой программе.

1.2. Уровень наукоемкости тем выше, чем более:

- основательны научные исследования, в результате которых она появилась;

- основательны дальнейшие научные исследования, необходимые для ее реализации.

1.3. Техническая значимость тем выше, чем большее влияние ее реализация окажет на уровень техники. «Пионерные» идеи (изобретения) – пенициллин, лазер, синтез алмазов - открывают новые отрасли науки и техники. Идеальная по инновационности идея неожиданна для рынка. Поэтому она им не может быть сейчас востребована, она сама формирует новую потребность и нишу рынка.

Высокий технический уровень имеют решения, например, многоотраслевого использования.

Оригинальные технические решения дают новые принципы решения известной задачи.

Есть решения, позволяющие решить проблему еще одним, дополнительно к известным, способом. Но и в этом случае, если уже известно 10 способов решения задачи, то новый способ может и не давать существенных преимуществ в решении задачи или давать их только в очень ограниченном по масштабам применения числе случаев.

1.4. Масштабность использования предложения тоже может сильно различаться – от решения локальной задачи одного местного потребителя до ...

1.5. Срок превращения идеи в конечный продукт с выходом его на рынок: новизна, рискованность идеи, объем необходимых научных исследований не позволяют уложиться в 2-3 года, но и не требуют 10-15 лет

1.6. Идея тем актуальней, чем меньше вероятность того, что за 5-7 лет в результате научных исследований появятся и «раскрутятся» до продукта другие более эффективные пути решения задачи.

Победитель сам организует работу по привлечению необходимого финансирования. Инновационная программа МГТУ им.Н.Э.Баумана оказывает ему поддержку на начальном этапе работы над его идеей.

# КОРРЕКТНОСТЬ И ТОЧНОСТЬ РЕШЕНИЙ ЖЕСТКИХ СИСТЕМ ОДУ В СИСТЕМЕ MATHCAD 14

Сахаров М.К.

*Научный руководитель к.т.н. доцент Маничев В.Б.*  
Кафедра РК6, МГТУ им. Н.Э. Баумана, Москва, Россия

## PRECISION AND ACCURACY OF STIFF SYSTEMS OF ODE IN MATHCAD 14

Sakharov M.K.

*The scientific chief: Ph.D. assistant professor Manichev V.B.*  
Department RK6, BMSTU, Moscow, Russia

### Аннотация

В докладе рассмотрена проблема получения в САЕ-подсистемах САПР/PLM качественно верного решения с гарантированной точностью. Рассмотрен набор тестовых задач состоящий из жестких систем дифференциальных уравнений с известным аналитическим решением. Произведено тестирование программного пакета MathCAD 14. Основным критерием тестирования являлось отличие численного решения от аналитического. Приведены наиболее интересные результаты тестирования.

### Abstract

This article deals with the task of obtaining qualitatively and quantitatively the correct solution in CAE-subsystems of CAD/CAE/PDM/PLM. The set of stiff systems of ODE selected for testing was considered. Every system from this set has an analytical solution. PTC MathCAD 14 was tested with these stiff systems. The main criterion was a value of the difference between numeric and analytical solutions. The most interesting results were considered in details in this article.

САЕ технологии и подсистемы в наши дни становятся все более востребованными при разработке технических изделий и объектов. Очевидно, что математическое моделирование процессов и их анализ (Simulation & Analysis – S&A) постепенно становятся неотъемлемой частью цикла проектирования не только в наукоемких отраслях, но и в отраслях занятых выпуском продукции массового спроса. Поэтому сегодня среднестатистическому инженеру бывает недостаточно встроенных в САПР/PLM-систему средств анализа. САЕ-подсистемы, в большинстве случаев, основаны на численном решении различных математических уравнений. При компьютерном моделировании и анализе разнообразных технических систем и объектов на основе дифференциальных уравнений, предъявляются жесткие требования к эффективности применяемых численных методов.

**Постановка задачи.** Эффективность численных методов решения включает в себя два аспекта:

1. Точность получаемых решений данных уравнений.
2. Затраты машинного времени на получение решений.

Фантастический рост производительности вычислительной техники в последнее время выдвигает на первый план проблему точности получаемых решений. Способность с определенной точностью решить задачу до конца, одна из важнейших характеристик программы. Не имеет смысла рассматривать эффективность программы игнорируя степень достоверности выдаваемого ей приближенного решения. В САЕ-подсистемах должна быть обеспечена 100% гарантия корректности и точности результатов, выдаваемых инженерам-проектировщикам. Необходимо учесть, что большинство инженеров-проектировщиков не являются специалистами в численных методах, поэтому требуемая точность должна обеспечиваться по умолчанию. Конечно, в большинстве случаев, при моделировании реальных технических систем не требуется высокая математическая точность результатов, но корректность результатов должна быть соблюдена. Количественно или качественно неверное решение, выданное системой без предупреждения, может привести к серьёзным авариям.

Универсальный программный продукт не может гарантировать корректность решения во всех возможных случаях. Простым решением данной проблемы является тестирование. Оно оказывается полезным в двух отношениях – во-первых, оно позволяет идентифицировать затруднения программ при решении определенных классов задач. Во-вторых, позволяет на достаточно большом наборе задач установить соотношение между задаваемой и действительно достигнутой точностями приближенного решения.

Но возникает вопрос - что считать эталоном решения? Нельзя принимать за эталонное решение результат, полученный в системе, считающейся более точной. Единственный строгий подход в данном случае для оценки этого соотношения состоит в том чтобы использовать только такие тестовые задачи, точное решение которых заранее известно, например задачи с известным аналитическим решением.

**Набор тестовых задач.** В этой работе набор тестовых задач состоит из жестких систем обыкновенных дифференциальных уравнений (ОДУ). То есть систем для решения которых необходимо использовать неявные методы, где значение шага интегрирования должно быть исключительно малым [3].

Наличие известного аналитического решения накладывает существенные ограничения на выбор тестовых задач. В первую очередь тестовые уравнения должны охватывать все известные свойства жестких систем, которые могут существенно влиять на эффективность численных методов. Поэтому в каждую тестовую задачу заложено одно или несколько определенных свойств, чтобы проверить влияние этих свойств на надежность и эффективность программы.

Набор тестовых задач состоит из 12 жестких систем ОДУ [1]:

1. Линейная система с постоянной диагональной матрицей.
2. Линейная система с постоянными коэффициентами и слабосвязанными компонентами.
3. Линейная система с постоянными коэффициентами и сильно связанными компонентами.
4. Линейная система с постоянными коэффициентами. Матрица системы после преобразовании подобия содержит клетку Жордана третьего порядка.
5. Линейная система с переменными коэффициентами. Одно собственное значение в начале интервала интегрирования находится в положительной области.
6. Линейная система с переменными коэффициентами. Одно собственное значение по модулю возрастает.
7. Линейная система с переменными коэффициентами. Одно собственное значение по модулю убывает.
8. Линейная система с переменными коэффициентами. Одно собственное значение по модулю убывает. Отличается от предыдущего интервалом интегрирования.
9. Линейная система со слабо осциллирующим решением.
10. Линейная система со средне осциллирующим решением.
11. Линейная система с сильно осциллирующим решением.
12. Расширенная линейная система с сильно осциллирующим решением.

Тестирование сводилось к оценке максимального значения отклонения решений полученных численным и аналитическим способами.

**Тестируемый программный пакет.** На сегодняшний день MathCAD справедливо считается одним из лидеров в сфере инженерных вычислений. Он дает возможность инженеру легко производить любые необходимые вычисления. Как следствие, этот программный продукт является широко распространенным во всех сферах связанных с математическим моделированием и анализом процессов.

В пакете MathCAD 14 существует несколько методов для решения дифференциальных уравнений. Например, для решения жестких систем используются методы Булирша-Штера, Розенброка, RADAU5 (неявный жестко-устойчивый метод Рунге-Кутты до 5-го порядка точности), также используются формулы обратного дифференцирования [2]. Все системы были решены методами предназначенными для жестких систем. Но также были

протестированы методы для решения нежестких систем дифференциальных уравнений, и в некоторых случаях их результаты оказывались более точными.

**Результаты тестирования.** Программа MathCAD 14 справилась почти со всеми тестами. Это значит, что при решении тестов хотя бы один из доступных методов обеспечил минимально необходимую точность  $1e-04$ . По данным тестирования лучшие результаты были достигнуты при использовании метода Булирша-Штера. Следует отметить, что при решении систем с сильно осциллирующим решением (тесты 11 и 12) наилучшая достигнутая точность составила всего лишь  $1e-03$ , с использованием формул обратного дифференцирования. Однако решение все равно является качественно верным (Рис. 1). Например тест 11:

Исходная система:

$$\begin{aligned} \dot{x}_1(t) &= ax_1 + bx_2 \\ \dot{x}_2(t) &= -bx_1 + ax_2 \end{aligned}$$

Аналитическое решение:

$$\begin{aligned} x_1(t) &= (3 \cos bt + \sin bt) \exp(at) \\ x_2(t) &= (\cos bt - 3 \sin bt) \exp(at) \end{aligned}$$

Интервал интегрирования:  $[0,10]$

Параметры:  $a=-1$ ;  $b=100$

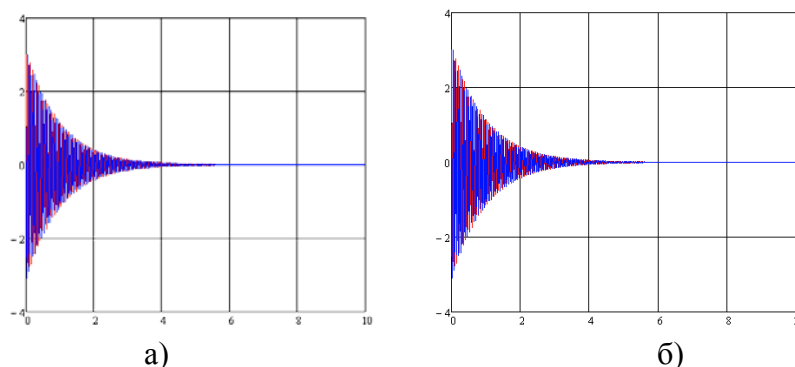


Рисунок 1 - Графики решения теста 11 полученные с помощью численного решения (а) и аналитическое решение (б)

Следует отметить тесты 7 и 8. Эти тесты отличаются лишь интервалом интегрирования ( $[0,10]$  – интервал для теста 7,  $[0.12,10]$  – интервал для теста 8), однако тест 7 является единственным тестом, который решается качественно неверно. Максимальное отклонение численного решения от аналитического исчисляется порядками (комбинированный метод Адамса и формул обратного дифференцирования). В тоже время, при решении этой системы на другом интервале (тест 8), обеспечивается высокая точность –  $1e-12$ , с использованием метода Булирша-Штера (Рис. 2).

Исходная система:

$$\begin{aligned} \dot{x}_1(t) &= \frac{1}{1-a}(\lambda_1(t) - a\lambda_2)x_1 + \gamma x_2 + \frac{1}{1-a}(\beta + a\gamma - \lambda_1(t))x_3 \\ \dot{x}_2(t) &= \frac{a}{1-a}(\lambda_1(t) - \lambda_2)x_1 + \beta x_2 + \frac{1}{1-a}(\beta + a\beta - a\lambda_1(t))x_3 \\ \dot{x}_3(t) &= \frac{a}{1-a}(\lambda_1(t) - \lambda_2)x_1 + \gamma x_2 + \frac{1}{1-a}(\beta + a\gamma - a\lambda_1(t))x_3 \\ \lambda_1(t) &= (s-p)(t\lambda + 1)e^{\lambda t} - s \end{aligned}$$

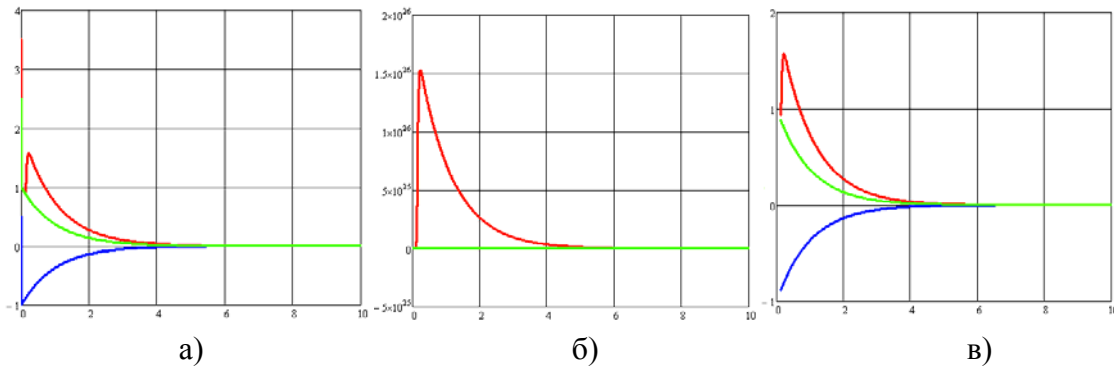


Рисунок 2 - Графики решения тестов 7 и 8: аналитическое решение (а), численное решение для теста 7 (б), численное решение для теста 8 (в)

### Выводы

В работе проведено тестирование программного пакета MathCAD 14 на корректность и точность решений. Набор тестовых задач состоял из специально подобранных жестких систем ОДУ. Результаты проведенного тестирования показали, что программа не до конца справляется с решением такого рода задач. Необходимо использование более эффективных численных методов, так как требуемая точность и качественный вид решения должны быть гарантированы всегда, независимо от вида задач.

### Литература

1. Загорин А.Н. Тестирование программ решения жестких систем обыкновенных дифференциальных уравнений. Материал информационного фонда РФАП Латвии. Инв.Но ИМ0020, ВЦ ЛГУ им. П.Стучки, Рига, 1984.
2. Хайер Э., Ваннер Г. Решение обыкновенных дифференциальных уравнений. Жесткие и дифференциально-алгебраические задачи. – М.: Мир, 1999.
3. Д.М. Жук, В.Б. Маничев, А.О. Ильницкий Методы и алгоритмы решения дифференциально-алгебраических уравнений для моделирования систем и объектов во временной области // Информационные технологии. - 2010. – часть 1 - №7, часть 2 - №8.

# ИЗГОТОВЛЕНИЕ ШАБЛОНА С НАПРАВЛЯЮЩИМИ ДЛЯ СВЁРЛ С ЦЕЛЮ УПРОСТИТЬ ПРОЦЕСС СВЕРЛЕНИЯ ОТВЕРСТИЙ В ЧЕЛЮСТИ ПАЦИЕНТА ВО ВРЕМЯ УСТАНОВКИ ЗУБНЫХ ИМПЛАНТАТОВ

**А. Михайловс**

*Научный руководитель к.т.н, доцент Д.М. Жук*  
МГТУ имени Н.Э.Баумана, кафедра РК6, Москва, Россия

## A TEMPLATE WITH GUIDING HOLES FOR DRILL PRODUCTION TO SIMPLIFY HOLES DRILLING PROCESS IN A PATIENT JAW AT A TIME OF PLACING A TOOTH IMPLANT

**A. Mihailovs**

*Supervisor D. Zhuk*

Bauman Moscow State Technical University

### **Аннотация**

Задачей является изготовление пластикового шаблона с нижней стороны частично повторяющего поверхность челюсти пациента, который бы помог просверлить отверстия в челюсти в нужном месте под нужным углом. Без шаблона такие отверстия может сделать только очень опытный хирург, тогда как шаблон допускает более широкий круг специалистов. Задача является экономически выгодной, так как найти опытного хирурга бывает непросто.

### **Abstract**

The goal is to produce a plastic template which could help drilling a hole in a jaw at right place with right angle. Without a template holes like this could make only very skillful surgeons but a template allows more wide range of professionals. The task is economically profitable because it could be difficult to find a skillful surgeon.

Для реализации данной задачи было выявлено несколько подходов:

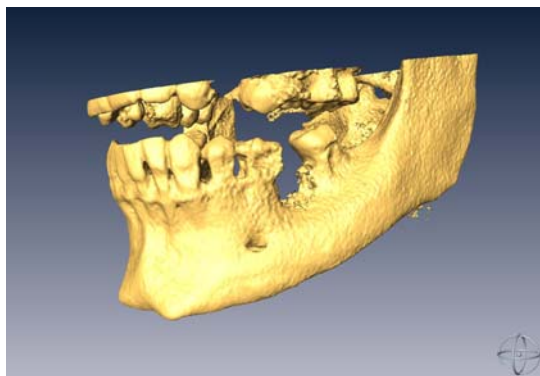
- использование готовых программных продуктов и немного программирования для доводки промежуточных результатов
- создание собственного программного комплекса для выполнения всех промежуточных стадий

С целью ускорить получение результата был выбран более простой подход к решению задачи – использовались уже готовые программные продукты, а результат выполнения некоторых подогнать для входных данных последующих.

Для начала необходимо сделать снимок челюсти человека с помощью томографа. Воспользуемся малодозным, так как он более распространен. Имплантолог крепит на челюсть человека контрастный материал, который задает направление и место сверления и делает снимок. Получаем некоторое количество файлов формата DICOM, из которых мы и будем выделять нужную нам поверхность.

DICOM (Digital Imaging and Communications in Medicine, цифровые изображения и обмен ими в медицине) - это индустриальный стандарт для передачи радиологических изображений и другой медицинской информации между компьютерами, опирающийся на стандарт Open System Interconnection (OSI), разработанный Международной организацией по стандартам (International Standards Organization, ISO). Стандарт DICOM описывает "паспортные" данные пациента, условия проведения исследования, положение пациента в момент получения изображения и т.п.

Существует программный продукт, называемый Amira, который способен читать данные DICOM и преобразовывать построенную на основе полутоновых изображений воксельную модель в единственный формат – VRML.



**Рис. 1.** Воксельная модель челюсти

Вóксел (в разговорной речи вóксель, англ. *Voxel* — образовано из слов: объёмный (англ. *volumetric*) и пиксел (англ. *pixel*)) — элемент объёмного изображения, содержащий значение элемента растра в трёхмерном пространстве. Воксели являются аналогами пикселей для трёхмерного пространства. Воксельные модели часто используются для визуализации и анализа медицинской и научной информации.

Программный продукт *Amiga* платный, но написан специально для работы с форматом *DICOM*, поэтому никаких нареканий не вызвал. Процесс работы достаточно прост. Для начала загрузим данные томографа — для этого нажимаем кнопку “*Open Data...*”. В открывшемся диалоговом окне выбираем все снимки томографа (файлов может быть много) и жмем загрузить. Теперь имеем набор снимков, по которым нужно построить 3д модель. Построение 3д модели происходит после нажатия на кнопку “*Isosurface*” и выставления необходимого *Threshold* (отклонение от максимальной контрастности вокселей модели). От *Threshold* зависит количество точек и, соответственно, качество модели. Задача определения оптимального *Threshold* решается отдельно для каждого набора снимков. Теперь осталось только экспортировать данные в формат *VRML*. Для этого нажмем кнопку “*VRML – Export*” и выберем место сохранения. В итоге имеем файл с описанием всех точек и связей между ними.

В качестве редактора 3д сеток выбираем программу *3ds max*, так как это один из наиболее простых в освоении и мощных редакторов на данный момент. *3ds max* позволяет не только редактировать поверхности, но и импортировать различные форматы данных, что нам и необходимо для работы.

Импортировать формат *VRML* программный продукт *3ds max* позволяет, начиная с версии 2010, но, для моделей с огромным количеством вершин этот процесс чрезвычайно длителен, и, зачастую, заканчивается неудачей. Поэтому мы вынуждены использовать дополнительное конвертирование в формат *3DS*, хорошо понимаемый программой *3ds max*. Конвертирование проходит в несколько этапов:

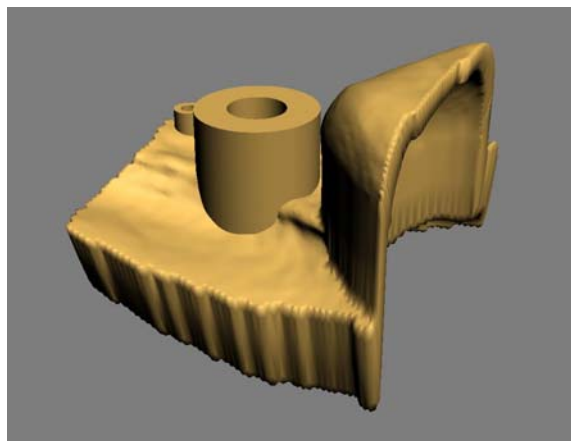
- загрузка *VRML* модели в память
- разделение модели на слои по максимальному количеству вершин в каждом (связано с особенностями *3DS* формата)
- сохранение модели в формат *3DS*

Теперь, воспользовавшись сторонним конвертером, в нашем распоряжении имеется *3DS* файл с информацией о модели челюсти, который пока что не пригоден к использованию из-за возникших трудностей и проблем на этапе создания. Все эти проблемы можно исправить в программе *3ds max* с помощью *MaxScript*, написав соответствующую программу.

*MaxScript* — это язык программирования *3ds max*, позволяющий автоматизировать монотонные процессы, на которые обычно уходит много времени. Если одни и те же операции постоянно повторяются, значит необходимо использовать *MaxScript*.

Для нормальной работы с 3д изображением необходимо исправить все проблемы, возникшие на этапе конвертации. Такими проблемами являются:

- несоответствие систем координат DICOM данных и 3d max сцены (модель получается повернутой на  $-90^\circ$  по оси X)
- раздробленность модели на слои (из-за особенностей формата 3ds имеем модель, разбитую на слои, для объединения которых необходимо слить совпадающие в пространстве вершины)
- модель отображается без сглаживания (для улучшения зрительного восприятия модель можно сгладить, добавив соответствующий модификатор)



**Рис. 2.** Полученный шаблон

Теперь, в нашем распоряжении имеется цельная модель, готовая к изменениям. На этом этапе нужно создать пластиковый шаблон, который бы своей нижней поверхностью повторял верхнюю поверхность челюсти пациента. Сделать это человеку, работающему с программой 3ds max, не составит труда. Для воплощения 3д модели в жизнь нужно лишь распечатать её на 3д-принтере. Для этого конвертируем нашу модель в бинарный STL и отправляем на печать.

В результате проделанной работы мы получили пластиковый шаблон, который поможет имплантологу при сверлении отверстий в челюсти.

По идее, конечной целью данного проекта является написание такого программного комплекса, который бы выполнял все описанные выше промежуточные функции самостоятельно, без использования сторонних программных продуктов. Поэтому, найденное решение является временным и, может существовать до тех пор, пока такой комплекс не будет написан. Предположительно, комплекс должен будет включать в себя такие этапы:

- чтение данных DICOM (в этом поможет SDK)
- построение изоповерхности используя методы трехмерной реконструкции (то, чем сейчас занимается Amira)
- редактор трехмерных сеток на основе OpenGL или Direct3D
- конвертер полученной модели в формат STL для отправки в печать

Реализовав предложенную концепцию, ручная работа останется только на этапе проектирования самого шаблона, но и её можно минимизировать.



# КОМБИНИРОВАННЫЙ МЕТОД ПОСТРОЕНИЯ АППРОКСИМАЦИИ МНОЖЕСТВА ПАРЕТО НА ГПУ С АРХИТЕКТУРОЙ *CUDA*

Антух А.Э., Семенихин А.С.  
д.ф.-м.н., проф. Карпенко А.П.  
МГТУ им. Н.Э. Баумана, Москва, Россия.

## COMBINED METHOD OF APPROXIMATE CONSTRUCTION OF THE PARETO SET ON *CUDA* ARCHITECTURE

A. Antukh, A. Semenikhin  
professor A. Karpenko.  
MSTU n.a. Bauman, Moscow, Russia.

### Аннотация

Рассматривается комбинация известного метода приближенного построения множества Парето «недоминируемая сортировка» и метода глобальной оптимизации роением частиц. Целью работы является исследование эффективности указанной комбинации методов при их реализации на графических процессорных устройствах с архитектурой *CUDA*

### Abstract

This paper discusses a combination of known method of approximate construction of the Pareto set “non-dominated sorting” and particle swarm optimization (*PSO*) method in a global optimization problem. Work purpose is efficiency research of these methods’ implementation on *GPU* with *CUDA* architecture.

**Введение.** Использование графических ускорителей (ГПУ) для научных расчетов началось относительно недавно [1]. Во многих прикладных областях до сих пор остается открытым вопрос об эффективности ГПУ по сравнению с классическими компьютерными системами. Целью данной работы является исследование эффективности ГПУ с архитектурой *CUDA* для задачи аппроксимации множества Парето с помощью метода роя частиц (*PSO*).

Основная масса публикаций, посвященных *CUDA*-вычислениям, ориентирована на задачи, обладающие явным параллелизмом по данным [2]. Такие задачи хорошо распараллеливаются и позволяют получать существенный прирост производительности вычислений по сравнению с последовательными вычислениями на *host*-ЭВМ. В задачах многокритериальной оптимизации не всегда присутствует параллелизм по данным и эффективность решения таких задач на ГПУ с архитектурой *CUDA* мало освещена в литературе. Исследование эффективности *CUDA*-вычислений при решении задачи однокритериальной оптимизации выполнено, например, в работе [3].

**Постановка задачи.** Задана совокупность частных критериев оптимальности  $\phi_1(X), \phi_2(X), \dots, \phi_m(X)$ , которые образуют векторный критерий  $\Phi(X)$ . Здесь  $X$  -  $n$ -мерный вектор варьируемых параметров [4]. Ставится задача минимизации каждого из указанных критериев в одной и той же области допустимых значений вектора варьируемых параметров  $D_X$ . Условно задача многокритериальной оптимизации записывается в виде

$$\min_{X \in D_X} \Phi(X) = \Phi(X^*). \quad (1)$$

Векторный критерий оптимальности  $\Phi(X)$  выполняет отображение множества  $D_X$  в некоторое множество  $D_\Phi$  пространства критериев, которое называется множеством достижимости задачи (1). Введем на этом множестве отношение предпочтения. Будем говорить, что вектор  $\Phi_1 = \Phi(X_1)$ ,  $X_1 \in D_X$  предпочтительнее вектора  $\Phi_2 = \Phi(X_2)$ ,  $X_2 \in D_X$  (или вектор  $\Phi_1$  доминирует вектор  $\Phi_2$ ), и писать  $\Phi_1 \succ \Phi_2$ , если среди равенств и неравенств  $\phi_k(X_1) \geq \phi_k(X_2), k \in [1 : m]$  имеется, хотя бы одно строгое неравенство.

Выделим из множества  $D_X$  подмножество точек  $D_\Phi^*$  (фронт Парето), для которых нет более предпочтительных точек. Множество  $D_X^* \in D_X$ , соответствующее множеству  $D_\Phi^*$ , называется множеством Парето [4]. Таким образом, если  $X \in D_X^*$ , то  $\Phi(X) \in D_\Phi^*$ .

Если  $\Phi(X_1) \succ \Phi(X_2)$ , то будем говорить, что вектор  $X_1$  предпочтительнее вектора  $X_2$  (или вектор  $X_1$  доминирует вектор  $X_2$ ), и писать  $X_1 \triangleright X_2$ .

Ставится задача приближенного построения множества Парето  $D_X^*$  или, что то же самое, фронта Парето  $D_\Phi^*$ , задачи многокритериальной оптимизации (1).

**Методы роя частиц и недоминируемой сортировки.** Множество частиц обозначим  $P = \{P_i, i \in [1:N]\}$ , где  $N$  – число частиц в рое (размер популяции). В дискретный момент времени  $t \in [0:T]$  координаты частицы  $P_i$  определяются вектором  $X_{i,t} = (x_{i,t,1}, x_{i,t,2}, \dots, x_{i,t,n})$ , а ее скорость – вектором  $V_{i,t} = (v_{i,t,1}, v_{i,t,2}, \dots, v_{i,t,n})$ ;  $T$  – число итераций. Начальные координаты и скорости частицы  $P_i$  равны  $X_{i,0} = X_i^0$ ,  $V_{i,0} = V_i^0$  соответственно.

Итерации в каноническом методе *PSO* выполняются по схеме [5]

$$V_{i,t+1} = \alpha V_{i,t} + U_1[0, \beta] \otimes (X_{i,t}^b - X_{i,t}) + U_2[0, \gamma] \otimes (X_{g,t} - X_{i,t}), \quad (2)$$

$$X_{i,t+1} = X_{i,t} + V_{i,t+1}. \quad (3)$$

Здесь  $U[a,b]$  представляет собой  $n$ -мерный вектор псевдослучайных чисел, равномерно распределенных в интервале  $[a,b]$ ;  $\otimes$  – символ покомпонентного умножения векторов;  $X_{i,t}^b$  – вектор координат частицы  $P_i$  с наилучшим значением целевой функции  $\Phi(X)$  за все время поиска  $[0:t]$ ;  $X_{g,t}$  – вектор координат соседней с данной частицы (в смысле используемой топологии соседства частиц) с наилучшим за время поиска  $[0:t]$  значением целевой функции  $\Phi(X)$ ;  $\alpha, \beta, \gamma$  – свободные параметры алгоритма. Важнейшее в методе *PSO* понятие топологии соседства и определено, например, в работе [5]. В процессе итераций вектор  $X_{i,t}^b$  образует так называемый собственный путь (private guide) частицы  $P_i$ , а вектор  $X_{g,t}$  – локальный путь (local guide) этой частицы.

Канонический метод роя частиц ориентирован на решение задач однокритериальной оптимизации. Для задачи многокритериальной оптимизации (1) используем модификацию этого метода, которая называется метод *MOPSO (Multi Objective Particle Swarm Optimization)* [6]. Важной частью метода *MOPSO* является определение глобально-лучшей (в смысле формулы (1)) частицы для каждого индивидуума в популяции. В силу специфики задачи многокритериальной оптимизации глобально-лучшая частица отыскивается на множестве Парето.

Существует несколько способов выбора глобально лучшей частицы. В данной работе используется метод «меняющихся соседей» Хью и Эберхарта [6]. Рассмотрим суть этого метода на примере задачи двухкритериальной оптимизации. Поиск глобально лучшей частицы для каждой частицы популяции осуществляется в пространстве критериев следующим образом. Сначала вычисляем расстояние от частицы  $P_i$  до других частиц, содержащихся в архиве  $A_i$ , используя значения первого («фиксированного») критерия оптимальности  $\phi_1(X)$ . Таким образом, для частицы  $P_i$  находим  $k$  ее ближайших локальных соседей. Затем, используя второй критерий  $\phi_2(X)$ , из числа указанных  $k$  соседей находим наилучшую частицу для частицы  $P_i$ , которая и полагается глобально лучшей частицей для частицы  $P_i$ .

Итерации продолжаются до тех пор, пока множество недоминируемых решений не перестанет меняться, либо до достижения заданного числа итераций.

**Реализация алгоритма и тестовая задача.** Рассматривается следующая двухмерная двухкритериальная тестовая задача, точные фронт и множество Парето для которой известны:

$$\begin{cases} \phi_1(X) = x_1^2 + x_2^2, \\ \phi_2(X) = (x_1 - 1)^2 + (x_2 - 1)^2; \end{cases} \quad (4)$$

$$D_X = \{X \mid -5 \leq x_i \leq 5, i = 1, 2\}. \quad (5)$$

Исследование плотности покрытия множества Парето и фронта Парето задачи (4, 5) при использовании метода *MOPSO* рассмотрено в работе [8]. В данной работе исследуется эффективность указанного метода при его реализации на ГПУ.

Исследование выполнено с использованием в качестве *host-ЭВМ* персональной ЭВМ, управляемой операционной системой *Windows XP*, в следующей конфигурации: процессор – *Intel Pentium Dual CPU E2160*, 1,8 ГГц; оперативная память – 2 Гб. В качестве ГПУ использована плата *NVidia GeForce 8500GT*, содержащая 16 потоковых процессоров. Архитектура *CUDA* рассмотрена, например, в работе [9].

В практически значимых задачах многокритериальной оптимизации основной объем вычислений связан с вычислениями значений частных критериев оптимальности. Например, если речь идет о задаче оптимизации сложной динамической системы, каждое такое вычисление требует численного интегрирования большой системы обыкновенных дифференциальных уравнений. Поэтому в работе принята следующая схема организации вычислений: метод *MOPSO* реализует *host-процессор* системы, а вычисления значений частных критериев оптимальности – ГПУ.

Эффективность параллельных вычислений оценивается с помощью ускорения, равного отношению времени решения задачи с использованием только *host-процессора* системы, к аналогичному времени при решении задачи с использованием ГПУ. Для моделирования разных вычислительных сложностей критериев (4) использовано их многократное вычисление. В качестве меры вычислительной сложности критериев использован коэффициент сложности  $C$ , равный числу этих вычислений.

**Результаты экспериментов.** Исследование выполнено при варьировании следующих параметров: число итераций  $T = 30; 50$ ; коэффициент сложности  $C = 1; 10; 100; 1000$ ; число частиц в популяции  $N = 32; 64; 128; 256$ . Поскольку эффективность метода существенно зависит от начальных параметров частиц  $X_i^0, V_i^0$  результаты исследования усреднены по этим начальным параметрам.

Некоторые результаты исследования представлены на рисунках 2, 3. При невысокой вычислительной сложности частных критериев оптимальности распараллеливание вычислений не эффективно, поскольку в этом случае велика доля коммуникационных расходов. При увеличении вычислительной сложности критериев, а также числа частиц в популяции, ускорение вычислений достигает величины 23.

Рисунки 2, 3 показывают практически линейный рост ускорения с ростом числа частиц  $N$ . Это обстоятельство позволяет надеяться на получение ускорения, превышающего максимально достигнутое в рассматриваемых экспериментах (равное 23) при дальнейшем увеличении числа частиц в популяции.

**Заключение.** Выполнено исследование эффективности комбинации метода *MOPSO* и метода недоминируемой сортировки при приближенном построении множества Парето в задаче многокритериальной оптимизации с помощью ГПУ с архитектурой *CUDA*. Исследование показало высокий потенциал использования графических ускорителей и библиотек для работы с ними (*CUDA*) для эффективного решения указанной задачи.

В развитие работы предполагается расширение класса тестовых задач многокритериальной оптимизации; определение для ГПУ различных архитектур оптимальных значений свободных параметров рассматриваемых методов, обеспечивающих максимальное ускорение вычислений; разработка методов теоретической оценки ускорения при заданных значениях свободных параметров методов и архитектуре ГПУ.

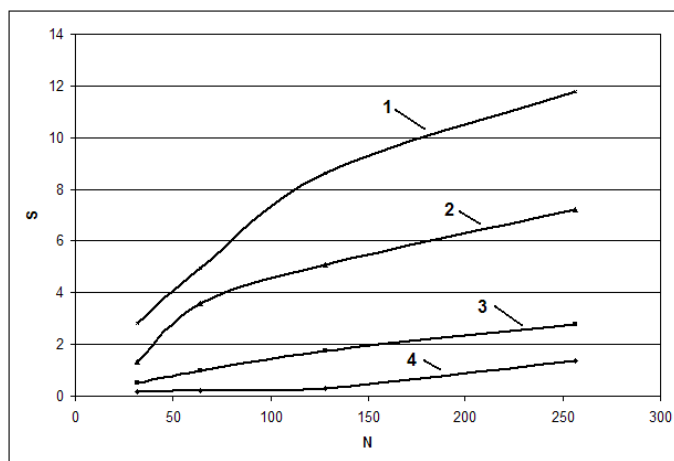


Рисунок 2 - Ускорение вычислений в функции числа частиц  $N$  при числе итераций  $T = 30$  :  
 1 -  $C = 1000$ ; 2 -  $C = 100$ ; 3 -  $C = 10$ ; 4 -  $C = 1$

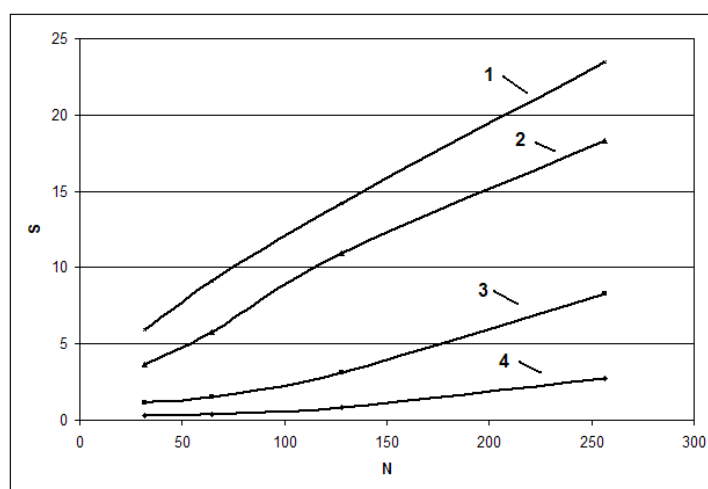


Рисунок 3 - Ускорение вычислений в функции числа частиц  $N$  при числе итераций  $T = 50$  :  
 1 -  $C = 1000$ ; 2 -  $C = 100$ ; 3 -  $C = 10$ ; 4 -  $C = 1$

### Литература

1. Вычисления на GPU. (<http://www.nvidia.ru/>).
2. NVIDIA CUDA C SDK Code. (<http://nvidia.com/>).
3. Карпенко А.П., Селиверстов Е.Ю. Глобальная безусловная оптимизация роём частиц на графических процессорах архитектуры CUDA // Наука и образование: электронное научно-техническое издание, 2010, 4. (<http://technomag.edu.ru/doc/142202.html>).
4. Штойер Р. Многокритериальная оптимизация. Теория, вычисления и приложения. М., Радио и связь, 1992. 504 с.
5. Карпенко А.П., Селиверстов Е.Ю. Глобальная оптимизация методом роя частиц. Обзор // Информационные технологии. 2010. № 2. С. 25-34.
6. Hu X., Eberhart R. Multiobjective optimization using dynamic neighborhood particle swarm optimization // World Congress on Computational Intelligence. Proceeding: 2002. P. 1677–1681.
7. Антух А.Э., Семенихин А.С., Хасанова Р.В. Построение множества Парето методом роя частиц на графических процессорах архитектуры CUDA // Научный сервис в сети Интернет: суперкомпьютерные центры и задачи, Труды международной суперкомпьютерной конференции (21-26 сентября 2009 г., г. Новороссийск). М.: Изд-во МГУ, 2010. С. 274-280.
8. Фролов В. Введение в технологию CUDA. (<http://cgm.computergraphics.ru/issues/issue16/cuda>).

# ОСОБЕННОСТИ ОБРАБОТКИ ВХОДНОЙ ТОПОЛОГИЧЕСКОЙ ИНФОРМАЦИИ В САПР СБИС ДЛЯ РАЗЛИЧНЫХ АРХИТЕКТУР ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

**Верстов В.А.**

*Научный руководитель Зинченко Л.А., д-р техн. наук, профессор*  
МГТУ им. Н.Э. Баумана, г. Москва, Россия

## FEATURES OF THE INPUT LAYOUT INFORMATION PROCESSING IN VLSI CAD FOR HIGH-PERFORMANCE COMPUTING SYSTEMS

**Verstov V. A.**

*Advisor Zinchenko L. A., D.Sc., professor*  
BMSTU n.a. N.E. Bauman

### **Аннотация**

В статье рассматриваются проблемы обработки входной топологической информации в САПР СБИС. Файлы, описывающие топологии современных СБИС в формате GDSII, являются крайне большие (до нескольких сотен гигабайт). Обсуждаются вопросы использования высокопроизводительных систем для обработки входных файлов топологии.

### **Abstract**

Paper describes problems of the input layout information processing in VLSI CAD. GDS II stream format (GDSII) is a database file format which is the de facto industry standard for data exchange of integrated circuit or IC layout artwork. The size of GDSII files could be about hundreds gigabytes. It was shown that processing of the input layout information should be run on high-performance computer systems.

### **Введение**

Сложность расчетов при проектировании и производстве СБИС увеличивается с переходом к меньшим топологическим нормам. В настоящее время осваиваются топологические нормы 65, 45, 32 нм в производстве интегральных микросхем. При проектировании и производстве СБИС важную роль играют процессы фотолитографии. Для создания фотошаблонов традиционно используются файлы в формате GDS II (GDSII, GDS) — формат файлов баз данных, являющийся де-факто промышленным стандартом для обмена данными по интегральным схемам и их топологиям. Данный формат описывает плоские геометрические формы, текстовые метки и иную информацию в иерархической форме. Для хранения файлов в формате GDSII, описывающих системы на кристалле (System-on-Chip (SoC)), состоящие из множества сложно-функциональных блоков (Intellectual Property блоков, IP-блоков), необходимы значительные объемы памяти [6]. Обработка больших объемов памяти требует значительных вычислительных ресурсов. Есть два пути решения проблемы получения значительных вычислительных мощностей: создание суперкомпьютеров или использование кластеров.

Суперкомпьютер - вычислительная машина, значительно превосходящая по своим техническим параметрам большинство существующих компьютеров. Однако в конце 90-х годов высокая стоимость специализированных суперкомпьютерных решений и нарастающая потребность в доступных вычислительных ресурсах привели к широкому распространению компьютерных кластеров. Эти системы характеризует использование отдельных узлов на основе дешёвых и широко доступных компьютерных комплектующих для серверов и персональных компьютеров и объединённых при помощи мощных коммуникационных систем и специализированных программно-аппаратных решений. Несмотря на кажущуюся простоту, кластеры довольно быстро заняли достаточно большой сегмент суперкомпьютерного рынка, обеспечивая высочайшую производительность при минимальной стоимости решений [3].

Кластерное решение позволяет достичь наилучшего соотношения цены и производительности, именно оно является в настоящее время наиболее перспективным для конструирования компьютеров с рекордными показателями производительности.

В настоящее время границы между специализированным суперкомпьютерным и стандартным программным обеспечением сильно размыты и продолжают размываться ещё более вместе с проникновением технологий параллелизации и многоядерности в процессорные устройства персональных компьютеров и рабочих станций. Исключительно суперкомпьютерным программным обеспечением сегодня можно назвать лишь специализированные программные средства для управления и мониторинга конкретных типов компьютеров, а также уникальные программные среды, создаваемые в вычислительных центрах под «собственные», уникальные конфигурации суперкомпьютерных систем.

## **1 Особенности использования высокопроизводительных вычислительных систем в САПР СБИС**

Существует два подхода к реализации высокопроизводительных вычислений: многопроцессорные вычислительные системы и многокомпьютерные вычислительные системы. Причем каждый отдельный вычислительный узел такой системы может представлять собой многопроцессорную систему. Существует несколько видов возможной архитектуры таких компьютеров [4]:

1. Uniform Memory Access (сокращённо UMA — «однородный доступ к памяти») — архитектура многопроцессорных компьютеров с общей памятью. Все микропроцессоры в UMA-архитектуре используют физическую память одновременно. При этом время запроса к данным из памяти не зависит ни от того, какой именно процессор обращается к памяти, ни от того, какой именно чип памяти содержит нужные данные. Однако каждый микропроцессор может использовать свой собственный кэш.
2. NUMA (Non-Uniform Memory Access — «неравномерный доступ к памяти») или Non-Uniform Memory Architecture — «Архитектура с неравномерной памятью») — схема реализации компьютерной памяти, используемая в микропроцессорах, когда время доступа к памяти определяется её расположением по отношению к процессору.

Мультимикрокомпьютер — вычислительная система без общей памяти, состоящая из большого числа взаимосвязанных компьютеров, у каждого из которых имеется собственная память. Процессоры мультимикрокомпьютера отправляют друг другу сообщения (используется сетевая топология двух-, трехмерной решетки или дерева и кольца) [3]. Причина разработки — техническая сложность создания мультимикропроцессоров. Кластер является одной из наиболее эффективных реализаций мультимикрокомпьютерных систем.

Кластер — группа компьютеров, объединённых высокоскоростными каналами связи и представляющая с точки зрения пользователя единый аппаратный ресурс.

Виды кластеров [1]:

1. Кластеры высокой доступности - обозначаются аббревиатурой HA (англ. High Availability — высокая доступность). Создаются для обеспечения высокой доступности сервиса, предоставляемого кластером.
2. Кластеры распределения нагрузки - принцип их действия строится на распределении запросов через один или несколько входных узлов, которые перенаправляют их на обработку в остальные, вычислительные узлы.
3. Вычислительные кластеры - кластеры используются в вычислительных целях, в частности в научных исследованиях. Специально выделяют высокопроизводительные кластеры (Обозначаются англ. аббревиатурой HPC Cluster — High-performance computing cluster).

Как правило, современные НРС-кластеры представляют собой большое число высокопроизводительных серверных компьютеров, соединённых друг с другом локальной высокоскоростной магистралью для достижения максимальной производительности в рамках подхода распараллеливания вычислительной задачи [2].

Идея распараллеливания вычислений основана на том, что большинство задач может быть разделено на набор меньших задач, которые могут быть решены одновременно. Обычно параллельные вычисления требуют координации действий. Параллельные вычисления использовались много лет в основном в высокопроизводительных вычислениях, но в последнее время к ним возрос интерес вследствие существования физических ограничений на рост тактовой частоты процессоров.

В 1965 году Гордон Мур высказал предположение, что число транзисторов на кристалле будет удваиваться каждые 24 месяца [5]. В последнее время, чтобы получить возможность задействовать на практике ту дополнительную вычислительную мощность, которую дает закон Мура, стало необходимо задействовать параллельные вычисления.

Характер увеличения скорости программы в результате распараллеливания объясняется законом Амдала. Закон Амдала иллюстрирует ограничение роста производительности вычислительной системы с увеличением количества вычислителей. Для получения математического выражения закона в случае параллелизации воспользуемся предположением:  $P$  — доля от общего числа вычислений, которая может быть распараллелена идеально, а соответственно  $(1 - P)$  — доля от общего числа вычислений, которая может быть получена только последовательными расчетами; тогда ускорение, которое может быть получено на вычислительной системе из  $N$  процессоров будет описываться выражением:

$$S_N \leq \frac{1}{(1 - P) + \frac{P}{N}}$$

На рис. 1 графически показана зависимость роста производительности системы в зависимости от количества процессоров при различных долях параллельных вычислений.

Из графика видно, что только алгоритм, вовсе не содержащий последовательных вычислений ( $P = 1$ ), позволяет получить линейный прирост производительности с ростом количества процессоров в системе.

Отсюда же очевидно, что при доле последовательных вычислений  $1 - P$  общий прирост производительности не может превысить  $1/(1 - P)$ . Так, если половина кода — последовательная, то общий прирост никогда не превысит двух.

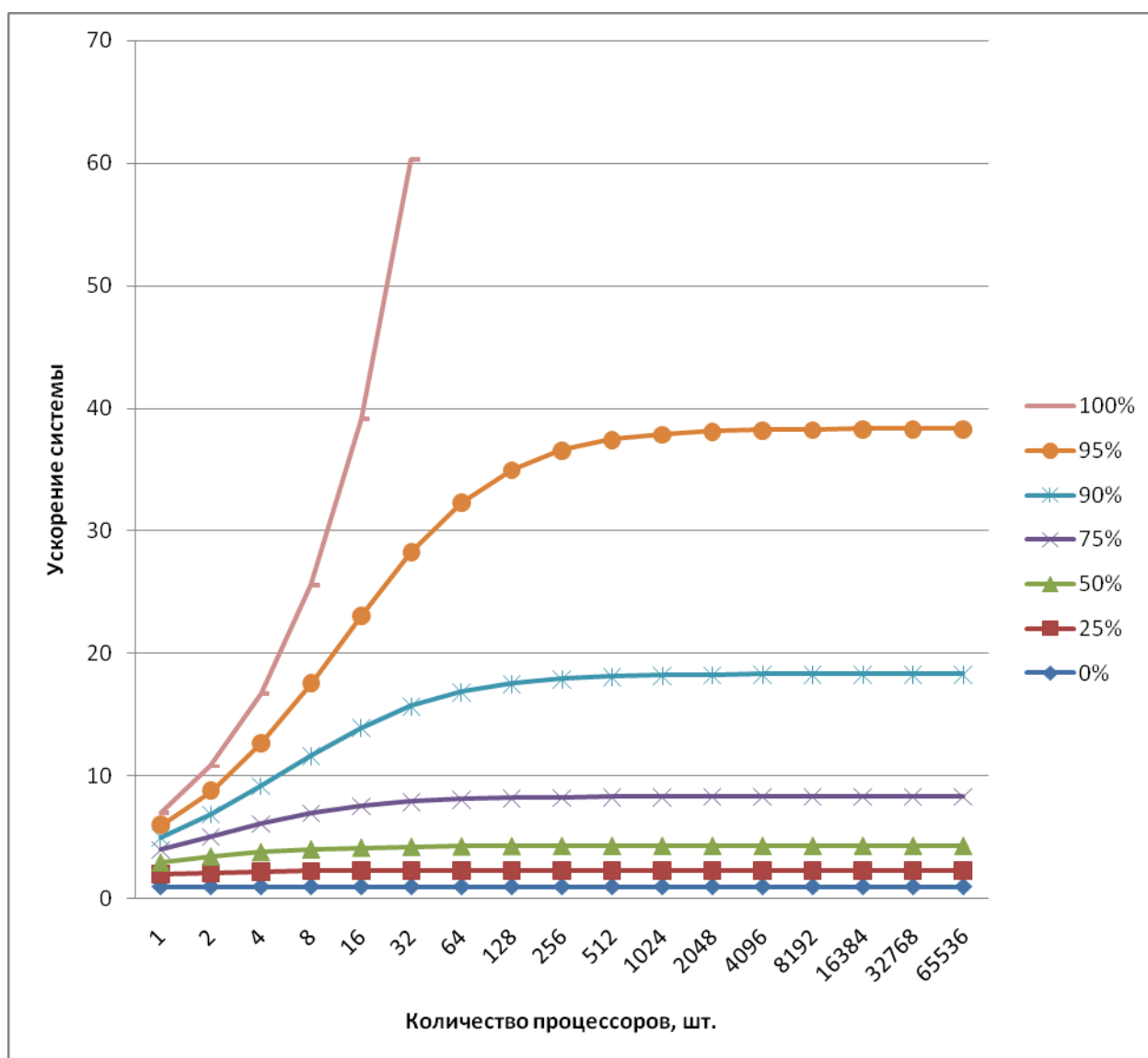


Рисунок 1 - Ускорение, которое может быть получено на вычислительной системе из  $N$  процессоров по сравнению с однопроцессорным решением

Закон Амдала показывает, что прирост эффективности вычислений зависит от алгоритма задачи и ограничен сверху для любой задачи с  $P \neq 1$ . Не для всякой задачи имеет смысл наращивание числа процессоров в вычислительной системе.

Более того, если учесть время, необходимое для передачи данных между узлами вычислительной системы, то зависимость времени вычислений от числа узлов будет иметь максимум. Это накладывает ограничение на масштабируемость вычислительной системы, то есть означает, что с определенного момента добавление новых узлов в систему будет увеличивать время расчёта задачи.

Таким образом, для получения максимально возможного ускорения необходимо максимально увеличить долю параллельно выполняемых операций в алгоритме, на котором строится САПР СБИС.

## 2. Использование кластерных систем для обработки входной топологической информации

Графовые модели являются наиболее эффективным подходом для различных операций с топологией СБИС. Построение графов сразу для всех геометрических объектов топологии СБИС не обязательно, т.к. конструкторские, технологические и физические ограничения



имеют место быть только для соседних объектов. Следствием этого является то, что возможно разбиение исходной задачи трансформации на ряд локальных, каждую из которых можно выполнять параллельно на различных вычислительных узлах кластера, что должно существенно снизить временные затраты и позволить обрабатывать достаточно большие по размеру файлы топологии, которые один компьютер не в состоянии выполнить из-за ограниченности его вычислительных ресурсов.

Топология современных СБИС достаточно сложна, и как следствие этого алгоритм для обработки входной топологической информации должен учитывать объемы данных, которые необходимо обработать. На рис. 2 показан пример топологии СБИС.

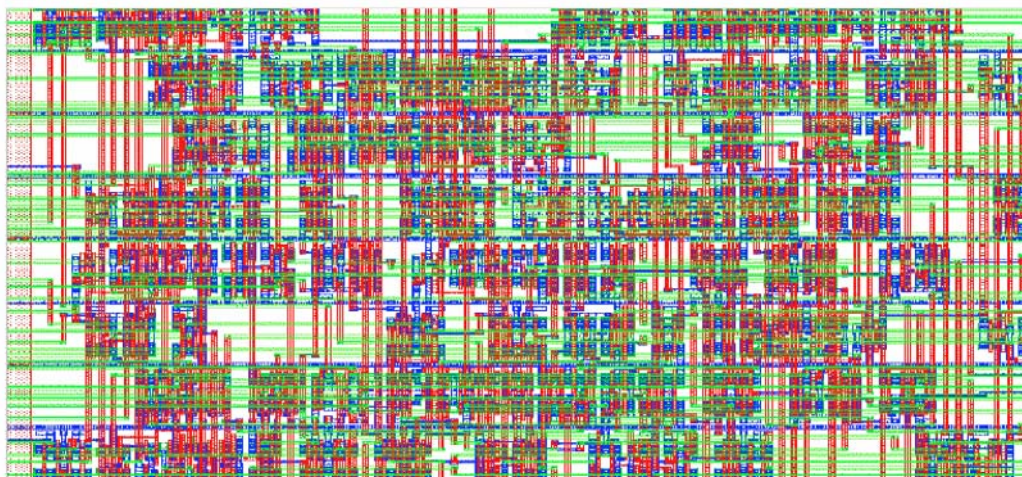


Рисунок 2 - Топология СБИС

В программировании параллельный алгоритм, противопоставляемый традиционным последовательным алгоритмам, — алгоритм, который может быть реализован по частям на множестве различных вычислительных устройств с последующим объединением полученных результатов и получением корректного результата. Параллельные алгоритмы весьма важны ввиду постоянного совершенствования многопроцессорных систем и увеличения числа ядер в современных процессорах. Сложность последовательных алгоритмов выражается в объеме используемой памяти и времени (числе тактов процессора), необходимых для выполнения алгоритма. Параллельные алгоритмы в свою очередь требуют учета подсистемы связей между различными процессорами. Существует два способа обмена между процессорами: использование общей памяти и системы передачи сообщений. Еще одной проблемой, связанной с использованием параллельных алгоритмов, является балансировка нагрузки между процессорами или вычислительными узлами. Разновидность параллельных алгоритмов, называемая распределенными алгоритмами, специально разрабатываются для применения на кластерах с учетом их особенностей.

Идея распараллеливания вычислений основана на том, что большинство задач может быть разделено на набор меньших задач, которые могут быть решены одновременно. Параллельные вычисления существуют в нескольких формах [7]:

1. Параллелизм на уровне битов основан на увеличении размера машинного слова. Увеличение размера машинного слова уменьшает количество операций, необходимых процессору для выполнения действий над переменными, чей размер превышает размер машинного слова.
2. Параллелизм на уровне инструкций заключается в распределении инструкций по группам, которые будут выполняться параллельно, без изменения результата работы всей программы.
3. Параллелизм данных — выполнение одной операции сразу над всеми элементами массива данных. Различные фрагменты такого массива обрабатываются на разных

процессорах параллельной машины. Распределением данных между процессорами занимается программа.

4. Параллелизм задач (многопоточность) основан на разбиении вычислительной задачи на несколько относительно самостоятельных подзадач и каждый процессор загружается своей собственной подзадачей.

Первые две формы параллелизма реализуются в архитектуре процессоров. Следовательно, распараллеливание обработки входной топологической информации можно реализовать либо используя параллелизм данных, а именно представить входную информацию как массив данных и параллельно обрабатывать его элементы на различных процессорах, либо используя принцип параллелизма задач, например, разрезать топологию на равные геометрические части и производить их обработку параллельно на разных процессорах.

### **Заключение**

В статье рассмотрены особенности использования высокопроизводительных компьютерных систем в виду проблем, возникающих при обработке входной топологической информации в САПР СБИС, приведено сравнение возможных реализаций таких систем. Показано, что параллельные вычисления позволяют существенно увеличить производительность моделирования и проектирования микро- и наносистем при помощи специализированного ПО (специализированных САПР), снизить временные затраты на проектирование. Предложена возможность использования кластерных систем для обработки входной топологической информации САПР СБИС.

Работа выполнена при частичной финансовой поддержке гранта Президента РФ по государственной поддержке ведущих научных школ (грант НШ-3484.2010.9).

### **Литература**

1. Barak A., Gunday S. and Wheeler R., The MOSIX Distributed Operating System, Load Balancing for UNIX. Lecture Notes in Computer Science, Vol. 672, Springer-Verlag, 1993, с. 37
2. Using MPI: Portable Parallel Programming with the Message-Passing Interface by William Gropp, Ewing Lusk, and Anthony Skjellum, 1994, с. 151
3. Суперкомпьютерные технологии в науке, образовании и промышленности/Под редакцией: академика В. А. Садовниченко, академика Г. И. Савина, чл.-корр. РАН Вл. В. Воеводина.-М.: Издательство Московского университета, 2009.-232 с.
4. Rodgers, David P. (June 1985). "Improvements in multiprocessor system design". ACM SIGARCH Computer Architecture News archive (New York, NY, USA: ACM) 13 (3): 225–231 с.
5. Moore G. Cramming More Components Onto Integrated Circuits Gordon Moore // Electronics. – 1965, №8.
6. Немудров В., Мартин Г. Системы-на-кристалле. Проектирование и развитие.- М.: Техносфера, 2004.- 212 с.
7. Roosta, Seyed H. (2000). "Parallel processing and parallel algorithms: theory and computation". Springer, p. 114.

---

Работа выполнена в рамках реализации ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009 – 2013 годы.

# ПОВЫШЕНИЕ ТОЧНОСТИ ПРОГНОЗИРОВАНИЯ УРОВНЯ ГЛЮКОЗЫ В КРОВИ БОЛЬНЫХ САХАРНЫМ ДИАБЕТОМ ПЕРВОГО ТИПА

Косоруков К.Н., Чернецов С.А.  
*Научный руководитель д.ф.м.н., проф., Карпенко А.П.*  
(МГТУ им. Н.Э. Баумана, Москва, Россия)

## INCREASING THE ACCURACY OF BLOOD GLUCOSE FORECASTING IN TYPE I DIABETES PATIENTS

**K. Kosorukov, S.Chernetsov**  
*Supervisor prof., A. Karpenko*  
(MSTU n.a. Bauman, Moscow, Russia)

### Аннотация

В данной работе рассматривается подход к увеличению точности прогноза уровня сахара в крови пациентов с сахарным диабетом I типа за счет увеличения числа учитываемых параметров. В работе показано, что учет расширенного вектора параметров позволяет существенно увеличить точность прогноза.

### Abstract

The approach to increasing the blood glucose forecast accuracy in type I diabetes patients is considered here. The paper demonstrates that taking the expanded input parameter vector considerably increases forecast accuracy.

**Введение.** Работа выполнена в рамках исследований по разработке автоматической системы прогнозирования оптимальных доз инсулина, которая при объединении с инсулиновой помпой и сенсором непрерывного мониторинга уровня глюкозы в подкожной ткани пациента, могла бы выполнять функции искусственной поджелудочной железы.

Известно значительное число публикаций, посвященных задаче прогнозирования уровня глюкозы в крови (*BG - Blood Glucose*) [1]. Данная работа продолжает наши работы [2 - 4] и посвящена исследованию точности нейросетевых алгоритмов прогнозирования значений уровня *BG*. В работе существенно используются понятия «гликемический индекс» и «физическая активность».

Гликемический индекс (англ. *Glycemic (Glycaemic) Index, GI*) - показатель влияния продуктов питания после их употребления на уровень *BG*. Индекс отражает сравнительную реакцию организма на данный продукт с его реакцией на чистую глюкозу, у которой гликемический индекс принят равным 100. Гликемический индекс продуктов питания изменяется от 0 до 100 и более единиц, пропорционально скорости усвоения организмом этих продуктов. Низкий гликемический индекс продукта означает, что при его употреблении уровень *BG* растет медленно. Чем выше гликемический индекс продукта, тем быстрее увеличивается уровень *BG* после употребления продукта и тем выше одномоментный уровень *BG*.

Физическая активность (*Physical Activity, PA*) – субъективный индекс, изменяющийся от 0 до 10 единиц пропорционально двигательной активности человека. Физическая активность улучшает процессы метаболизма в организме человека и повышает чувствительность организма к инсулину. Поэтому при высоком уровне физической активности для понижения уровня *BG* требуемые дозы инсулина могут быть меньше.

Исследование выполнено с использованием программной системы *MatLab* (версия 7.9).

**Постановка задачи.** Имеется выборка данных реального пациента, которая кроме информации, представленной на рисунке 1, содержит значения гликемического индекса *GI* и индекса физической активности *PA*. На рисунке приняты следующие обозначения: *BG* – уровень глюкозы в крови; *insulin (I)* - дозы введенного инсулина; *food, (C)* - количество углеводов потребленных с пищей.

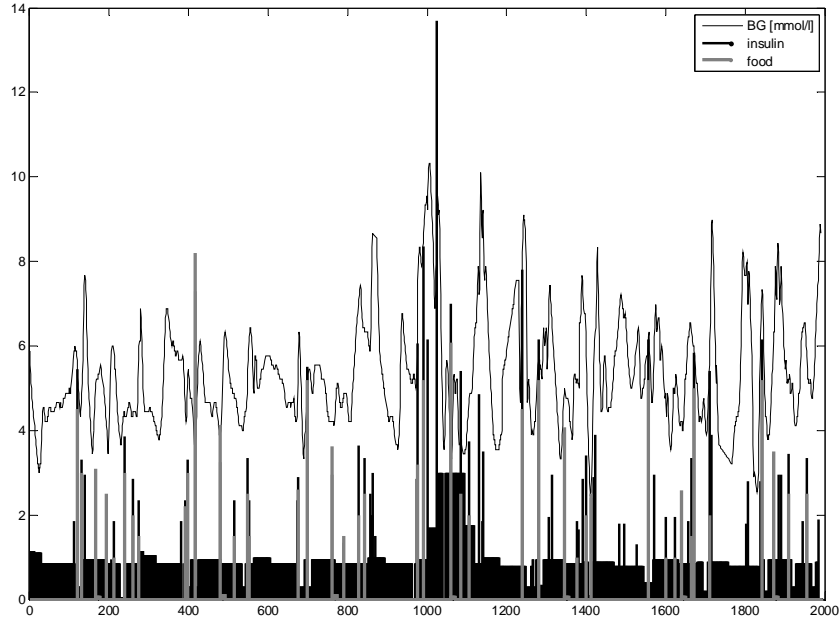


Рисунок 1 - Пример исходных данных

Величины  $I$ ,  $C$ ,  $GI$ ,  $PA$  полагаются заданными на дискретной временной сетке  $\Delta = (\Delta_{1-1}, \Delta_2)$  с постоянным шагом  $\delta t = 20$  минут, где

$$\Delta_1 = (t_{-n}, t_{-(n-1)}, \dots, t_{-1}, t_0), \Delta_2 = (t_1, t_2, \dots, t_m).$$

Здесь  $t_0$  - текущий момент времени;  $\Delta_1$  - сетка «предыстории»;  $\Delta_2$  - сетка, на которой выполняется прогноз.

Ставится задача прогнозирования уровня  $BG$  на период прогноза  $T = [t_l - t_0]$ ,  $l \leq m$ .

Эффективность прогнозирования оценивается среднеквадратической ошибкой (*mean square error, mse*) и величиной *mape* (*mean absolute percentage error*):

$$mse = \frac{1}{l} \sum_{i=1}^l e_i^2, \quad mape = \frac{1}{l} \sum_{i=1}^l \left| \frac{e_i}{Y_i} \right| * 100.$$

Здесь  $e_i = Y_i - \tilde{Y}_i$ ,  $i \in [1:l]$  - ошибка прогнозирования в момент времени  $t_i$ ;  $Y_i$ ,  $\tilde{Y}_i$  - точное и прогнозные значения величины  $Y \in \{I, C, GI, PA\}$  в момент времени  $t_i$  соответственно.

**Сводка основных предыдущих результатов [2 - 4].** Исследование выполнено на выборке данных длиной 1900 значений, при этом первые 1000 значений использованы для обучения нейронных сетей ( $n = 1000$ ), а оставшиеся 900 значений - для их тестирования.

Рассмотрено три типа нейронных сетей: рекуррентная нейронная сеть (*Layer-Recurrent Network, LRN-сеть*) и ее частный случай - сеть Элмана (*Elman network*); нелинейная авторегрессионная сеть (*Nonlinear Autoregressive Network, NARX-сеть*).

*LRN-сетью.* Исследование эффективности *LRN*-сетей выполнено при варьировании числа нейронов в сети, алгоритма обучения, архитектуры сети, длительности периода прогнозирования и длины обучающей выборки.

Результаты исследования показывают, что лучшим алгоритмом обучения *LRN*-сети является алгоритм градиентного спуска (*traingdx-algorithm*). По сравнению с *LRN*-сетью, нейронная сеть Элмана обеспечивает более высокую точность прогнозирования. В сети Элмана нецелесообразно использовать число нейронов скрытого слоя, превышающее 40. Точность прогнозирования резко падает с увеличением длительности периода прогноза, и приемлемую точность сеть Элмана обеспечивает при длительности периода прогноза, не

превышающей 40 мин. Увеличение длины обучающей выборки до двух дней существенно не улучшает точность прогноза.

Таким образом, показано, что *LRN*-сети позволяют получить достаточно хорошие результаты при краткосрочном прогнозировании и обучении алгоритмом градиентного спуска. При числе нейронов скрытого слоя, большем 49, точность прогноза почти не зависит от этого числа. *LRN*-сети не могут быть использованы для долгосрочных прогнозов.

*NARX-сети*. Исследование эффективности *NARX*-сетей выполнено при варьировании следующих величин: число нейронов в промежуточном слое сети; длина линии задержки; алгоритм обучения сети.

Результаты исследования показывают, что наиболее высокую точность прогнозирования обеспечивает число нейронов в промежуточном слое, равное пяти. Лучшие результаты достигаются при длине линии задержки, равной двум. Значительно лучшие результаты прогнозирования показывают *NARX*-сети, обученные методом градиентного спуска.

**Повышения точности прогнозирования.** Рассматриваются два типа нейронных сетей – многослойная сеть прямого распространения (*Feed-Forward Back-propagation Network, FF-сеть*) и *NARX-сеть*.

При обучении указанных сетей в качестве обучающей выборки использовались данные за 27 и 40 дней (1920 и 2880 отсчетов); в качестве тестовой выборки использовались данные за 27 и 13 дней (1920 и 960 отсчетов соответственно).

Если не оговорено противное, приняты следующие значения параметров сетей: число нейронов - 10; алгоритм обучения – Левенберга-Марквардта для *FF*-сети и градиентного спуска с возмущением для *NARX*-сети; длительность периода прогноза – 20 минут; длина обучающей выборки соответствует одному месяцу измерений; число циклов обучения - 100.

*Сглаживание входных данных* выполнено с помощью специальных линий задержки. Рассмотрено сглаживание по трем точкам. Результаты исследования показывают, что, как для *FF*-, так и для *NARX*-сетей, использование сглаживания позволяет уменьшить среднеквадратичную ошибку прогнозирования на 5 - 10%.

*Расширение вектора входных данных.* Некоторые результаты данного исследования представлены на рисунке 3. Расширение вектора входных данных рассматриваемых нейронных сетей за счет включения в него помимо уровней *BG*, инсулина и углеводов еще и индекса и индекса физической активности, позволяет существенно повысить точность прогнозирования – для *FF*-сетей среднеквадратичная ошибка прогнозирования снижается на ~10%, а для *NARX*-сетей – на ~15%.

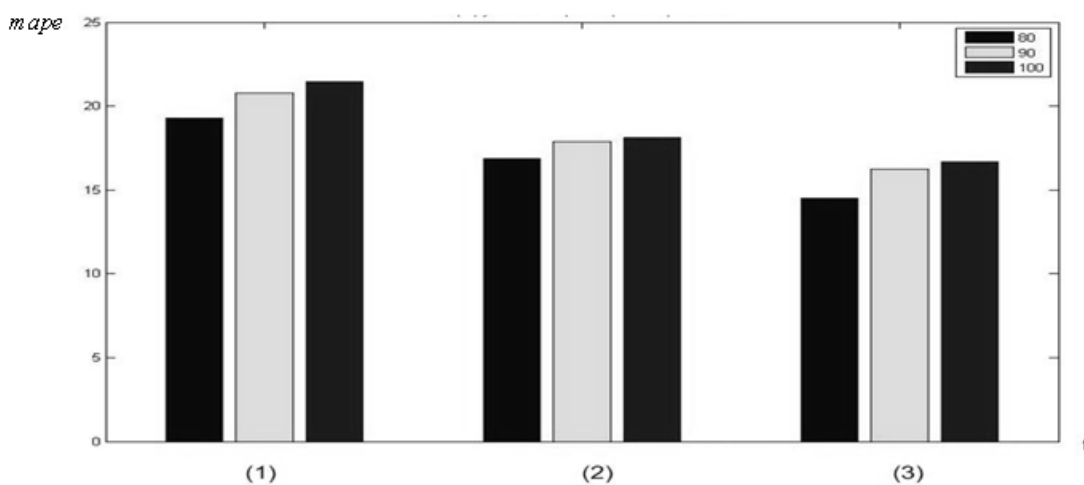


Рисунок 3 - Влияние числа входных параметров на точность прогнозирования, где прогноз делался на основе только глюкозы(1); глюкозы, инсулина и углеводов(2); глюкозы, инсулина, углеводов, гликемического индекса пищи и физической активности (3)

*Варьирование критериев оценки эффективности прогнозирования.* Рассмотрено использование в качестве указанных критериев величин *mse* (среднеквадратичная ошибка) и *mare* (средняя относительная ошибка). Результаты исследования показывают, что с точки зрения точности прогнозирования использование указанных критериев приводит к примерно одинаковым результатам. Однако с точки зрения удобства интерпретации результатов, критерий эффективности *mare* оказывается существенно более предпочтительным.

### Заключение

Результаты исследования показывают, что использование сглаживания входных данных позволяет уменьшить среднеквадратичную ошибку прогнозирования на 5 - 10%. Расширение вектора входных данных позволяет повысить точность прогнозирования на 10% - 15%. Использование *mse*- и *mare*-критериев оценки эффективности прогнозирования приводит к примерно одинаковым результатам.

### Литература

1. Гоменюк С.М., Емельянов А.О., Карпенко А.П., Чернецов С.А. Обзор методов прогнозирования оптимальных доз инсулина для больных сахарным диабетом I типа // Информационные технологии, 2010, №3, с. 46-57.
2. Емельянов А.О., Чернецов С.А. Нейросетевое прогнозирование уровня глюкозы в крови для больных инсулинозависимым диабетом // XII Всероссийская научно-техническая конференция «Нейроинформатика-2010»: Сборник научных трудов. Ч.2. М.: НИЯУ МИФИ, 2010, С. 326-334.
3. Абрамов Ю.В., Емельянов А.О., Карпенко А.П., Чернецов С.А. Исследование эффективности нейросетового прогнозирования оптимальных доз инсулина для больных сахарным диабетом I типа // Информационные и математические технологии в науке и / Труды XV Байкальской Всероссийской Конференции. Часть II., -Иркутск: ИСЭМ СО РАН, 2010, С. 156–164.
4. Карпенко А.П., Сабуров А.М., Чернецов С.А. Нейросетевое прогнозирование оптимальных доз инсулина для больных сахарным диабетом I типа // Труды Девятого Международного симпозиума «Интеллектуальные системы» (INTELS'2010), 28 июня – 2 2010 г., г. Владимир, С. 477 – 480.

---

Работа выполнена при частичной поддержке гранта РФФИ №10-08-00816-а.

# ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ ОПТИМИЗАЦИИ МЕТОДОМ НЕПРЕРЫВНО ВЗАИМОДЕЙСТВУЮЩЕЙ КОЛОНИИ МУРАВЬЕВ (CIAC).

Чернобривченко К.А.  
д.ф.-м.н., проф. Карпенко А.П.  
МГТУ им. Н.Э. Баумана, Москва, Россия.

## RESEARCH OF THE EFFICIENCY OPTIMIZATION BY CONTINUOUS INTERACTING ANT COLONY (CIAC).

K. Chernobrivchenko  
professor A. Karpenko.  
MSTU n.a. Bauman, Moscow, Russia.

### Аннотация

Рассматривается задача непрерывной многомерной глобальной условной оптимизации и алгоритм CIAC (алгоритм непрерывно взаимодействующей колонии муравьев) решения этой задачи. Приводятся результаты широкого исследования эффективности указанного алгоритма при решении ряда тестовых многоэкстремальных и овражных задач оптимизации. На основе результатов исследования даны рекомендации по оптимизации выбора свободных параметров алгоритма.

### Abstract

This paper discusses a problem of multi-dimensional continuous global constrained optimization and continuous interacting ant colony algorithm to solve this problem. The work presents the results of extensive research on the effectiveness of this algorithm for solving a number of test multiextremal and ravine optimization problems. Recommendations about optimization of a choice of free parameters of algorithm have been made on the basis of results of research.

**Введение.** В настоящее время значительное внимание уделяется развитию метаэвристических поведенческих многоагентных методов оптимизации таких, как метод роя частиц, метод пчелиного роя, метод колонии муравьев.

Первым алгоритмом, основанным на идее функционирования колонии муравьев, стал алгоритм Ant System (система муравьев) [1]. Алгоритм был использован при решении многих комбинаторных задач, таких как задача коммивояжера, задача раскраски графа и др. Первым «муравьиным» алгоритмом, созданным для решения задач непрерывной оптимизации, стал алгоритм CACO (*Continuous Ant Colony Optimization* - непрерывная оптимизация колонией муравьев) [2]. Взаимодействие агентов в алгоритме CACO осуществляется на основе модели стигмертии - использования муравьями феромонов для пометки лучших путей.

Реализация идеи стигмертии не является единственным способом использования муравьиного интеллекта при решении задач оптимизации. Так в работе [3] предложена модель, в которой канал стигмертии проигнорирован, а взаимодействие между агентами осуществляется на межиндивидуальном уровне с использованием механизма, названного мобилизацией. Алгоритм получил название CIAC (*Continuous Interacting Ant Colony* - непрерывно взаимодействующая колония муравьев).

Доклад посвящен исследованию эффективности алгоритма CIAC при решении ряда тестовых задач непрерывной многомерной безусловной оптимизации.

**Постановка задачи.** Рассмотрим задачу многомерной глобальной условной оптимизации: найти максимум целевой функции  $\Phi(X)$ , определенной на множестве  $D$  евклидова пространства  $R^n$ ,

$$\max_{X \in D \subset R^n} \Phi(X) = \Phi(X^*) = \Phi^*, \quad (1)$$

где  $X = (x_1, x_2, \dots, x_n)$  - вектор варьируемых параметров,

$$D = \{x \mid x_i^- \leq x_i \leq x_i^+, i \in [1:n]\} - \quad (2)$$

множество допустимых значений вектора варьируемых параметров, представляющее собой гиперпараллелепипед.

**Метод непрерывно взаимодействующей колонии муравьев.** Метод *CIAC* основан на моделировании двух каналов взаимодействия агентов колонии: канале стигмертии (при помощи феромонов) и канале прямого взаимодействия. Множество агентов обозначим  $\mathbf{A} = \{A_i, i \in [1:\eta]\}$ , где  $\eta$  – число муравьев в колонии. Каждый агент системы совершает перемещения в определенных направлениях в соответствии с двумя каналами связи.

Первый канал связи (феромонный) побуждает муравья двигаться к центру масс феромонных точек, координаты которого рассчитываются по формуле

$$G_j = \sum_{i=1}^N \bar{\varpi}_{i,j} X_i, \quad (3)$$

где  $X_i$  - вектор координат  $i$ -ой феромонной точки,  $N$  - их общее число,

$$\bar{\varpi}_{i,j} = \frac{\varpi_{i,j}}{\sum_{i=1}^N \varpi_{i,j}} - \quad (4)$$

нормированный «интерес»  $j$ -го муравья к  $i$ -ой феромонной точке, определяемый по формуле

$$\varpi_{i,j} = \frac{\bar{\delta}}{2} \theta_i \exp(-\delta_{i,j}). \quad (5)$$

Здесь  $\bar{\delta}$  - среднее расстояние между двумя агентами в популяции,  $\theta_i$  - количество феромона в точке  $X_i$ ,  $\delta_{i,j}$  - расстояние между  $j$ -м муравьем и  $i$ -ой феромонной точкой.

Муравей  $A_j$  делает скачек в направлении центра тяжести  $G_j$  на случайное расстояние в пределах своего размерного параметра  $r_j = \frac{d}{\psi}$ , где  $d \in [0,1]$  – случайная величина,  $\psi$  - делитель размерного параметра.

Второй канал связи опирается на индивидуальное взаимодействие агентов, реализованное путем передачи сообщений. В соответствии с этим каналом связи муравей  $A_j$  берет случайное сообщение из своей памяти и сравнивает значение целевой функции  $\Phi(X)$  в своей точке со значением, полученным из сообщения. Если значение целевой функции из сообщения лучше, то муравей  $A_j$  совершает скачек в случайную точку гиперсферы, центром которой являются координаты, указанные в сообщении, а радиусом - размерный параметр  $r_j$  муравья  $A_j$ . Если значение целевой функции лучше в точке расположения муравья  $A_j$ , то этот муравей посылает сообщение со своими координатами и значением целевой функции случайному агенту популяции, а затем совершает случайное перемещение в окрестности точки  $X_j$  в пределах размерного параметра  $r_j$ .

Закончив перемещение, муравей оставляет новую феромонную точку, значение феромона в которой, по сравнению с предыдущей, пропорционально улучшению целевой функции.

После того, как все муравьи популяции завершат свои перемещения, моделируется испарение феромонных точек. Если при этом значение феромона в точке становится меньше определенного минимума, то эта точка удаляется.

Затем проверяется критерий останова, в качестве которого может использоваться условие достижения заданного числа итераций  $\lambda_{stop}$  или следующие условие: расстояние между двумя последовательными лучшими точками не превышает заданную величину  $\varepsilon_X$  в течение  $10\eta$  итераций.

**Реализация алгоритма и тестовые задачи.** Алгоритм *CIAC* реализован в среде программной системы *MatLab*. Исследование эффективности алгоритма выполнено на многоэкстремальной функции Растргина (*Rastrigin*)



$$\Phi(X) = -\sum_{i=1}^n (10 + x_i^2 - 10 \cos(2\pi x_i)) \rightarrow \max_X \quad (6)$$

и одноэкстремальной овражной функции Розенброка (*Rosenbrock*)

$$\Phi(X) = -100((x_2 - x_1^2)^2 + (1 - x_1)^2) \rightarrow \max_X \quad (7)$$

из известного пакета тестовых функций *CES* (*Congress of Evolutionary Computing*) [4].

**Результаты экспериментов.** Исследование каждой функции проводилось при числе измерений, равном два и пять. При этом производилось варьирование следующих параметров: максимального числа сообщений в памяти муравьев  $\mu_{\max}$ , что позволяет оценить вклад канала индивидуально взаимодействия между агентами в общую эффективность работы алгоритма; делителя размерного параметра  $\psi$ , что позволяет найти компромисс между скоростью, вероятностью локализации экстремума и точностью.

Для примера, ниже приведены результаты исследования при варьировании величины  $\psi$  на пятимерной функции Растригина (рисунки 1-3).

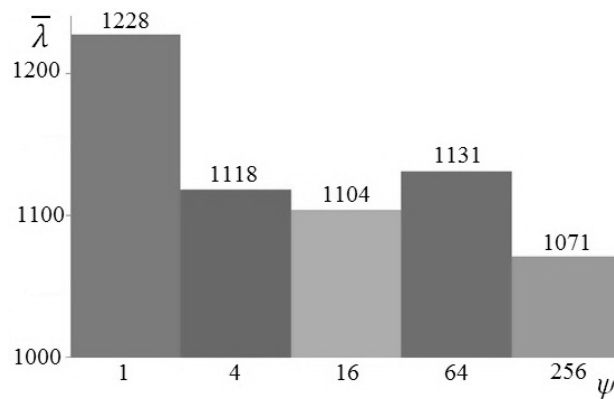


Рисунок 1 – Усредненное значение числа итераций в функции величины  $\psi$ : функция Растригина;  $n = 5$

На гистограммах мы видим зависимости усредненного числа итераций от значения делителя размерного параметра  $\psi$ , а также усредненной погрешности и вероятности локализации глобального максимума. Видно, что при увеличении величины  $\psi$  (другими словами, при уменьшении размерного параметра  $r_j$ ) точность локализации решения значительно возрастает. Однако вероятность локализации глобального максимума при этом падает со 100% до 0%.

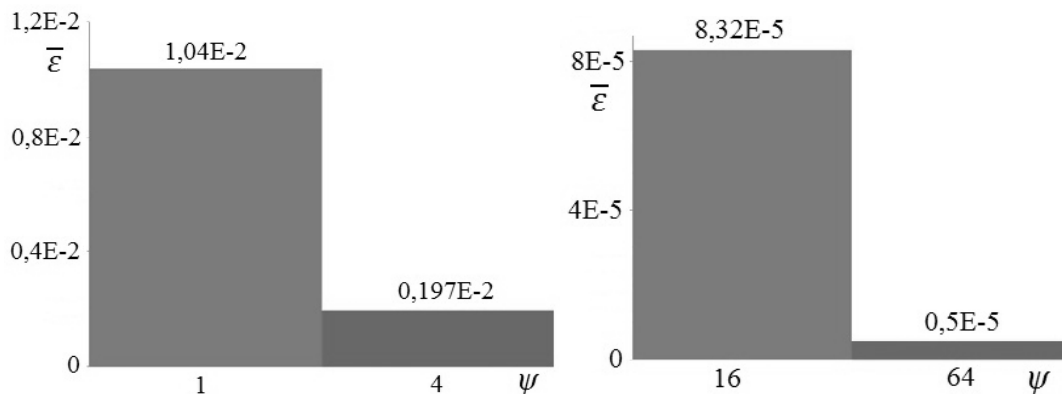


Рисунок 2 – Усредненное значение погрешности решения в функции величины  $\psi$ : функция Растригина;  $n = 5$

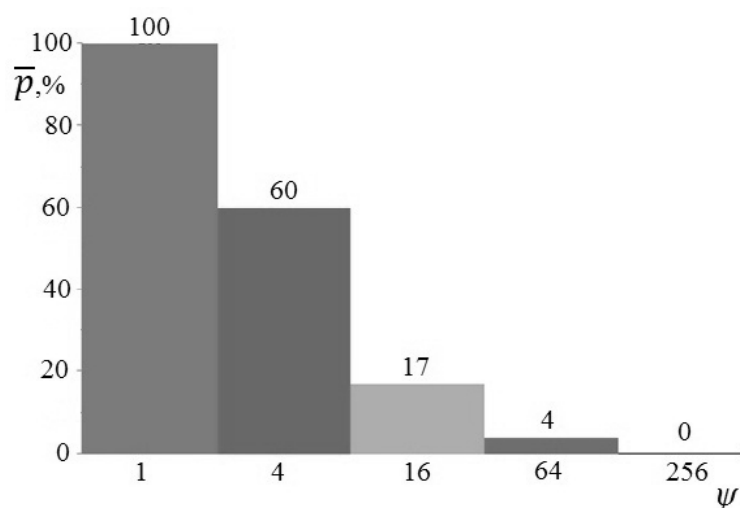


Рисунок 3 – Усредненная вероятность  $\bar{p}$  локализации глобального максимума с заданной точностью  $\varepsilon \leq 10^{-3}$ : функция Растригина;  $n = 5$

### Заключение

В работе реализован алгоритм муравьиной колонии *CIAC*. Реализация произведена в среде *MatLab*. Приложение способно работать как с одномерными, так и многомерными функциями.

Было проведено исследование возможности применения алгоритма для поиска минимумов одно- и пятимерных функций Растригина и Розенброка. Было проведено широкое исследование эффективности указанного алгоритма при решении ряда тестовых многоэкстремальных и овражных задач оптимизации. На основе результатов исследования были даны рекомендации по оптимизации выбора свободных параметров алгоритма.

### Литература

1. A. Colomi, M. Dorigo, V. Maniezzo. Distributed optimization by ant colonies // Proceedings of the First European Conference on Artificial Life, ECAL'91.- Elsevier, Paris, France, 1992.- pp. 34–142.
2. G. Bilchev, I.C. Parmee. The ant colony metaphor for searching continuous design spaces // Lect. Notes Comput. Sci., 1995, 993.- pp. 25–39.
3. J. Dréo, P. Siarry/ Continuous interacting ant colony algorithm based on dense heterarchy // Future Generation Computer Systems, 2004, 20.- pp. 841–856.
4. Tang K., Yao X., Suganthan P.N., MacNish C., Chen Y.P., Chen C.M., Yang Z. Benchmark Functions for the CEC'2008 Special Session and Competition on Large Scale Global Optimization. - Nature Inspired Computation and Applications Laboratory, USTC, China, 2007.

# ИССЛЕДОВАНИЕ И РАЗРАБОТКА ПМК ДЛЯ ДИСТАНЦИОННОГО КОНТРОЛЯ И РАСПОЗНАВАНИЯ 3D ОБЪЕКТОВ

Сахаров М.К.

*Научный руководитель к.т.н. доцент Волосатова Т.М.*

Кафедра РК6, МГТУ им. Н.Э. Баумана, Москва, Россия

## SOFTWARE RESEARCH AND DEVELOPMENT FOR REMOTE MONITORING AND RECOGNITION OF 3D OBJECTS

Sakharov M.K.

*The scientific chief: Ph.D. assistant professor Volosatova T.M.*

Department RK6, BMSTU, Moscow, Russia

### Аннотация

В работе приведен обзор решений задачи распознавания 3D объектов. Рассмотрены их технические особенности. Подробно проанализирован метод создания цифровой голографии. Разработано ПО для записи и воспроизведения цифровой голограммы по входному изображению. Проиллюстрированы результаты работы ПО.

### Abstract

The article provides an overview of the solutions for the task of the 3D objects recognition. Its technical features were considered. The method of creating a digital holography was analyzed in details. Software for recording and playback of a digital holography was developed. The program output was included into the article.

Известно, что компьютеры не могут «видеть» также как человек. Фотокамеры не эквивалентны системе зрения человека и не могут опираться на догадки. Системы машинного зрения должны «видеть» путем изучения отдельных пикселей изображения, обрабатывая их и пытаясь сделать выводы с помощью базы знаний и системы распознавания образов. Применение машинного зрения разнообразно, оно охватывает различные области деятельности, включая, но, не ограничиваясь крупной промышленностью, системами безопасности и визуального контроля качества.

**Постановка задачи.** Разработка ПМК для дистанционного распознавания и контроля трехмерных объектов задача крайне трудоёмкая и разнонаправленная. В данной сфере не существует какого-либо классического решения. Среди множества различных подходов, для каждой задачи можно выделить ту или иную технологию, которая является оптимальной. При разработке данного программного комплекса реализовывались различные подходы, каждый из которых является предпочтительным в каком-либо отдельно взятом случае - это может быть формирование 3D модели по фотоизображениям объекта или создание динамической картины перемещения.

В качестве среды реализации программного комплекса был выбран пакет Matlab, обладающий широкими возможностями для решения задач оптимизации, обработки сигналов и изображений. Язык Matlab оптимизирован для работы с матрицами и многомерными массивами, что позволяет эффективно использовать его для попиксельной обработки входных изображений.

В работе рассматривается метод, основанный на использовании цифровой голографии. Основным преимуществом данного подхода является возможность получения не только графического изображения объекта (амплитудная картина), но и его 3D отображения (фазовая картина), что особенно полезно при анализе перемещений.

**Цифровая голография.** Метод, о котором в дальнейшем пойдет речь, отличается от общепринятых технологий распознавания, широкими возможностями для контроля передвижения объекта. При перемещении исследуемого предмета в пространстве его фазовая картина будет не только перемещаться, но и видоизменяться, что позволяет легко отслеживать даже незначительные перемещения.

Словосочетание «цифровая голография» указывает на то, что практическая реализация голографического метода записи и воспроизведения информации осуществляется не в физическом эксперименте, а с помощью моделирования указанных процессов численными методами с использованием вычислительной техники [1].

Процесс создания цифровой голограммы условно состоит из двух частей. Записи волнового поля, то есть формирования интерференционной картины с заданной объектной волной и последующей ее регистрации. А затем воспроизведения волнового поля, то есть формирования опорной волны и считывания голограммы.

Алгоритм получения цифровых голограмм можно представить в виде следующих этапов:

1. Ввод голографируемого участка изображения в ЭВМ.
2. Вычисление амплитудного и фазового спектров изображения с помощью определенных алгоритмов
3. Выполнение подготовительных процедур, зависящих от выбранного алгоритма выдачи цифровой голограммы из ЭВМ.
4. Вывод синтезированной голограммы.

В дальнейшем восстановление исходного изображения по его цифровой голограмме может проводиться как вычислительными, так и оптическими (физическими) методами [4].

В данной работе был реализован алгоритм синтеза голограмм Фурье простейших объектов. Фурье-голограммой называется голограмма, на которой регистрируется интерференция двух волн, комплексные амплитуды которых в плоскости голограммы являются Фурье-образами предмета и опорного источника [5]. В нашем случае был использован точечный опорный источник.

**Реализация метода.** Любая система, входными данными которой является графическая информация, должна реализовывать некий алгоритм обработки изображений. Изображение является двумерным сигналом. Оно обрабатывается в частотной области, с использованием двумерного дискретного преобразование Фурье (ДПФ). Преобразование Фурье позволяет нам описать функцию с помощью совокупности составляющих ее частот (Рис. 1). Дискретное преобразование Фурье в среде Matlab выполняется по быстрому алгоритму (БПФ). Сложность вычислений в этом случае определяется зависимостью  $O(N \log(N))$ , где  $N$  - количество пикселей в одном из измерений изображения [2].

Разработанная программа состоит из отдельных модулей (функций), направленных на выполнение конкретных задач, например формирование видео фрагмента, восстановление записанной голограммы и т.п.

Отметим, что основной задачей стоящей перед данной программой является создание и анализ цифровой голограммы объекта, без возможности его распознавания и сравнения с содержимым некоторой базы знаний. Также не предусмотрена возможность обучения программы.

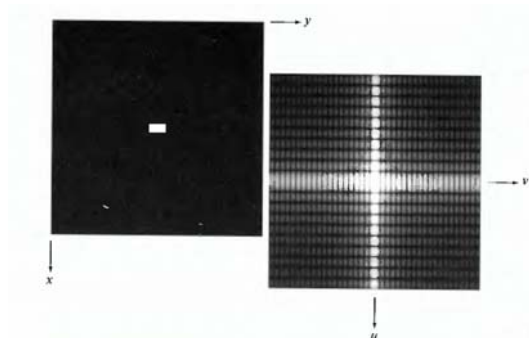


Рисунок 1 - Частотный спектр, полученный с помощью преобразования Фурье

Далее приведено описание работы созданной программы. Исходные данные передаются в рабочую функцию и преобразуются в изображение, состоящее из оттенков серого. При использовании простых объектов этого достаточно. В дальнейшем потребуется

выделение контура объекта и основных граней с последующей бинаризацией изображения. После этого генерируется точечный опорный пучок с амплитудой и фазой волны, задаваемыми пользователем на основе рекомендаций программы.

С помощью алгоритма быстрого преобразования Фурье (БПФ) из исходного изображения получается спектральная картина. Затем моделируется интерференция волн объекта и опорного пучка. После записи голограммы начинается процесс ее воспроизведения. Моделируется освещение голограммы восстанавливающим пучком, который является комплексно-сопряженным к опорному пучку. С помощью обратного быстрого преобразования Фурье (ОБПФ) получается амплитудная картина, на которой виден интерференционный максимум в центре, действительное и перевернутое мнимое изображения по разные стороны от максимума (Рис. 2).

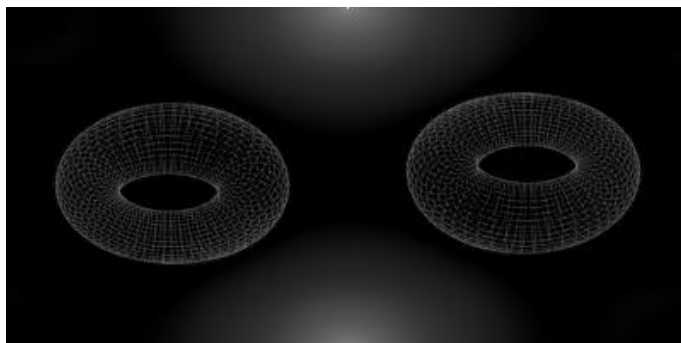


Рисунок 2 - Амплитудная картина обрабатываемого объекта

После создания амплитудной картины начинается синтезирование фазовой картины, для получения более четкого представления о положении объекта в пространстве. Программа работает со спектром, полученным после обработки входного изображения с помощью БПФ. Спектр представляет собой матрицу комплексных значений амплитуды. Элементы фазовой матрицы находятся как арктангенс отношения мнимой и действительной частей соответствующего значения спектра комплексных амплитуд. Изменение именно этих значений фазовой матрицы позволяет нам отследить перемещение объекта (Рис. 3).

В программе предусмотрена возможность создания видеофрагмента по предварительно обработанным кадрам. Процесс записи в реальном времени отображается на экране. Последовательность кадров может быть как созданной заранее, так и подготовленной вручную средствами данной программы. Все результаты можно сохранить в виде графических или видео файлов. Фазовый портрет сохраняется в численной форме, в виде исполняемого файла среды Matlab [3].

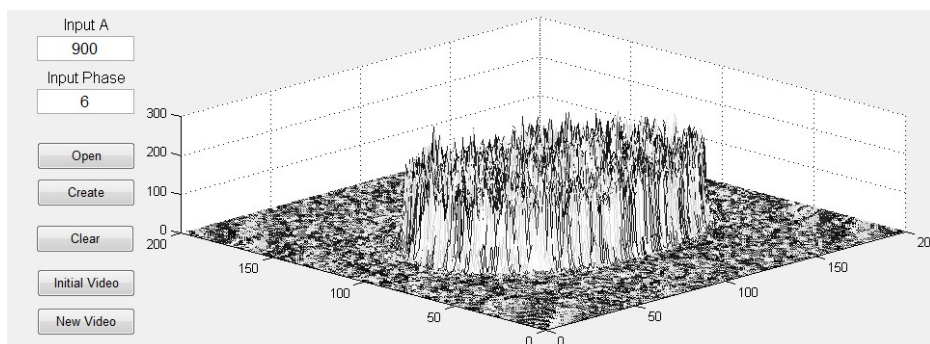


Рисунок 3 - Фазовая картина обрабатываемого объекта в главном окне программы

В процессе работы программы формируются промежуточные результаты для анализа голограммы, которые также могут быть сохранены. Введено ограничение на размер обрабатываемого изображения для обеспечения оптимального быстродействия. В целях стандартизации параметров ввода введено ограничение на формат входных данных –

программа использует изображения формата jpg и png. Предусмотрены стандартные проверки корректности ввода данных.

Созданная программа имеет встроенное руководство пользователя, включая краткое теоретическое описание метода создания голограммы. Графический интерфейс пользователя (GUI) состоит из главного окна, в котором вводятся все необходимые данные, и отображаются результаты работы. Интерфейс программы обеспечивает контекстный вызов справки.

### **Заключение**

С помощью программы пользователь получает возможность создавать цифровую голограмму, проводить анализ голограммы и сохранять полученные результаты. Выходными файлами программы являются графические файлы формата jpg и видео файлы формата avi, трехмерное изображение фазовой картины может быть сохранено в виде числового файла среды Matlab. Предусмотрена возможность варьирования входных параметров для анализа зависимости создаваемой голограммы от параметров опорного пучка. Результирующая фазовая картина, являющаяся особенностью данного подхода, позволяет отслеживать даже незначительные перемещения наблюдаемого объекта. Программа стабильно функционирует и способна вести диалог с пользователем для контроля вводимой информации.

### **Литература**

1. Schnars U., Juptner W. Digital Holography. Springer, 2005.
2. Гонсалес Р., Вудс Р., Эддинс С. Цифровая обработка изображений в среде MATLAB. Москва: Техносфера, 2006.
3. Дьяконов В.П. MATLAB. Полное руководство. – М.: ДМК Пресс, 2010.
4. Андреева О.В. Прикладная голография, Санкт-Петербург, 2008.
5. Ярославский Л.П., Мерзляков Н.С. Методы цифровой голографии, Наука, 1977.

## ГЕНЕРАТОР ИНФОРМАЦИОННЫХ РЕСУРСОВ

**Зорин Е. Л., Кокорева Н. М., Макутонин Р. С., Ядаринкин Д.С.**  
*Научный руководитель: к. т. н., доцент Медведев Н.В.*  
кафедра ИУ8, МГТУ им. Н.Э. Баумана, Москва, Россия

## GENERATOR OF INFORMATION RESOURCES (GIR)

**Zorin E. L., Kokoreva N. M., Makutonin R. S., Yadarinkin D.S.**  
*Scientific adviser: Ph.D., Assoc. Prof. Medvedev N.V.*  
department IU8, BMSTU, Moscow, Russia

### **Аннотация**

В работе рассматривается генератор информационных ресурсов, сопровождающий в настоящее время учебный процесс кафедры информационной безопасности. Основные составляющие проекта - программное средство автоматизации процесса формирования учебных планов и подсистема защиты от несанкционированного доступа. Программное обеспечение позволяет строить понятийный граф, устанавливающий семантическую взаимосвязь дисциплин; дает возможность проводить расчеты требуемого рабочего времени студентов и преподавателей; автоматизирует задачу определения загрузки аудиторного фонда и технических средств обучения. Подсистема защиты успешно устраняет противоречия между требованием обеспечения доступа широкому кругу пользователей (до 700 студентов) и требованием защиты от несанкционированного копирования материалов.

### **Abstract**

The article provides an overview of the generator of information resources (GIR), which maintains at the present time the learning process of the Department "Information security" (IU8). The main components of the project are the software tool of the curricula compilation automation and the security subsystem. The software of GIR enables the user to construct a conceptual graph that establishes a semantic connection among subjects; provides an opportunity to perform calculations of the working time, required by students and teachers; automates the process of the aggregate of classrooms and laboratory equipment workload determination. Security subsystem successfully resolves the contradiction between the requirement for a wide range of users access (up to 700 students) and the requirement for protection against unauthorized copying of data and unauthorized access.

Повсеместное распространение компьютерной техники и связанных с ней информационных технологий порождает новые направления информатизации деятельности человека. С этой точки зрения образовательный процесс не является исключением. В частности, вычислительные средства применяются как в профессиональной подготовке студентов и школьников, так и при решении различных вопросов, связанных с планированием и организацией обучения. Разработанная система представляет собой средство автоматизированного планирования и сопровождения учебного процесса. Кроме того, она предлагает решение неизбежно возникающих при проектировании подобного рода систем проблем безопасности. Таким образом, разработчики стремились к повышению эффективности процесса обучения путем обеспечения удобства, надежности и оперативности формирования планов учебных дисциплин и их содержания.

**Общие сведения.** Целью разработки является создание электронной системы (далее по тексту - Электронная Энциклопедия), предназначенной для автоматизированного планирования и сопровождения учебного процесса. Электронная энциклопедия представляет собой систематизированную совокупность графической, текстовой, цифровой, речевой, музыкальной, видео-, фото- и другой информации. Она способна к развертыванию на любом несъемном носителе и/или в компьютерной сети.

Существенным отличием большинства аналогичных средств является то, что Электронная Энциклопедия не может быть редуцирована к бумажному варианту без потери дидактических свойств.

Высокое качество визуальной и аудио - информации становится доступным благодаря широким возможностям современной технологии мультимедиа. Представление графической информации при использовании компьютерных технологий делает доступной динамическую

интерпретацию изучаемых свойств не только реальных объектов, но и абстрактно – теоретических (виртуальных).

В состав программного обеспечения Электронной Энциклопедии, предназначенного для организации учебного процесса, входят сервисные программные средства общего назначения, программные средства для математического и имитационного моделирования, программные средства лабораторий (используется удаленный доступ), информационно-поисковые справочные системы, автоматизированные обучающие системы, электронные учебники.

### **Преимущества использования Электронной Энциклопедии в обучении.**

Сопровождение учебного процесса техническими и вычислительными средствами обеспечивает достижение двух стратегических целей. Первая из них заключается в повышении эффективности всех видов образовательной деятельности на основе использования информационных и телекоммуникационных технологий. Вторая - в повышении качества подготовки специалистов с новым типом мышления, соответствующим требованиям информационного общества.

Достижение этих целей невозможно без применения специально разработанных компьютерных аппаратных и программных средств. К числу таких средств в полной мере относятся электронные средства обучения.

С помощью этих методов и средств будущий специалист должен научиться получать ответы на вопросы о том, какие имеются информационные ресурсы, где они находятся, как можно получить к ним доступ и как их можно использовать в целях повышения эффективности своей профессиональной деятельности.

Процесс информатизации образования повышает уровень активности обучаемого, развивает способности альтернативного мышления, формирует умение разрабатывать стратегию поиска решений как учебных, так и практических задач, позволяет прогнозировать результаты реализации принятых решений на основе моделирования изучаемых свойств объектов, явлений, процессов и взаимосвязей между ними.

**Организация учебного процесса.** Основной задачей разработанной подсистемы является автоматизация процесса формирования учебно-методических комплексов дисциплин для различных ВУЗов. Кроме того, уделяется внимание обеспечению удобства редактирования учебных планов, внесения в них как количественных изменений (связанных с изменением количества часов, выделенных на дисциплину), так и качественных изменений (связанных с добавлением или удалением учебных дисциплин), а также обеспечение доступа к просмотру учебных планов широкому числу пользователей.

Разработанная подсистема состоит из нескольких взаимосвязанных частей, каждая из которых даёт возможность пользователю увеличить скорость управления учебным процессом. Подсистема позволяет просмотреть понятийный граф как отдельной дисциплины (на котором отображены все взаимосвязи между разделами, частями и статьями), так и всех дисциплин, включенных в Электронную Энциклопедию (такой граф отражает взаимосвязь курсов и позволяет планировать порядок изучения дисциплин, формируя при этом семестровые расписания); просматривать список материалов по каждому курсу в отдельности или по всему методическому сопровождению Электронной Энциклопедии в удобной для восприятия древовидной структуре; управлять отдельными разделами любого предмета с одновременным просмотром его местоположения в понятийном графе.

Рассматриваемая подсистема способна самостоятельно классифицировать изучаемые дисциплины по степени их взаимосвязи, что позволяет расставить акценты и выделить приоритетные научные направления при написании дипломных и курсовых работ. Также она позволяет легко и своевременно включать и исключать из учебных планов курсы, получивших или, соответственно, утративших свою практическую и академическую ценность. Дополнительным положительным аспектом подсистемы является то, что обучающийся учится определять важность всего информационного пространства изучаемых дисциплин и расставлять их по степени приоритетности; более того, теперь он может потратить на получение необходимых ему знаний такое количество времени, которое он сам



сочтет нужным и самостоятельно сможет рассчитать, а также восполнить пробелы в знаниях по любой интересующей его дисциплине.

**Пример использования подсистемы организации учебного процесса.** Все размещенные в Электронной Энциклопедии материалы после редакторских правок и проверок проходят процедуру категоризации – внесения статьи в одно из информационных подпространств. В Электронной Энциклопедии допускается любое число категорий. Отдельные категории представлены для каждой учебной дисциплины, в свою очередь, каждый раздел конкретного предмета тоже организован в виде категории. Таким образом, информационное пространство ресурса является строго структурированным (характеризуется некоторым базисом). Это позволяет при обращении к отдельному материалу быстро найти все логически связанные с ним разделы и подразделы.

Строгая структура каждой представленной в Электронной Энциклопедии дисциплины вкупе с наличием понятийного графа категорий и позволяет эффективно организовывать и планировать учебный процесс.

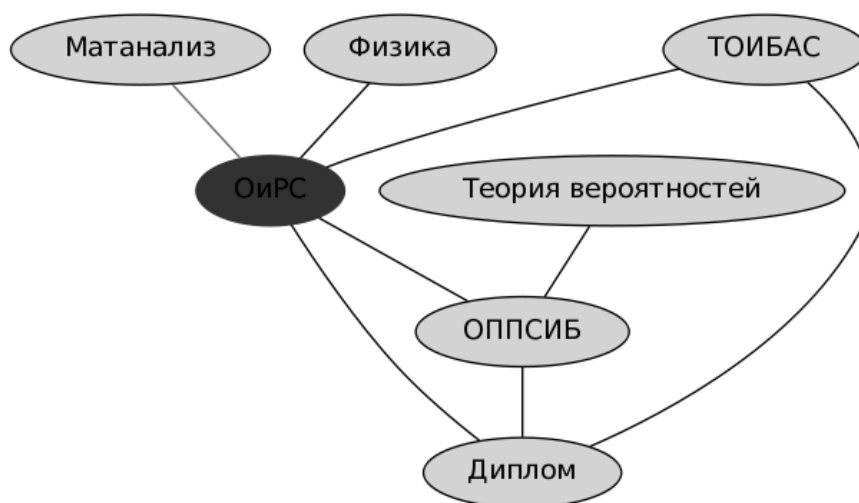


Рисунок 1 - Пример графа учебных дисциплин

Любая вершина представленного графа кодирует одну из учебных дисциплин. В свою очередь, для каждого представленного курса его содержание также имеет вид подграфа, что устанавливает взаимосвязи между разделами дисциплины.

Электронная Энциклопедия обеспечивает возможность сопровождать любую учебную программу. Поэтому материал может быть разделен на обязательный к освоению и дополнительный (дидактический). По обязательному к изучению материалу, представленному в Электронной Энциклопедии, могут быть сделаны оценки временных интервалов, требуемые для его освоения. Расчет временных интервалов для каждой темы дает в сумме время, необходимое студентам на изучение дисциплины.

**Подсистема защиты материалов Электронной Энциклопедии.** Разработаны две концепции по защите содержимого Электронной Энциклопедии: во-первых, защита от несанкционированного доступа (НСД), а во-вторых, защита от копирования материалов Электронной Энциклопедии. Защита от НСД начинается с зачисления студента (школьника) в образовательное учреждение, где для него создается учётная запись, которая присваивается ему на всё время обучения. Получить учётную запись можно только у работника, имеющего соответствующие полномочия, после предоставления документов, подтверждающих личность студента. Пароли для каждого студента сгенерированы специальным образом, из всего буквенно-цифрового пространства латинского алфавита с использованием специальных символов. Вслед за приобретением личной учётной записи студент (школьник) получает доступ к Электронной Энциклопедии из любой точки выхода в сеть Интернет. Для каждой учётной записи ведётся сбор статистики по количеству посещений в час, день, месяц; по IP - адресу пользователя, по посещаемым страницам Электронной Энциклопедии. Кроме

того, присутствует нетривиальный набор мер, предотвращающих возможные способы порчи содержимого. Так, пользователи Электронной Энциклопедии объединены в группы, каждой из которых предоставлен определенный набор прав для работы с Электронной Энциклопедией. Указанный метод организации разграничения доступа удобен тем, что он позволяет облегчить администрирование большого числа клиентов.

По результатам полученной статистики подсистема защиты проводит анализ и устанавливает некоторый набор правил, позволяющий решить следующие задачи:

- предотвращение неавторизованного доступа,
- ограничение доступа к содержимому Электронной Энциклопедии, её служебным ресурсам и программному коду,
- запрет полного или частичного копирования содержимого Электронной Энциклопедии стандартными средствами,
- предотвращение сохранения злоумышленником страниц Электронной Энциклопедии на жёсткий диск или съемный носитель информации,
- предотвращение копирования злоумышленником текста Электронной Энциклопедии при получении исходного кода страницы.

### **Заключение**

Электронная Энциклопедия является перспективной разработкой, направленной на повышение качества образования. Проект стимулирует познавательную деятельность студентов, снабжает их специфическими знаниями во многих областях науки и техники, помогает определиться со сферой интересов и будущей деятельностью. Также он является площадкой для различных дискуссий на научные и образовательные темы, тем местом, где возможен конструктивный диалог между студентами и методистами кафедры.

Электронная Энциклопедия находится в стадии постоянного обновления и добавления материала. Вследствие ежедневного чтения материала множеством студентов большая часть неточностей, ошибок и прочих логических противоречий своевременно выявляется и устраняется коллективом редакторов совместно с преподавателями. Это придает Электронной Энциклопедии оперативность в обновлении материала, которой так сильно не хватает печатным изданиям.

Как известно, планирование учебного процесса является одной из важнейших задач, решаемых неоднократно каждым учебным заведением. Использование Электронной Энциклопедии дает образовательному учреждению возможность унифицировать, автоматизировать и модернизировать формирование учебного процесса. Также она способствует повышению уровня методического контроля.

Из всего вышесказанного логически следует первостепенная задача сохранения работоспособности и безопасности Электронной Энциклопедии. Об эффективности и новизне внедренных, в целом, решений позволяет говорить разработанная подсистема защиты от несанкционированного доступа. Как и средства, направленные на предотвращение несанкционированного использования информации (в большинстве случаев защищенной авторскими правами), подсистема позволяет эффективно предоставлять доступ большому числу пользователей, при этом обеспечивая достаточную степень противодействия нарушениям.

### **Литература**

1. Бурмистрова Е.В. Проектирование учебного процесса (с использованием дистанционных технологий) (в вузе): Дис. ... канд. пед. наук: 13.00.08: Омск, 2005, 271 с.
2. Скляр Д.В. Искусство защиты и взлома информации. СПб.: БХВ-Петербург, 2004. 288 с.

# МЕТОДЫ РЕШЕНИЯ ЗАДАЧИ ОПТИМАЛЬНОГО ПРОЕКТИРОВАНИЯ

С.М. Гоменюк

*Научный руководитель д. ф.-м. н., профессор А.П. Карпенко*

МГТУ имени Н.Э.Баумана, Москва, Россия

## METHODS OF SOLVING THE PROBLEM OF OPTIMAL DESIGN

S. Gomenyuk

*Supervisors A. Karpenko*

Bauman Moscow State Technical University

### Аннотация

Рассматривается постановка задачи оптимального проектирования. Приводятся основные современные подходы к решению задачи непрерывно-дискретного программирования, а так же рассматриваются некоторые методы решения задачи оптимального проектирования.

### Abstract

This article considers the problem statement for the optimal design. The paper includes main modern approaches to solving problem of nonlinear discrete programming, as well as considers some methods for solving the problem of optimal design.

**Постановка задачи.** Среди прочих задач оптимизации выделяется класс задач оптимального проектирования. Для оптимального проектирования характерно сочетание, как непрерывных варьируемых параметров, так и дискретных, т.к. линейные размеры материалов определяются их сортаментом, и, значит, могут принимать только некоторые фиксированные дискретные значения. Поэтому при оптимальном проектировании возникает задача непрерывно-дискретной глобальной оптимизации.

Пусть задана целевая функция  $\Phi(X)$ , где  $X = (x_1, \dots, x_n)$ , при этом параметры  $x_j \in D_j = \{d_{j,1}, \dots, d_{j,m}\}$ ,  $j = [1:k]$  – дискретные, а  $x_i \in R$ ,  $i = [(k+1):n]$  – непрерывные. Для определенности будем считать, что все элементы множеств  $D_j$  упорядочены по возрастанию. Кроме того, пусть на компоненты вектора  $X$  заданы следующие ограничения, которые определяют множество допустимых значений  $D$ :

$$g_p(X) \leq 0, \quad p = [1:N_g], \quad (1)$$

$$h_t(X) = 0, \quad t = [1:N_h], \quad (2)$$

$$x_i \in [x_i^-, x_i^+], \quad i = [k+1:n], \quad (3)$$

Требуется найти минимум критерия оптимальности  $\Phi(X)$ , удовлетворяющий ограничениям (1) – (3):

$$\min_{X \in D} \Phi(X) = \Phi(X^*) = \Phi^* \quad (4)$$

Если все множества  $D_j$  содержат только целые числа, то задача (1) – (4) называется задачей непрерывно-целочисленного программирования. Если лишь некоторые из множеств  $D_j$  содержат только целые числа, то задача (1) – (4) называется задачей непрерывно-дискретно-целочисленного программирования. Постановка задачи (1) – (4) является самой общей. Некоторые виды ограничений могут отсутствовать. Кроме того, множества дискретных параметров  $D_j$  могут совпадать, в этом случае задача несколько упрощается.

Можно выделить следующие основные подходы к решению поставленной задачи: методы случайного поиска на дискретных множествах, методы штрафов, методы округления и переборные методы. Рассмотрим, как наиболее перспективные, первые две группы методов.

**Методы случайного поиска на дискретных множествах.** Чаще всего методы этой группы стоят на основе эволюционных алгоритмов таких, как генетический алгоритм, алгоритм дифференциальной эволюции и т.д. Методы этой группы организуют случайный поиск на множествах, как дискретных, так и непрерывных параметров.

Рассмотрим в качестве примера модификацию метода дифференциальной эволюции, предложенную для решения задачи непрерывно-дискретного программирования [1]. Метод относится к классу эволюционных алгоритмов.

Положим, что в задаче (1) – (4) ограничения (2) отсутствуют. В векторе варьируемых параметров выделим непрерывные ( $X^C$ ), целочисленные ( $X^I$ ) и дискретные ( $X^D$ ) компоненты, т.е. примем, что  $X = (x_1, \dots, x_n) = (X^C, X^I, X^D)$ . Классический метод дифференциальной эволюции разработан для решения задачи непрерывного программирования. Для решения задачи непрерывно-дискретно-целочисленного программирования предлагается ввести замену переменных.

Пронумеруем все элементы дискретных множеств  $D_j$  и сопоставим каждому элементу его индекс. Таким образом, все дискретные значения заменяются целочисленными. Далее произведем замену переменных. Ограничения типа (1) будем учитывать с помощью штрафных функций. Тогда целевая функция  $\Phi(X)$  заменится функцией  $f(X)$ , учитывающей штрафы.

Пусть теперь все элементы вектора  $X$  – непрерывные переменные, тогда при вычислении функции  $f$  будем считать, что  $f(X) = f(Y)$ , где элементы вектора  $Y$  определяются формулой

$$y_i = \begin{cases} \lceil x_i \rceil, & \text{если } i \in [0 : k] \\ x_i & \text{если } i \in [(k+1) : n] \end{cases}$$

Здесь  $\lceil x_i \rceil$  – целая часть  $x_i$

Как и все эволюционные алгоритмы, алгоритм дифференциальной эволюции работает не с единичным решением задачи оптимизации, а с целым набором решений, который называется популяцией. Популяция  $P^m$  поколения  $m$  содержит  $p$  векторов – решений задачи оптимизации. Каждый вектор  $X_i \in P^m$  называется особью популяции, и потенциально может быть решением задачи. Все элементы вектора  $X_i$  называются хромосомами особи.

Для начала работы алгоритма необходимо задать начальную популяцию  $P^0 = (X_1^0, X_2^0, \dots, X_p^0)$ . Как правило, о положении глобального минимума априори ничего неизвестно, поэтому в качестве начальной популяции выбирают случайно сгенерированные вектора, удовлетворяющие ограничениям задачи оптимизации:

$$X_i^0 = \begin{cases} \{x_{i,j}^0 \mid x_{i,j}^0 = r_{i,j}(x_j^- - x_j^+) + x_j^-, i = [1 : p], j = [1 : k]\}, \\ \{x_{i,j}^0 \mid x_{i,j}^0 = r_{i,j}(x_j^- - x_j^+ + 1) + x_j^-, i = [1 : p], j = [(k+1) : n]\}, \end{cases} \quad (5)$$

где  $r_{i,j}$  – величина, равномерно распределенная из отрезка  $[0, 1]$ .

Каждая итерация алгоритма – это генерация нового поколения, которая происходит в два этапа.

На первом этапе генерируется так называемая временная популяция  $P_i^{m+1} = (X'_1, X'_2, \dots, X'_p)$  по схеме:

$$x'_{i,j} = \begin{cases} x_{i,j}^m + F(x_{q,j}^m - x_{t,j}^m), & \text{если } r_{i,j} \leq C_r \text{ или } j = s_i, \\ x_{i,j}^m, & \text{иначе,} \end{cases} \quad (6)$$

здесь  $C_r \in [0, 1]$ ,  $F \in [0, 2]$ ,  $r \in [0, 1]$ ,  $l, q$  и  $t$  – случайные индексы, указывающие на три различные особи популяции. Для каждой  $i$ -ой особи эти значения должны вычисляться

вновь. Случайный индекс  $s_i$  определяет номер хромосомы  $i$ -ой особи и всегда обеспечивает изменение хотя бы одной хромосомы. Величины  $F$  и  $s_i$  являются параметрами алгоритма и остаются постоянными на протяжении поиска минимума, так же, как и параметр  $p$ .

Второй этап определяется схемой

$$X_i^{m+1} = \begin{cases} X'_i, & \text{если } f(X'_i) \leq f(X_i^m), \\ X_i^m, & \text{иначе.} \end{cases}$$

При вычислении  $x'_{i,j}$  по формуле (6) ограничения (3) могут нарушаться, тогда для генерации значения  $x'_{i,j}$  используется формула (5). Таким образом, происходит случайная замена хромосом каждой особи и в следующее поколение попадают только те особи, которые наилучшим образом удовлетворяют целевой функции.

**Методы штрафов.** Метод штрафов не является самодостаточным, но используется в комбинации с любым методом локальной или глобальной оптимизации. Метод заключается в том, что минимизируется не целевая, а оценочная функция.

Целевая функция дополняется штрафами, и становится функцией, которая называется оценочной функцией. Оценочная и целевая функции совпадают в точках дискретного множества. По мере удаления от дискретных значений, оценочная функция возрастает. В работе [2] рассматривается метод штрафов в комбинации с методом роя частиц.

Не вдаваясь в подробности метода роя частиц, рассмотрим задачу непрерывно-дискретного программирования (1) – (4). Предлагается следующая схема ее решения. Преобразуем все дискретные переменные в непрерывные и заменим целевую функцию  $\Phi(X)$  функцией вида

$$F(x) = \Phi(x) + s\phi(x) + r \sum_{i=1}^{N_g} \max[0, g_i(x)], \quad (7)$$

где

$$\phi(x) = \sum_{j=1}^k \frac{1}{2} \left[ \sin \frac{2\pi \{x_j^c - 0.25(d_{j,l+1} + 3d_{j,l})\}}{d_{j,l+1} - d_{j,l}} + 1 \right], \quad (8)$$

При введении штрафов (8) на дискретные переменные, задача (1) – (4) становится задачей нелинейного программирования, к которой применим метод роя частиц. Следует отметить, что функция (7) сильно нелинейная и многоэкстремальная. Поэтому очень важно найти глобальный минимум этой функции.

Рассмотрим пример применения функции (8) в качестве штрафной функции. Пусть целевая функция имеет вид

$$\Phi(x) = x^4 - \frac{8}{3}x^3 - 2x^2 + 8x,$$

и пусть на переменную  $x$  наложены ограничения  $x \in \{-1, 0, 1, 2\}$ . Тогда функция (7) для приведенного примера будет выглядеть так, как показано на рисунке 1.

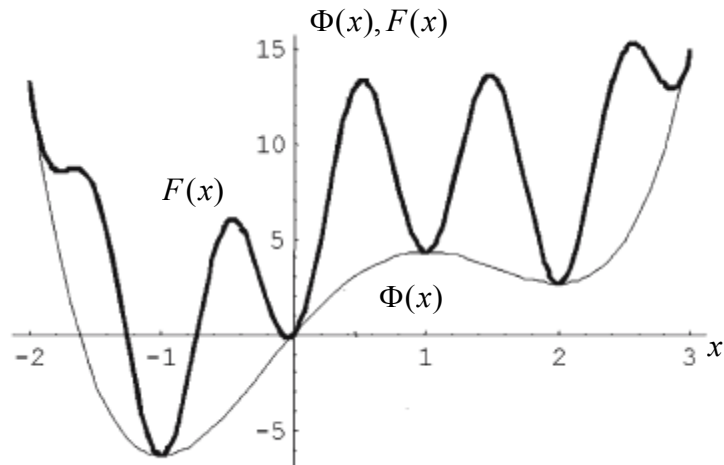


Рисунок 1 - Целевая и оценочная функции

Таким образом, задача (1) – (4) сводится к задаче нелинейной глобальной оптимизации.

**Другие методы.** Рассмотренные группы методов не единственные. Например, в работе [3] приводится подход к решению задачи непрерывно-дискретного программирования, основанный на округлении варьируемых дискретных параметров. Это - группа методов, в которой также происходит переход к задаче нелинейного программирования. На каждом шаге алгоритма находят оптимальные значения варьируемых параметров. Далее, по некоторой стратегии, определяется, какие параметры считать оптимальными на данном шаге. Эти параметры фиксируются и не изменяются при дальнейшем поиске минимума. Когда все дискретные переменные зафиксированы минимум считается найденным.

Для решения задачи непрерывно-дискретного программирования возможно также использование некоторых модификаций метода ветвей и границ. В этом случае перебор дискретных значений осуществляется методом ветвей и границ, а поиск оптимальных значений непрерывных переменных производится каким-либо методом нелинейного программирования.

### Литература

1. J. Lampinen, I. Zelinka. Mixed Integer-Discrete-Continuous Optimization by Differential Evolution, Part 1: the optimization method. In: Ošmera, Pavel (ed.) (1999). Proceedings of MENDEL'99, 5th International Mendel Conference on Soft Computing, June 9–12. 1999, Brno, Czech Republic. Brno University of Technology, Faculty of Mechanical Engineering, Institute of Automation and Computer Science, Brno (Czech Republic), pp. 71–76.
2. S. Kitayama, M. Arakawa, K. Yamazaki. Penalty function approach for the mixed discrete nonlinear problem by particle swarm optimization, Struct Multidisc Optim (2006) 32, pp. 191–202.
3. A.A. Groenwold, N. Stander, J.A. Snyman. A pseudo-discrete rounding method for structural optimization. Structural Optimization 11, Springer Verlag 1996, pp. 218-227.

# ГЕНЕТИЧЕСКОЕ ПРОГРАММИРОВАНИЕ В РАМАНОВСКОЙ СПЕКТРОСКОПИИ

**Ивченко М.О.**

*Научный руководитель: канд. техн. наук, доцент, Власов А.И.*

МГТУ им. Н.Э. Баумана, Москва, Россия

## GENETIC PROGRAMMING IN RAMAN SPECTROSCOPY

**Ivchenko M.O.**

*Ph.D., associate professor, Vlasov A.I.*

MSTU n.a. N.E. Bauman, Moscow, Russia

### **Аннотация**

В данной работе представлены описание процесса и результаты моделирования метода, основанного на принципах генетического программирования и предназначенного для идентификации состава вещества по его спектру рамановского рассеяния. Эффективность разработанного метода при проведенном моделировании составляет 76,4%.

### **Annotation**

Presented in this paper is a description and results of the modeling of a method based on the principles of genetic programming and intended for identification of a substance's components from its Raman scattering spectrum. The efficiency of this method as obtained from said modeling is 76,4%.

Рамановская спектроскопия – неупругое рассеяние оптического излучения на молекулах вещества, как твёрдого, жидкого, так и газообразного, сопровождающееся заметным изменением его частоты. В спектре рассеянного излучения появляются спектральные линии, число и расположение которых определяется молекулярным строением вещества. Одно из важнейших применений рамановской спектроскопии заключается в определении молекулярного состава образца по его спектру рамановского рассеяния.

Распознаванию спектров рамановского рассеяния мешает ряд проблем. Присутствие флуоресцентных соединений, примесей и сложных смесей создают трудности при идентификации соединений по их спектрам. Кроме того, самые интенсивные пики в спектрах компонентов вещества могут приходиться на одну область в спектре исследуемого вещества. Ранее автором в [1] был описан метод генетического программирования (ГП), в котором были учтены указанные проблемы. Вместо того, чтобы иметь целью лишь эволюцию правил, корректно классифицирующих обучающие выборки данных, в методе был также оптимизирован выбор правил с тем, чтобы минимизировать вероятность ошибочной классификации будущих выборок (была использована двухуровневая функция приспособленности).

Каждая особь генетического алгоритма (ГА) представляет собой математическую формулу, составленную из функций и переменных. К функциям относятся простые математические операторы «+» и «-», а переменные соответствуют частотам спектра. Популяция инициализируется случайными сочетаниями функций и переменных для создания бинарных деревьев ограниченной глубины (в данной работе она равна 5).

В совокупности функции и частоты, выбранные особью  $i$ , формируют формулу  $E_i$ , вычисление значения которой по соответствующим интенсивностям для заданного спектра  $S_i$  дает значение  $E_i(S_i)$ . Это значение в свою очередь интерпретируется как индикатор присутствия ( $E_i(S_i) > 0$ ) либо отсутствия ( $E_i(S_i) < 0$ ) соответствующего компонента в веществе.

В данной работе описан процесс и представлены результаты моделирования данного метода в среде MATLAB.

## Моделирование

При генерации пиков в моделируемых базовых спектрах на основании изучения спектров рамановского рассеяния было принято приближение, что уровень интенсивности в пике можно смоделировать случайной величиной, подчиняющейся распределению Коши:

$$f_p(x; x_0; \gamma) = \frac{1}{\pi\gamma \left[1 + \left(\frac{x-x_0}{\gamma}\right)^2\right]} = \frac{1}{\pi} \left[ \frac{\gamma}{(x-x_0)^2 + \gamma^2} \right], \quad (1)$$

где  $x_0 \in \mathbb{R}$  — параметр сдвига, а  $\gamma > 0$  — параметр масштаба.

В соответствии с вышесказанным, базовые спектры  $I_i^B$  моделируемых веществ были заданы следующим образом:

$$\begin{aligned} I_1^B(x) &= f_p(x; 2400; 1,3 \cdot 10^{-4}) + A; \\ I_2^B(x) &= f_p(x; 2415; 2,61 \cdot 10^{-4}) + A; \\ I_3^B(x) &= f_p(x; 1800; 2 \cdot 10^{-4}) + A; \\ I_4^B(x) &= f_p(x; 1800; 2 \cdot 10^{-4}) + f_p(x; 2850; 4 \cdot 10^{-4}) + A; \\ I_5^B(x) &= f_p(x; 600; 1 \cdot 10^{-3}) + f_p(x; 1200; 5 \cdot 10^{-4}) + f_p(x; 1800; 4 \cdot 10^{-4}) + A, \end{aligned} \quad (2)$$

где  $A$  – белый шум, добавленный для моделирования естественной зашумленности реальных спектров рамановского рассеяния.

На основании описанных выше формул (2) были сгенерированы спектры базовой выборки  $B = \{I_j^B \mid j = \overline{1,5}\}$ , по характеристикам схожие со спектрами рамановского рассеяния, которые использованы в дальнейшем как набор базовых спектров (необходимые для построения обучающей и тестовой выборок). Данные спектры представлены на рисунке 1 с отображением частот пиков в каждом.

Изучение особенностей спектров рамановского рассеяния показало, что во многих случаях рамановский спектр образца, содержащего компоненты в разных концентрациях, представляет собой линейную комбинацию рамановских спектров входящих в него веществ. С учетом вышесказанного, спектр  $I_j^E$  обучающей выборки задается следующим образом:

$$I_j^E(w) = \sum_{i=1}^B c_{ji}^E \cdot I_i^B(\omega), \quad (3)$$

где  $I_i^B(\omega)$  –  $i$ -й базовый спектр, а  $c_{ji}^E$  – его концентрация в  $j$ -м обучающем спектре  $I_j^E(w)$ .

Значения концентраций  $c_{ji}^E$  при этом должны удовлетворять следующей системе уравнений:

$$\begin{cases} c_{ji}^E = n \cdot C_S^E, \\ c_{ji}^E \in [0,1], \\ \sum_{i=1}^B c_{ji}^E = 1, \end{cases} \quad (4)$$

где  $C_S^E$  – шаг изменения концентрации для обучающих спектров,  $n \in \mathbb{N}$ .

Обучающую выборку  $E$  в таком случае можно задать как  $E = \{I_j^E \mid j = \overline{1, N_E}\}$ , где  $N_E$  – количество спектров в обучающей выборке, зависящее от шага изменений концентрации  $C_S^E$ . В данной работе  $C_S^E = 0,25$ , а  $N_E = 70$ .

Аналогичным образом была сгенерирована тестовая выборка  $T = \{I_j^T \mid j = \overline{1, N_T}\}$ , где  $N_T$  – количество спектров в тестовой выборке, зависящее от шага изменений концентрации  $C_S^T$ . В данной работе  $C_S^T = 0,2$ , а  $N_T = 126$ .

Для каждого базового спектра создается популяция из 2 000 особей путем задания случайных сочетаний функций и переменных для создания бинарных деревьев. После инициализации популяции происходит вычисление приспособленности каждой ее особи. Самые приспособленные особи отбираются для скрещивания, в результате чего формируется следующее поколение популяции.



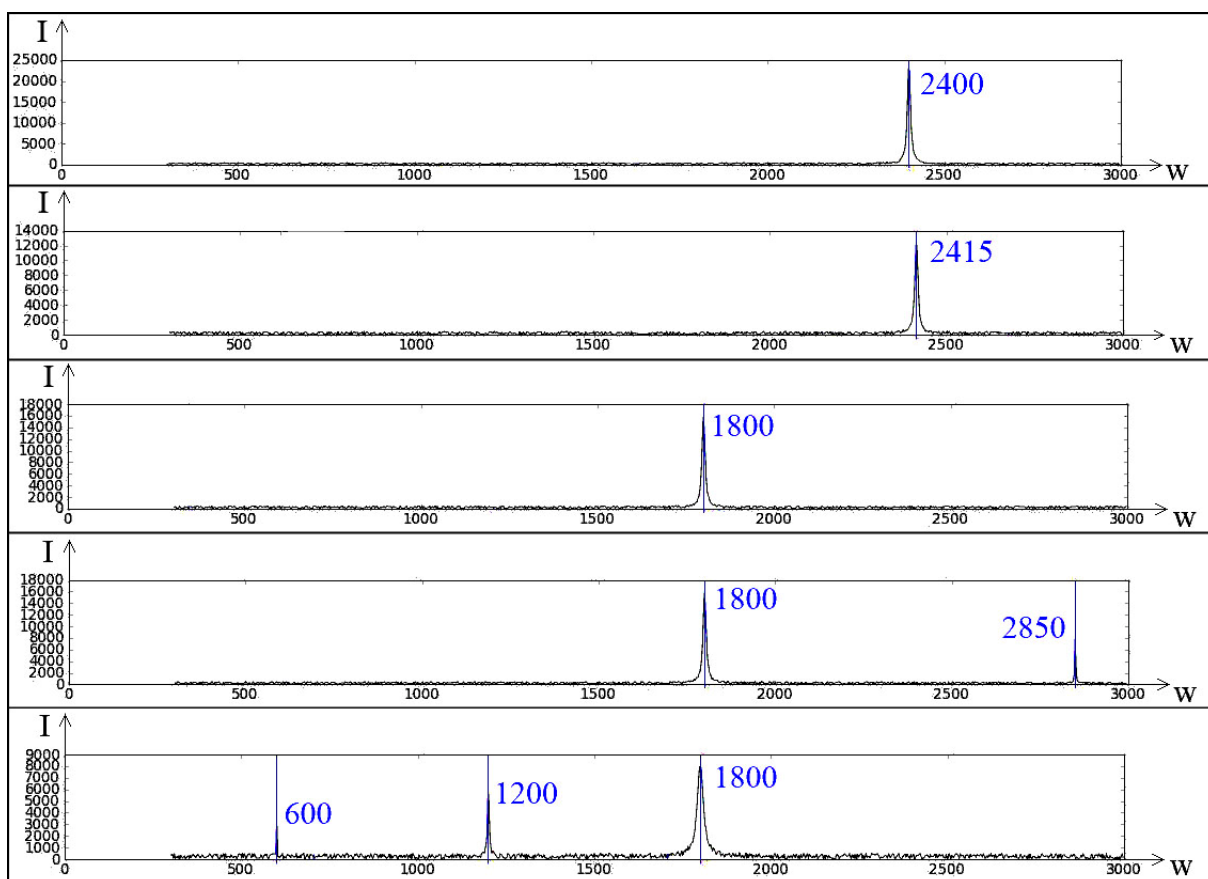


Рисунок 1 – Набор базовых спектров  $B$  для моделирования спектров виртуальных веществ обучающей выборки  $E$  и тестовой выборки  $T$

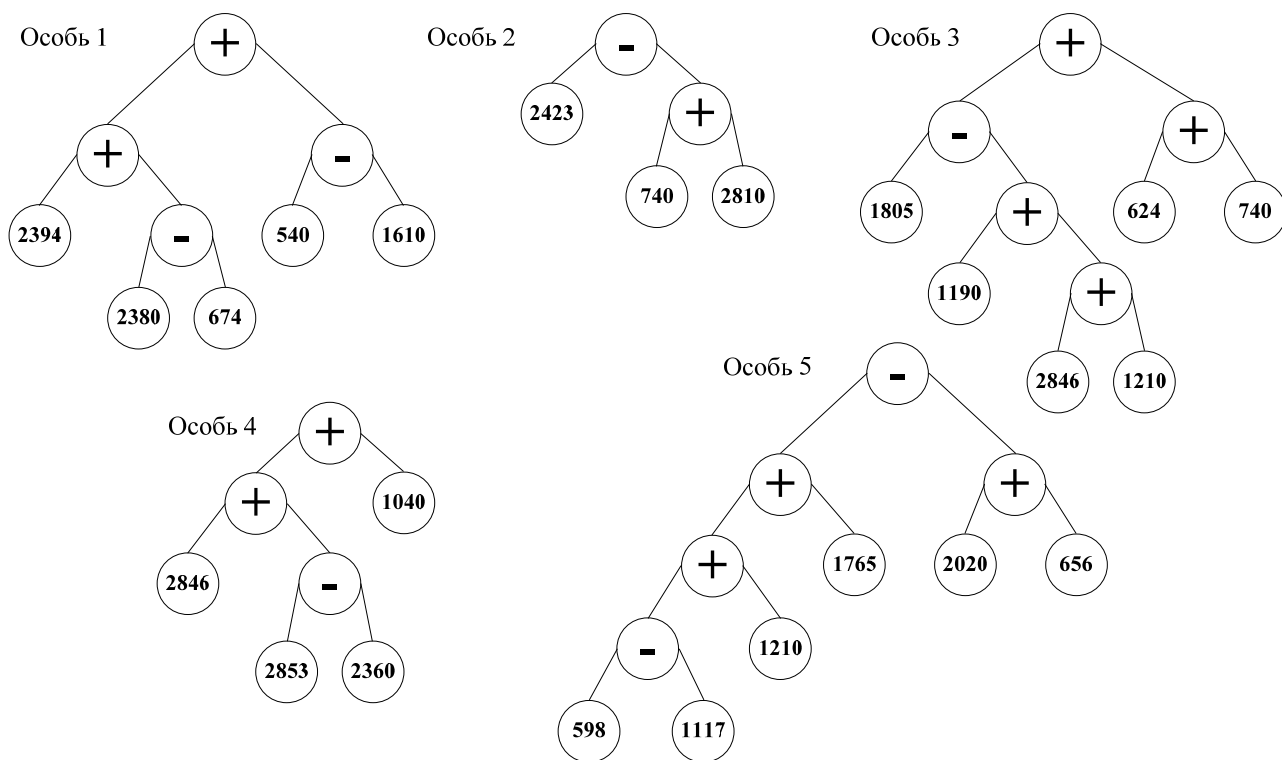


Рисунок 2 – Полученные особи (программы) для идентификации 5 моделируемых компонентов виртуальных веществ (базовых спектров)

Параллельно со скрещиванием в соответствии со стратегией элитизма происходит копирование 1,5% самых приспособленных особей из каждой популяции в следующее поколение без изменений. Следом наступает мутация, заключающаяся в появлении случайных изменений в особи новой популяции и имеющая фиксированную вероятность 20% [2].

Наконец, осуществляется проверка на наступление состояния схождения. В случае если это состояние не наступило, происходит создание нового поколения. В случае же наступления состояния схождения осуществляется выход из цикла и вывод лучшей особи оператору.

### Результаты моделирования

Особи, полученные при моделировании для идентификации пяти базовых моделируемых компонентов, представлены на рисунке 2.

В ходе эволюции были получены формулы, в которых частоты, близкие к пикам в базовых спектрах, присутствуют как операнды операции арифметического сложения. Остальные частоты служат для исключения ложной идентификации в присутствии других моделируемых компонентов.

При тестировании особей оценивается число верных и ложных попаданий при идентификации спектров тестовой выборки (моделируемых веществ). При этом ведется отдельный учет ложноположительных  $W_p$  и ложноотрицательных  $W_n$  срабатываний, так как в некоторых ситуациях один вид ошибки является более критичным, чем другой (как то определение наличия взрывчатого вещества в исследуемом материале). Верные положительные и отрицательные результаты обозначены как  $R_p$  и  $R_n$  соответственно.

Таблица 1 – Оценка эффективности разработанного метода

Базовый спектр	Верные срабатывания, % ( $R_p + R_n$ )	Ложные срабатывания, % ( $W_p + W_n$ )
$B_1$	78 (36 + 42)	22 (12 + 10)
$B_2$	80 (42 + 38)	20 (12 + 8)
$B_3$	74 (39 + 35)	26 (14 + 12)
$B_4$	74 (34 + 40)	26 (16 + 10)
$B_5$	76 (44 + 32)	24 (13 + 11)
Средн. арифм.	<b>76,4</b>	<b>23,6</b>

Как видно из представленных данных, с помощью разработанного метода в рамках проведенного моделирования достигнута эффективность идентификации базовых веществ в размере 76,4%.

## **Выводы**

Для повышения эффективности идентификации компонентов веществ по их спектрам рамановского рассеяния с использованием представленного метода генетического программирования требуется решение задачи нахождения оптимальных параметров для моделирования, так как изменение этих параметров влияет на эффективность метода.

Также в дальнейшем представленный метод предлагается совершенствовать с тем, чтобы определять химическую концентрацию каждого компонента вещества по его спектру рамановского рассеяния.

Вместе с тем, полезным представляется совершенствование функции приспособленности с целью ранжированного учета ложноположительных и ложноотрицательных срабатываний, так как в некоторых случаях один вид ошибки является более критичным, чем другой.

## **Литература**

1. Ивченко М.О. Нейросетевая обработка данных рамановской спектроскопии. // Научные технологии и интеллектуальные системы 2010 – М.: МГТУ им. Н.Э.Баумана. 2010. С. 168-171.
2. Панченко Т.В. Генетические алгоритмы: учебно-методическое пособие. – Астрахань: Издательский дом «Астраханский университет», 2007. – 87 с.

# МНОГОКРИТЕРИАЛЬНАЯ ОПТИМИЗАЦИЯ НА ОСНОВЕ АППРОКСИМАЦИИ ФУНКЦИИ ПРЕДПОЧТЕНИЯ ЛИЦА, ПРИНИМАЮЩЕГО РЕШЕНИЯ

Моор Д.А., Мухлисуллина Д.Т.  
*д.ф.-м. н., профессор Карпенко А.П.*  
МГТУ им. Н.Э.Баумана, Москва, Российская Федерация

## MULTI-CRITERIA OPTIMIZATION BASED ON APPROXIMATION OF THE DECISION MAKER'S UTILITY FUNCTION

Moor D.A., Mukhlisullina D. T.  
*Professor, Karpenko A.P.*  
Bauman Moscow State Technical University, Moscow, Russian Federation

### Аннотация

Рассматривается прямой адаптивный метод многокритериальной оптимизации на основе аппроксимации функции предпочтения лица, принимающего решение, с помощью нейронных сетей, аппарата нечеткой логики и нейро-нечеткой системы. Исследуется эффективность метода при решении 2-х и 3-х критериальных тестовых задач с использованием различных сверток критериев оптимальности.

### Abstract

The paper considers the direct adaptive method for multi-criteria optimization based on approximation of a decision maker preferences function by using neural networks, fuzzy and neuro-fuzzy systems. Research results of the method effectiveness for solving 2- and 3-criterion test problems with different criteria convolutions are presented.

**Постановка задачи.** Рассматривается следующая задача многокритериальной оптимизации

$$\min \Phi(X) = \Phi(X^*), X \in D_X, \quad (1)$$

где  $X$  – вектор варьируемых параметров задачи;  $D_X$  – ограниченное и замкнутое множество допустимых значений вектора  $X$ ;  $\Phi(X)$  – векторный критерий оптимальности  $\Phi(X) = (\phi_1(X), \dots, \phi_m(X))$ . В этой задаче лицо, принимающее решение (ЛПР), стремится найти такой вектор  $X^*$  – искомое решение задачи многокритериальной оптимизации (МКО-задачи) – которое минимизирует на множестве  $D_X$  каждый из частных критериев оптимальности. Предполагается, что частные критерии оптимальности тем или иным способом нормализованы.

Предлагаемый метод решения МКО-задачи (1) основан на том предположении, что на множестве  $D_X$  существует функция предпочтений ЛПР  $\psi(X)$  (пси), которая отображает это множество во множество действительных значений, т.е.  $\psi: X \rightarrow R^1$ . Более строго, МКО-система преобразует введенные ЛПР значения лингвистической переменной  $\psi(X)$  в действительные числа, принадлежащие диапазону [1..9]. В результате МКО-задача сводится к задаче поиска вектора  $X$ , который максимизирует функцию предпочтений ЛПР:

$$\max \psi(X) = \psi(X^*) = \psi^*.$$

Обозначим  $\varphi(X, \Lambda)$  операцию свертки частных критериев оптимальности, где  $\Lambda \in D_\Lambda \subset R^m$  – вектор весовых множителей;  $D_\Lambda = \{\lambda_i | \lambda_i \geq 0, \sum_i \lambda_i = 1, i \in [1:m]\}$  – множество допустимых значений этого вектора. Для любого  $\Lambda \in D_\Lambda$  метод скалярной свертки сводит решение задачи (1) к решению однокритериальной задачи глобальной условной оптимизации (ОКО-задачи)

$$\min \varphi(X, \Lambda) = \varphi(X^*, \Lambda), X \in D_X. \quad (2)$$

Условие (2) позволяет полагать [1], что функция предпочтений ЛПР определена не на множестве  $D_X$ , а на множестве  $D_\Lambda$ , т.е.  $\psi: \Lambda \rightarrow R^1$ . В результате МКО-задача сводится к задаче поиска вектора  $\Lambda^* \in D_\Lambda$  такого, что

$$\max \psi(\Lambda) = \psi(\Lambda^*) = \psi^* . \quad (3)$$

**Метод решения задачи.** Общая схема рассматриваемого прямого адаптивного метода решения МКО-задачи является итерационной и состоит из перечисленных ниже следующих основных этапов [2].

*Этап «разгона» метода.* МКО-система некоторым образом (например, случайно) последовательно генерирует  $k$  векторов  $\Lambda_1, \dots, \Lambda_k$  и для каждого из этих векторов  $\Lambda_i$  выполняет следующие действия. 1) Решает соответствующую ОКО-задачу (2). 2) Предъявляет ЛПР найденное решение  $X_i^*$ , а также соответствующие значения всех частных критериев оптимальности  $\Phi(X_i^*)$ . 3) ЛПР оценивает эти данные и вводит в МКО-систему соответствующее значение своей функции предпочтений  $\psi(\Lambda_i)$ .

*Первый этап.* На основе всех имеющихся в МКО-системе значений векторов  $\Lambda_1, \dots, \Lambda_k$  и соответствующих оценок функции предпочтений  $\psi(\Lambda_1), \dots, \psi(\Lambda_k)$ , МКО-система выполняет следующие действия. 1) Строит функцию  $\bar{\psi}_1(\Lambda)$ , аппроксимирующую функцию  $\psi(\Lambda)$  в окрестности точек  $\Lambda_1, \dots, \Lambda_k$ . 2) Отыскивает максимум функции  $\bar{\psi}_1(\Lambda)$  – решает ОКО-задачу (3). 3) С найденным вектором  $\Lambda_1^*$  решает ОКО-задачу вида (2) – находит вектор параметров и соответствующие значения частных критериев оптимальности, а затем предъявляет их ЛПР. ЛПР оценивает указанные данные и вводит в систему соответствующее значение своей функции предпочтений  $\psi(\Lambda_1^*)$ .

*Второй этап.* На основе всех имеющихся в МКО-системе значений векторов  $\Lambda_1, \dots, \Lambda_k, \Lambda_1^*$  и соответствующих оценок функции предпочтений  $\psi(\Lambda_1), \dots, \psi(\Lambda_k), \psi(\Lambda_1^*)$  МКО-система строит функцию  $\bar{\psi}_2(\Lambda)$ , аппроксимирующую функцию  $\psi(\Lambda)$  в окрестности точек  $\Lambda_1, \dots, \Lambda_k, \Lambda_1^*$  и т.д. по схеме первого этапа до тех пор, пока ЛПР не примет решение о прекращении вычислений.

**Алгоритм и программная реализация метода.** Поскольку на компоненты вектора весовых множителей  $\Lambda$  наложено ограничение  $\sum_{i=1}^m \lambda_i = 1$ , один из весовых множителей (пусть

это будет множитель  $\lambda_m$ ) можно выразить через остальные:  $\lambda_m = 1 - \sum_{i=1}^{m-1} \lambda_i$ . Вследствие этого,

в качестве входов нейронных сетей, нечеткой и нейро-нечеткой систем рассматриваются только компоненты  $\lambda_1, \dots, \lambda_{m-1}$  вектора  $\Lambda$ . Выходной переменной является лингвистическая переменная  $\psi$ , ядро которой  $\psi$  принимает значения  $1, 2, \dots, 9$  [1].

В работе рассматривается нейросетевая аппроксимация функции с использованием перцептронной нейронной сети (MLP-сеть) с 3-я, 5-ю, 7-ю и 9-ю нейронами в скрытом слое и нейронной сети на основе радиальных базисных функций (RBF-сеть) [2].

Построение нечеткой модели происходит в два этапа. На *первом этапе* осуществляется формирование базы знаний и грубая настройка модели, которая включает в себя назначение весов  $\delta$  нечетких правил ЕСЛИ-ТО (ЕСЛИ <значения входных переменных> ТО <значение выходной переменной> ( $\delta$ )) и форм функций принадлежности:  $a$  - полуширина функций принадлежности входных переменных,  $b$  - полуширина функций принадлежности выходных переменных. Суть *второго этапа* состоит в тонкой настройке модели, которая заключается в

подборе таких весов нечетких правил ЕСЛИ-ТО и таких параметров функций принадлежности, которые минимизируют различие между экспериментальным и желаемым поведением объекта:  $E(a,b,\delta) = \sqrt{\sum_{i=1}^N (\psi(\Lambda_i) - \bar{\psi}(\Lambda_i))^2} \rightarrow \min_{a,b,\delta}$ , где  $N$  – номер итерации [3].

В работе используется схема нечёткого логического вывода Мамдани, которая состоит из пяти основных этапов:

Фазификация → Агрегирование → Активизация → Аккумуляция → Дефазификация.

Структура нейро-нечеткой системы *ANFIS* (*Adaptive Neuro-Fuzzy Inference System*), реализующей нечеткий вывод Сугено представлен на рис. 1 [4].

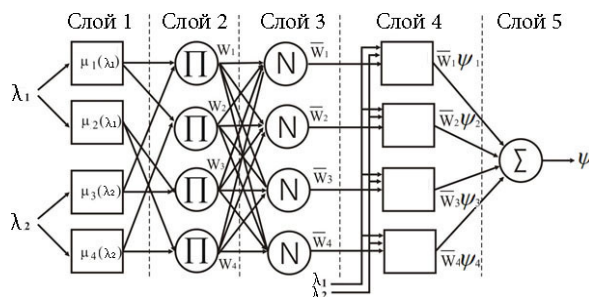


Рисунок 1 - Структура нейро-нечеткой сети *ANFIS*.

**Исследование эффективности метода.** Исследование выполнено для двух двумерных двухкритериальных задач и одной трехмерной трехкритериальной тестовых МКО-задач.

Двухкритериальная задача 1, имеющая выпуклый фронт Парето,

$$\phi_1(X) = (x_1 - 5)^2 + x_2^2 + 10; \quad \phi_2(X) = x_1^2 + (x_2 - 5)^2 + 20; \quad D_x = \{X \mid 0 \leq x_i \leq 5, i = 1,2\}.$$

Двухкритериальная задача 2 (невыпуклый многосвязный фронт Парето)

$$\phi_1(X) = -x_1; \quad \phi_2(X) = -f(X) \left( 1 - \sqrt{\frac{x_1}{f(X)}} - \frac{x_1}{f(X)} \sin(10\pi x_1) \right);$$

$$f(X) = 1 + 9x_2; \quad D_x = \{X \mid 0 \leq x_i \leq 1, i = 1,2\}.$$

Трехкритериальная задача 3 (выпуклый фронт Парето)

$$\phi_1(X) = (x_1 - 5)^2 + x_2^2 + x_3^2 + 10; \quad \phi_2(X) = x_1^2 + (x_2 - 5)^2 + x_3^2 + 20;$$

$$\phi_3(X) = x_1^2 + x_2^2 + (x_3 - 5)^2 + 5; \quad D_x = \{X \mid 0 \leq x_i \leq 5, i = 1,2,3\}.$$

Погрешность аппроксимации функций предпочтений определялась по формуле

$$\varepsilon_a = \frac{1}{N} \sum_{i=1}^N \left| \psi(\Lambda_i) - \bar{\psi}(\Lambda_i) \right|, \quad \text{где } N \text{ – номер итерации. Аппроксимирующие функции}$$

представлены на рис. 2 и рис. 3.

**Выводы.** Результаты исследования показывают, что ошибка аппроксимации  $\varepsilon_a$  во всех случаях находилась в пределах от 0.3 до 0.7. Исключением являются *RBF*-сети, при использовании которых ошибка достигала значения  $\sim 200$  (что связано с плохими экстраполяционными свойствами этих сетей). Минимальная погрешность аппроксимации при решении всех тестовых задач достигается при использовании нейро-нечеткой системы *ANFIS*. Погрешность аппроксимации и число итераций зависят от числа нейронов в скрытом слое *MLP*-сети, решаемой МКО-задачи и начальных значений компонентов разгонных векторов  $\Lambda$ .

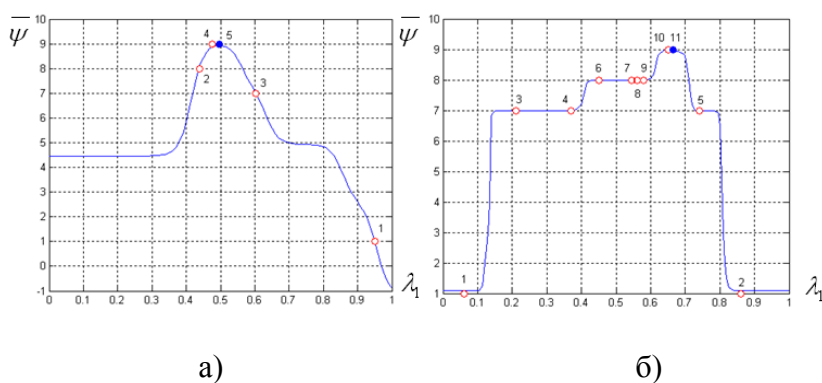


Рисунок 2 - Аппроксимирующие функции предпочтения ЛПР:  
 а) МКО-задача 1; *MLP*-сеть с 5-ю нейронами; 5-я итерация;  
 б) МКО-задача 2; Нечеткая логика; 11-я итерация

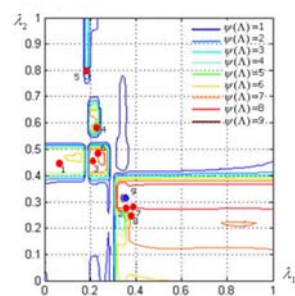


Рисунок 3 - Линии уровня функции, аппроксимирующей функцию предпочтений ЛПР: МКО-задача 3; *ANFIS*; 9-я итерация

По сравнению с нейросетевой и нейро-нечеткой аппроксимацией функции предпочтений ЛПР, решение задачи с помощью нечеткой логики требует большего времени вычислений. Так, время обучения нейронной сети во всех экспериментах не превышало в среднем двух секунд, в то время как выполнение тонкой настройки нечеткой модели в среднем составило 20 секунд. Этот факт объясняется тем, что вычисление значений аппроксимирующих функций требует большего числа вычислений на пути от Фаззификации до Дефаззификации. При этом с каждой следующей итерацией вычислительная сложность указанного пути повышается, поскольку увеличивается количество входных правил нечеткой системы.

## Литература

1. А.П. Карпенко, В.Г. Федорук. Один класс прямых адаптивных методов многокритериальной оптимизации // Информационные технологии. - 2009. - №5. - с.24-30.
2. А.П.Карпенко, Д.Т.Мухлисуллина, В.А. Овчинников. Нейросетевая аппроксимация функции предпочтений лица, принимающего решения, в задаче многокритериальной оптимизации // Информационные технологии, 2010, №10, с.2-9.
3. А.П. Карпенко, Д.А. Моор, Д.Т. Мухлисуллина. Многокритериальная оптимизация на основе нечеткой аппроксимации функции предпочтений лица, принимающего решения // Наука и образование: электронное научное издание. Инженерное образование, 2010, №1 (<http://technomag.edu.ru/doc/135375.html>).
4. А.П. Карпенко, Д.А. Моор, Д.Т. Мухлисуллина. Многокритериальная оптимизация на основе нейро-нечеткой аппроксимации функции предпочтений лица, принимающего решения. // Наука и образование: электронное научное издание. Инженерное образование, 2010, №6 (<http://technomag.edu.ru/doc/143964.html>).

# ИСПОЛЬЗОВАНИЕ БИЗНЕС-ОРИЕНТИРОВАННЫХ МОДЕЛЕЙ ДЛЯ ИТ-ИНФРАСТРУКТУРЫ

Плехов С. Ю.

МГТУ им.Н.Э.Баумана, Москва

## USE OF BUSINESS-ORIENTED MODELS FOR IT INFRASTRUCTURE

Plekhov S. Y.

MSTU named after Bauman, Moscow

### Аннотация

В статье изложены основные концепции подхода к моделированию бизнес процессов, связь между моделированием бизнес процессов и концепцией сервис ориентированной архитектуры, рассмотрен конкретный пример использования с использованием описанных методов.

### Abstract

The article describes the basic concepts of the approach to modeling business processes, communication between the modeling of business processes and the concept of service-oriented architecture, is considered a concrete example of using the methods described.

Роль ИТ - инфраструктуры практически любого предприятия играет все большую роль с каждым годом, однако пропасть между конечными бизнес-процессами компании и их технической реализацией по-прежнему огромна.

Одним из популярных решений в последнее время стало использование моделирования при анализе или разработке бизнес процессов .

Существует довольно обширный выбор необходимого инструментария. Все ведущие поставщики программных комплексов масштаба предприятия готовы предложить обширный выбор необходимого ПО.

Жизненный цикл управления бизнес-процессами (Business Process Management - BPM) на базе сервис - ориентированной архитектуры (Service Oriented Architecture - SOA) представляет собой непрерывный процесс взаимодействия Модель, Компонировка, Развертывание, Управление, Модель. Разработка модели включает в себя 4 уровня. Уровень бизнес процессов, сервисный, уровень приложения и технологический уровень. Этот цикл подразумевает корректировку ситуации в реальном времени и совершенствование процессов.

Нотация и инструментарий, которые используются при моделировании бизнес-процессов, стандартизированы. Существует два распространенных стандарта: нотация The Business Process Modeling Notation (BPMN), описывающая стандарт моделирования и графическое представление шагов в бизнес процессе и Business Process Modeling Notation (BPEL), стандартизированный язык, на основе XML используемый для определения бизнес-процессов.

Также существует возможность импорта моделей, написанных с использованием других нотаций. К примеру, при использовании IBM WebSphere Business Modeler Advanced существует возможность импорта моделей описанных в Microsoft Visio, моделей, описанных с использованием IDEF0 стандарта, SDL диаграмм и т.д.

Можно заметить, что данный подход с использованием управления бизнес-процессами (BPM) похож на концепцию сервис ориентированной архитектуры (SOA), но все же есть несколько концептуальных отличий. SOA – концепт, основанный на разработке многоуровневых, стандартизированных процессов, позволяющий преодолеть разрыв между процессами предприятия и ИТ-базой с использованием шаблонов и стандартизированных методов. SOA представляет сервисы, в то время как BPM является потребителем сервисов.

Однако при моделировании не всегда детализация процессов опускается до технологического уровня, т.к. это сильно усложняет конечную модель и большинство



характеристик, таких как стоимость и доступность попросту включаются в компоненты на уровне приложения. Как следствие, технологический уровень остается непрозрачным и слабоуправляемым.

Приведем простой пример. Рассмотрим небольшую компанию, в которой используется система ERP (Enterprise Resource Planning System — Система планирования ресурсов предприятия) для которой стало доступно обновление, позволяющее существенно улучшить комфортность работы конечных пользователей, однако перед переходом на новую версию, необходимо убедиться, что такой переход не отразится на функциональных модулях, необходимых для корректной работы системы. Следовательно, необходимо провести предварительное тестирование.

Для проведения тестирования, обычно создается полная копия продуктивной системы в изолированном окружении, где проводятся все необходимые тесты, однако перед началом работ необходимо провести ряд согласований. Убедиться, что в распоряжении есть достаточное количество ресурсов, провести резервное копирование, назначить ответственных специалистов, квалификация которых позволяет проводить работы.

Распространенной практикой является назначение ответственных специалистов, которые сами должны провести все необходимые согласования. Таким образом, человеческий фактор выходит на первое место, что, потенциально, является источником проблем, т.к. задача может быть назначена на некомпетентного специалиста, либо не будут проведены все необходимые предварительные работы, либо сделаны не все согласования, что в свою очередь может отказать негативное воздействие на эффективность работы всего предприятия. Если мы вернемся к нашему примеру, при создании копии существует вероятность того, что проведение работ совпадет с периодом высокой нагрузки, и как следствие проведения таких работ, будет снижение скорости расчетов, что не позволит конечным пользователям продуктивного ландшафта получить необходимые отчеты в срок.

Специалист, ответственный за непосредственное выполнение технической части не в силах знать все особенности ландшафта. Информация о критичных промежутках времени должна учитываться смежными отделами, способными прогнозировать нагрузку на системы. Таким образом, для того чтобы не допустить падения производительности во время проведения критичных вычислений, необходимо перед началом проведения работ проводить согласование со специалистами, владеющими такой информацией.

Данную проблему обычно решают прописыванием всех необходимых согласований в регламенте работы предприятия, но регулирование процессов с помощью регламентов кажется недостаточно эффективными. Объем таких документов может достигать нескольких сот страниц, а в случае когда в данном документе регулярно вносятся изменения, донести их до ИТ персонала становится непростой задачей. Довольно часто, к таким документам относятся как к простой формальности. Таким образом любой, даже отлично составленный и согласованный документ регламентирующий работу ИТ персонала становится бесполезным. Но при правильном подходе к моделированию процессов, все необходимые регламенты можно реализовать программно.

Если мы вернемся к нашему примеру с тестированием обновлений ERP системы, при корректном и детальном моделировании, все необходимые согласования проводятся автоматически, попадая исполнителю со всей необходимой информацией, такой как дата и время проведения работ, технологическое окно и пр. В случае если какой то из пунктов, как например, резервное копирование не будет выполнен, то исполнитель не получит запроса на выполнение и не сможет начать работы. Для того, чтобы специалист не мог по своей инициативе начать работы, необходимые учетные записи на управляющем ПО заблокированы, либо для данных учетных записей доступен только режим чтения. Разблокирование проводится только в рамках технологического окна.

Если пойти еще дальше, то такую систему можно еще более усовершенствовать, разработав правила, необходимые для согласования. Так загрузку систем можно прогнозировать, основываясь на исторических данных, полученных с управляющего ПО, либо средств мониторинга, таким образом, полностью исключив из процесса человеческий

фактор, оставив только функции аудита и нотификации по принятым решениям и внесенным изменениям. Кроме того, такой подход гармонично сочетается с идеями распределенных облачных вычислений

Существует довольно обширный портфель решений, который позволяет реализовать описанное решение. IBM может предложить целое семейство продуктов Rational, предназначенных для разработки анализа и оптимизации, а для управления, мониторинга и автоматизации семейство продуктов Tivoli.

Безусловно, для реализации такого подхода может потребоваться еще и множество предварительных шагов, к тому же довольно затратных. Но положительные стороны кажутся довольно убедительными. После описания всех процессов, наглядная графическая иллюстрация может подсказать, где находятся узкие места. Для более сложных случаев можно воспользоваться оптимизаторами, что в конечном итоге, может принести существенные улучшения - повысить надежность работы систем, улучшить прозрачность и существенно снизить стоимость информационной структуры предприятия, что в конечном итоге должно окупить все издержки по внедрению.

### Литература

1. Luc Chamberland, John Colgrave, Dermot Flaherty, Greg Flurry, Laura Gardash, Michael Hudson, Ben Mann, Anthony O'Dowd, Robert Phippen, Piotr Przybylski, Marc-Thomas Schmidt, Gari Singh IBM Business Process Management Reviewer's Guide REDP-4433-02
2. John Bergland, Luc Maquil, Kiet Nguyen, Chunmo Son BPM Solution Implementation Guide REDP-4543-00
3. Jim Hagemann Snabe, Ann Rosenberg, Mark von Rosing Applying Real-World BPM in an SAP Environment 978-1-59229-343-8
4. Jim Hagemann Snabe, Ann Rosenberg, Charles Møller, Mark Scavillo Business Process Management: The SAP Roadmap 978-1-59229-231-8
5. Lucas Jellema Oracle BPM Suite 11g Handbook 9780071754491
6. Judith Hurwitz, Robin Bloor, Marcia Kaufman, Fern Halper Service Oriented Architecture (SOA) For Dummies, 2nd Edition 978-0-470-52549-4

## MEASURING REDUNDANCY LEVEL ON THE WEB

Alexander Afanasyev, Chunyi Peng, and Jiangzhe Wang  
Computer Science Department, University of California, Los Angeles  
{afanasev, chunyi, lucas}@cs.ucla.edu

## ИЗМЕРЕНИЕ УРОВНЯ ИЗБЫТОЧНОСТИ В WEB

Афанасьев Александр, Чуньи Пенг и Джянджи Ванг  
Факультет Информатики, Университет Калифорнии, Лос Анджелес

### Abstract

In this project we estimate redundancy level on the Web employing existent search engines as a source for our measurements. We performed our measurement using a three step process. First, we collected a large set (100K) of randomly chosen pages from the Internet catalogs (DMOZ and Delicious). Second, for each page in the set we extracted a 32-word random phrase that may uniquely identify the content on this page. Finally, we used exact phrase queries (queries in quotes) to the three of most popular search engines (Google, Yahoo, and Bing) to discover the estimate of the number of pages satisfying our query. The obtained results show that even such small randomly chosen phrases are able to uniquely identify page content in a large number of cases (about 40% for all search engines). Also, we discovered that in our sampling set most of the pages (75% for Google and 90% for Yahoo and Bing) have less than 10 similar pages. Thus, though page redundancy is an important issue in some cases (news, mailing lists, etc.), it does not have yet a global widespread across all types of webpages.

### 1 Introduction

Simplicity of information distribution in electronic form created a number of problems in various areas. One example of such problems is battle for copyrights on the Internet. Another not less important problem is increasing complexity of relevant information retrieval for search engines. In particular, if essentially the same information appears on thousands of websites, any relevant query will yield all these pages no matter what search engine we use; and it will be very hard for a human to discover other related information on the same topic. For example, if a same mailing list question is reposted on numerous websites, but only one website contains the answer to this question, it is virtually impossible to find this answer. Due to the increasing commercialization of the Web (i.e., various models of ad-based revenue services) more people and automated systems are interested in cloning the information from one website to another. Thus, the replication phenomenon can potentially become more and more ubiquitous and nobody yet knows the degree of its widespread globally.

In this project we are trying to answer the question whether replication (redundancy) is a real problem across all types of Web pages or not. To get this answer we first randomly select a set of webpages (based on DMOZ <http://www.dmoz.org> and Delicious <http://www.delicious.com> catalogs). And second, for each page in the set we discover how many pages on the Web replicate a portion of the content of this page. As long as existent search engines already indexed a large portion of the Internet and they became an integral part of our daily life, we decided to use them as the tools to discover a number of replicas. This decision limited us to a very small portion of page content that we can use for comparison: search engines do not answer queries that are more than 32 words long. However, even with such short queries, we can use exact phrase matches (i.e., search for phrases in quotes). In this case, if a phrase is properly identified, we should be able to uniquely identify a particular page (more exactly, a content on this page) in the index of a search engine. Section 2 provides detailed information about implementation aspects of our measurements.

Our initial expectation was that a considerable portion of webpages in our sampling set will be uniquely identified. At the same time, we believed that another big portion of the pages will have a moderate-to-large redundancy level. In other words, we expected some form of a bi-modal distribution (Figure 1), meaning that a significant portion of the Web content is reduplicated by semi-automated or automated systems (bots) to attract visitors from search engines. However, as we show in Section 3 this expectation is not entirely true.

In our experiments we used three different search engines (Google, Yahoo, and Bing) and compared the obtained results from all of them. As we show in Section 3, this yielded in a very interesting observation about relation between Yahoo and Bing search engines. Though it was announced recently that Yahoo search is now powered by Bing [10], our results confirm this claim by showing a very high correlation between Yahoo and Bing, unlike the results obtained from Google.

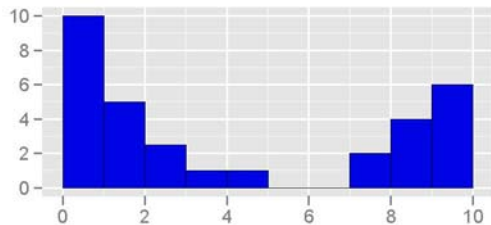


Figure 1: Expected distribution of page redundancies

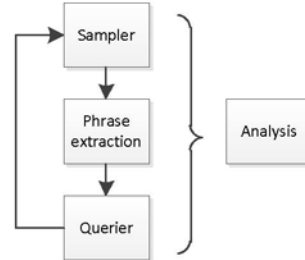


Figure 2: Main components of redundancy measurements

In addition to different search engines, we tried to separate our sampling sets in several different categories: recreation, sports, home, health, computer, news, food, games, research, culture. However, we explicitly excluded very redundancy-prone categories, like news and mailing lists, as we expected they would be clear outliers in our measurements.

## 2 Methodology and Experiment Design

To perform our experimental evaluation we implemented three basic components: page sampler, phrase extraction, and querier (Figure 2). For software components we chose Java language, because it has a reach set of open source libraries, such as HTML parsing library [11], library for morphological analysis [1], and many others. In the rest of this section we present in detail each of the implemented components.

### 2.1 The sampler

The sampler is one of the crucial components in our project. All redundancy measurement are performed based on the data provided by the sampler. Thus, the quality of the sampler will define largely the overall quality and representativeness of our analyses.

On a high level, the design of the sampler is practically identical to a common search crawler. The sampler obtains a list of initial start pages, downloads content of the pages, extracts links, adds extracted list to a download queue, and recursively repeats download-add steps until a predetermined condition (e.g., a limit of the total number of downloaded pages) is reached. However, there are two important differences between functionality of the crawler and the sampler. First of all, the sampler does not simply add all of the extracted links to the download queue (excluding various search engine policies and exclusions in robots.txt files), but uses a random algorithm to choose a link for further processing. Second of all, the sampler has important restrictions on types of the start pages. For example, if the task is to get a sample of links related to news, <http://www.cnn.com/> will not be a good starting point, because it will generally provide links to articles on the same website. As a result, no matter which random algorithm we choose, we will not receive a representative sampling set of all available news articles on the Web. One of the best starting points in the news example is a news aggregator. If it provides a lot of links to a wide range of news websites and news articles, by randomly choosing a portion of them, we will get a set of high-diversity links.

We based our implementation of the sampler on a simple Java search crawler implementation [13]. As a backend of the crawler we chose PostgreSQL database [12], which is extremely powerful and open-source object-relational database system with more than 15 years of active development. We chose the simplest, but yet powerful random algorithm to select links for the sampling set:

1. Select the first link from the download queue and shift the queue
2. Download page content for this link
3. Extract all links from the page and add them to the download queue

4. For each extracted link we throw a dice: If the random value is less than a predefined threshold, we add the link to the sampling set
5. If the number of elements in sampling set is less than a predefined threshold and there are links in download queue, start from step #1

We created several sampling sets based on different topics of our choice. To approximate topic separation we performed sampling of DMOZ <http://www.dmoz.org> on-line catalog starting from pages correspondent to different categories. The advantage of using such a catalog is two-fold. First, this catalog by definition contains a diversity of links. Second, the links in the catalog are moderated, which limits the number of spam sites in our sampling set. We configured our sampler to pick 2% of the discovered links, which provide good enough approximation of pure random links in particular category from DMOZ.

Another good source of the high-diversity links is the on-line bookmark service Delicious <http://www.delicious.com/>. This service provides users with an ability to save their book-marked links online, and at the same time gives all other users ability to browse all these links. Because users choice to make a bookmark can be considered as a random process, there is no need to perform any additional randomization when picking links while crawling. Though, we decide to introduce a small level of randomness by picking 80% of the discovered links.

Figure 3 illustrates the diversity of links in our sampling sets. The image show mappings of the top level domain names present in sampled URLs, separately for sampling sets obtained from DMOZ catalog and Delicious bookmark service. It can clearly be seen that in both cases we have a high variety of domain names (practically all country TLDs are present) and domain name frequencies reflect to some degree penetration of the Internet in the world [2] (assuming that .com domains are mostly used in the United States). This observation gives us hope that the sampling set we obtained has some level of the representativeness world-wide.

## 2.2 Phrase extraction

Another crucial component of our system is forming for each webpage a set of phrases that are able to identify (or partially identify) this page among all other in a search engine. For this purpose, we implemented PhraseExtractor component, which uses HTMLParser [11] library to extract all visualized textual information from the HTML.<sup>1</sup> However, in a raw form, this textual information is not very useful. Textual representation of navigation links, copyright notices, and other similar information is essential part of the HTML, but does not have any use for our redundancy measurement purposes. Running HTML-to-text conversion on various webpages allowed us to develop two simple heuristics that significantly improve quality of our phrases.

In the first heuristic, we eliminate all textual information that came from the separately visualized HTML tags (such as <div>, <p>, etc.) and have less than N words (in our experiment N = 10).

In the second heuristic we eliminate from textual representation of HTML all sentences that are less than N words long (similarly, we use N = 10 in our experiments). To extract sentences from the text we are relying on MorphAdorner Java library [1], which is open-source and implements “intelligent” algorithms for sentence splitting (e.g., guessing whether abbreviations “Mrs.”, “a.m.”, and similar end a sentence or not, handling sentences in quoted phrases, etc.). To make sentence splitting more predictable, we consider all separately visualized HTML tags as complete sentence(s) (i.e., we are adding period at the end of such tags, because usually periods are missing there). This heuristic allows more intelligent elimination of short (and thus not-identifying) phrases.

<sup>1</sup>Unfortunately, the library is limited only to HTML formatting and does not handle any CSS visual effects.

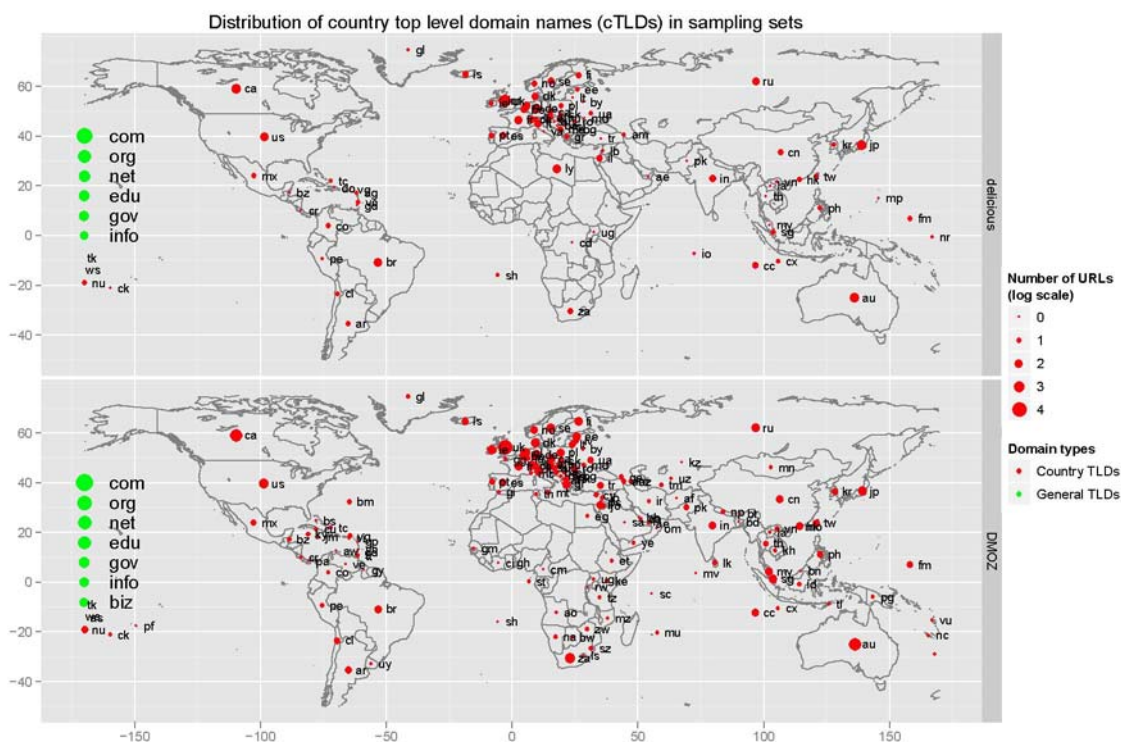


Figure 3: Coverage of country top level domain names (TLDs) in the sampling sets

Even after elimination of short phrases, textual representation of web pages still contains a lot of text. Because our objective is to use a search engine to identify set of pages with the same (or partially the same) content, we are able to use only a limited number of words in a query (e.g., Google allows searches on phrases that are up to 32 words). There are two ways to handle this issue: simple and complicated.

The simple way of solving this problem is to randomly select up to 32 consecutive words from the text and perform exact phrase search using only these words. (Note that if in the original text there were three consecutive sentences and our short-sentence elimination heuristic removed the second sentence, we select two phrases, separately the first and the third sentence.) The downside of a such simple solution is that 32 words (and even 32 consecutive words) may not be a good content identifier. For example, we could choose to search over copyright notice, which will produce enormous amounts of search results, but may not have any meaning for redundancy measurements (e.g., if our search domain is source code, copyright notice can be the same for a large number of different programs). Another potential problem can arise from different interpretations of special characters. Initial experiments revealed that Google ignores all periods, commas, question and exclamation marks, colons, semicolons, and brackets, but treats “&” in phrases as a word. This means, that single search on a random phrase from the text may not be enough to get meaningful results.

Another and more complicated way to use Google’s limited-word search and have more confidence in results, is to use several independent experiments using the same page. In other words, we can several times (e.g., 5 times) randomly extract a 32-word phrase and obtain several independent search results. The problem here is how interpret such separate results. Due to time limitation and restrictions imposed by search engines on number of queries (see Section 5), we have not used this heuristic. As we show in Section 3 even with the simple heuristic we still have a high confidence in our phrase extraction algorithm (due to the fact that most of the pages were identified by the search engines using our phrases).

### 2.3 Querier

After extracting phrases from webpages in the sampling set, we made the decision to get potential duplicate webpages by querying the phrases against a search engine. There are two reasons for this choice. First of all, it saves us lots of efforts and time compared with the alternative approach, i.e. building a large webpage index from scratch by ourselves. Secondly, it is much easier to carry out our

measurement result into a practical use in the future. More specifically, if the results showed that duplication level is extremely high for certain set of topics, we could share our result with commercial search engines and possibly contribute to better simplified searching result. On the other hand, if our measurements were performed on a set of webpages that are not indexed by any search engine, our measurement result would not be a practically valuable. Because of we choose publicly available links from DMOZ and Delicious catalogs, the links in our sampling sets are guaranteed to be included in the search indices.

We now illustrate the details of retrieving potential duplicate pages from Google, Yahoo, and Bing. Google used to provide a search API for use by third-party applications. However, for some reason they discontinued their previous generation API, but did not fully open a new generation API (AJAX API). For this reason we decided to use generally available HTML search interface for Google. To search for “Isaac Newton” against Google, the GET request URL would be [http://www.google.com/search?...&q="Isaac+Newton"&...](http://www.google.com/search?...&q=), where the ellipses represent other parameters. For example, we used additional parameter “filter=0” to prevent Google from filtering results that it considers to be related (as long as we are interested in a whole range of duplicates for a particular page).

After getting back a query result page, the task is to extract the estimated number of the results and target links. (A target link here means a URL that identifies a query result.) We use the aforementioned Java HTML parser library to identify both of the components using CSS selectors (DIV#resultStats and A.1 for Google). Also, it is possible to get a whole set (up to first 1000) of the links returned by Google using additional GET requests with specified “start” parameter. For instance, <http://www.google.com?...&start=20&...> returns query results between the 20th and the 29th. Unfortunately, search engines (and Google in particular) do not like being automatically queried and impose a variety of limitation and blocks (e.g., blocking computer’s IP if querying with high frequency, presentation of a reCaptcha code after a high volume of queries, etc.). Thus, it is virtually impossible to perform many queries and we abandoned the idea of extra queries per phrase.

For Yahoo, we adapted the querier implementation for Google, that uses different search URL and CSS selectors. For Bing, we registered our application and obtained Bing API key that allowed us to obtain results for the queries in XML format. Parameters for each search engine are summarized in Table 1.

To tackle with search engine limitation problem we deployed the querier on 20 different machines located in different networks. From each machine we performed 1 query in 10 seconds

Table 1: Search engine parameters

Engine, For- mat	Search URL	Estimate block <sup>□</sup>	Result block <sup>□</sup>
Google, HTML	<a href="http://www.google.com/search?q=">http://www.google.com/search?q=</a>	DIV#resultStats	A.1
Yahoo, HTML	<a href="http://search.yahoo.com/search?p=">http://search.yahoo.com/search?p=</a>	SPAN.count	A.yshintl
Bing, XML	<a href="http://api.search.live.net/xml.aspx?Appid=&lt;apikey&gt;&amp;sources=web&amp;query=">http://api.search.live.net/xml.aspx?Appid=&lt;apikey&gt;&amp;sources=web&amp;query=</a>	//WebResult/Url	//Total

<sup>□</sup> for HTML formats blocks are defined by CSS selector, for XML format blocks are defined by XPath query for each search engine. Even with such low-rate queries, Google and Yahoo were temporarily (for 24 hours) blocking computers from the search engine. On the bright side, Bing has not blocked us, meaning that it allows bigger limits either for general queries or for queries via Bing API. Totally, we were able to perform 100,000 queries for each search engine, analyses of which are presented in the next section.

### 3 Measurement results

In this section we present results of our analysis. First, we present quantitative characteristics of overall redundancy levels for each of the evaluated search engines (Google, Yahoo, and Bing). Then, we show detailed breakdown of redundancy level by different category.

In our analyses we eliminated clear outliers that resulted from random phrase extraction process. In particular, we invalidated all records in our database that correspond to empty result sets (due to our link discovery process, page has to be present in the search indices, see Section 2.1). Also, we excluded all high-frequency results (result set that larger than 1,000,000), because they are likely to have been generated by a very common or invalid phrase extracted from the page (see Section 5).

#### 3.1 Overall redundancy levels

Our first finding was that the most of the phrase queries to all search engines resulted in a very small number of search results. As it can be seen from Figure 4, over 86% for Google and over 97% for Yahoo and Bing queries yielded the number of results in a range from 1 to 60.

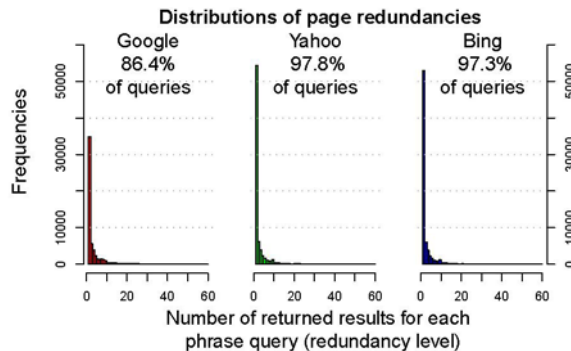


Figure 4: Distributions of page redundancies for Google, Yahoo, and Bing

More interestingly that a largest component on these graph is region correspondent to the queries resulted in exactly one search results. If we look to the same results in log-log scale (Figure 5), we can see that such behavior applies not only for the initial area (say, pages with redundancy level from 1 to 60), but also to any other region. In other words, distribution of page redundancies loosely follows a power law distribution. One observation that we are making from these log-log plots is that results from Google follow a power law distribution across all redundancy levels, while Yahoo and Bing show more sharp distribution for “small” redundancy levels (from 1 to  $10^2$ ), but after express an almost uniform distribution. This observation can have various explanations. For instance, it may indicate that Yahoo/Bing more effectively eliminate duplicate/similar pages from the result sets. Also it can happen because Google makes more optimistic predictions about the result set, while Yahoo and Bing employ a slightly more pessimistic strategy. Another plausible explanation is that Google is less restrictive to indexed pages, resulting in a bigger search index, and thus making more pages available during query resolving process. Such behavior could also be due to a different interpretation of phrase queries by search engines. We have not fully investigate this possibility, but in some cases search engines ignored our exact phrase searches and presented us with results for a keyword search. If this is the case, then the results for large redundancy levels could be eliminated. However, even with such elimination, our initial observation (that large number of queries resolving to a very small number of pages) still holds.



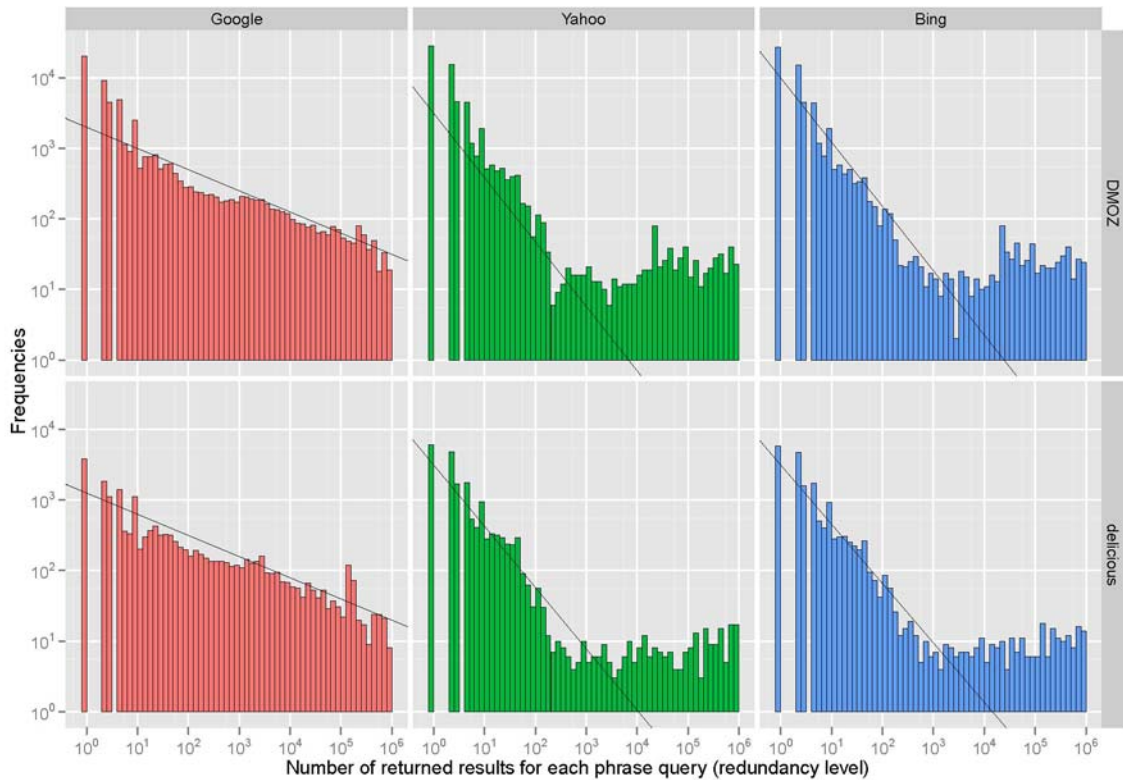


Figure 5: Distributions of page redundancies for Google, Yahoo, and Bing for each sampling set

To understand the exact behavior of redundancy distribution in the initial region (e.g., for redundancy levels from 1 to 10) we built a cumulative distribution function (CDF) of page redundancies in log-log scale (Figure 6). From this figure we can easily obtain percentage of queries (pages) that resolved in a defined number of the results (duplicate pages). On the graph we marked the most interesting region—the initial region with redundancy levels from one to ten. In addition to our previous observation (Figure 4) that most of the pages have no more than 60 redundant pages, now we see that the same can be concluded even for tighter region. According to CDF, Google in 75% and Yahoo and Bing in 90% cases gave us results set of size up to 10.

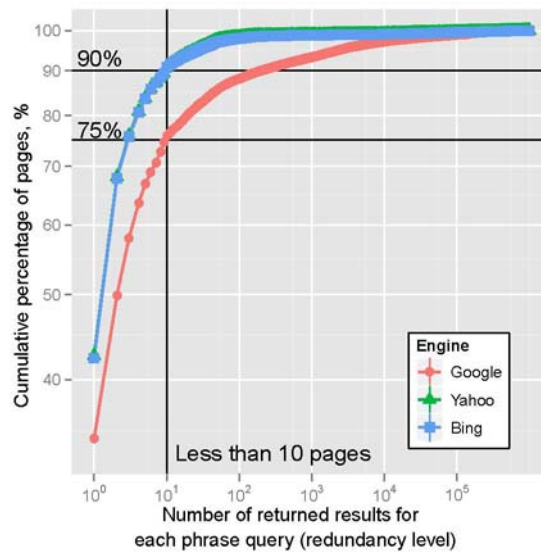


Figure 6: Cumulative distribution function (CDF) of page redundancies in log-log scale

We have already seen that the major redundancy component corresponds to pages without redundancy at all. Using CDF we can say that in our experimental evaluation about 40% pages belong to this category. Next biggest component, resulting total to 50% and 70% for Google and Yahoo/Bing,

corresponds to pages with pages with redundancy level up to two. In large number of cases such behavior can be explained by the fact that search engines consider pages from different domain names as different pages, but most of the time pages with and without “[www.](http://www.)” prefix point to the same content.

Manual analysis of several queries with redundancy level from 3 to 10, gave us some understanding of one of the potential sources for the redundancy. There are many cases when some important parts of the page content (e.g., detailed information about a company or company/person portfolio) are duplicated several times as within single site, as well as on various different sites. For example, Yahoo query ““Mirage Systems Inc. is very different than other container manufacturers” “At Mirage we do one thing and we do it very well” “We produce high quality harness / container”” resulted in 6 pages, 2 of them belong to the company Website [www.miragesys.com](http://www.miragesys.com) (the main and “about us” page present the same paragraph with short information about the company), 2 belong to people directory website [www.zoominfo.com](http://www.zoominfo.com), and rest 2 belong to other websites with the same company description. This example shows a very natural way people distribute information. However, it would be very useful to always discover initial source (and potentially distribution path) for the same piece of the information. Such feature may not be very interesting for small result sets like in the example above, but may greatly help to reorder output results based on the degree from the original source.

### 3.2 Redundancy levels for different topics

To understand the difference in redundancy levels across different topics, we build a collection of histograms in log-log scale for each pair of search engine and sample set category (Figure 7). As it can be seen, results for each category are consistent across all evaluated search engines relatively to other categories. Results from Bing/Yahoo almost duplicate each other, indicating a close relation between the search engines. This results confirm the announcement [10] that since August 2010 Bing powers Yahoo search engine. During our evaluations we were unaware about this announcement, and practically identical behavior of Yahoo and Bing was very suspicious. Though results from Yahoo and Bing are very close to each other, they still have a large variance. This may be a result of slight randomization (geographical or topological proximity, user preferences, etc.) while query processing. At the same time, this may indicate the fact that Yahoo and Bing are still two separate search engines with separate data centers, but use the same algorithms to crawl and process queries.

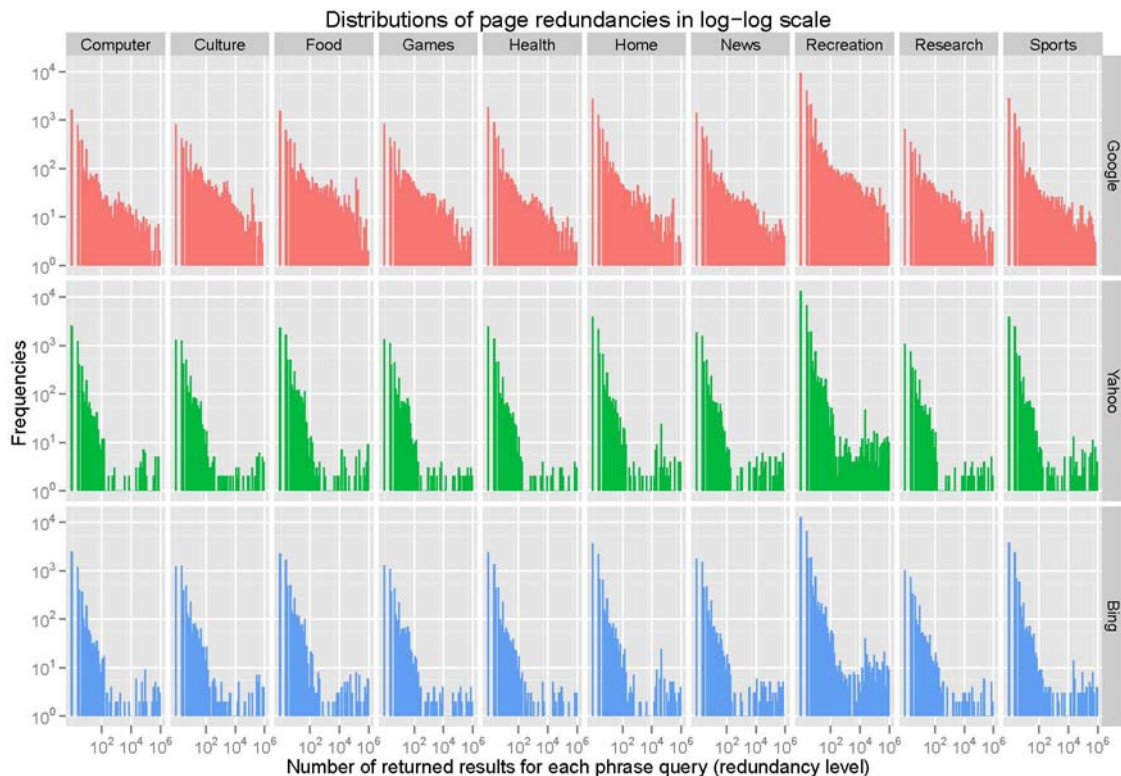


Figure 7: Distribution of page redundancies in log-log scale by engine and category

The redundancy levels for each category presented on Figure 7 are consistent to overall statistics for each search engine (Figure 5): there is almost ideal power law distribution in the results from Google, and sharper power law distribution in initial redundancy zone ( $1-10^2$ ) for Yahoo/Bing. One interesting observation that can be made from these graphs is that category “Recreation” stand out among all other categories. This category includes a variety of topics (audio, autos, aviation, birding, boating, bowling, camps, climbing, collecting, crafts, drugs, fireworks, fishing, food, gambling, games, gardening, and many others<sup>2</sup>) and we have not fully investigated which one of this subcategories may have influenced our results the most. Because we used Internet catalog structure to categorize links, we have considerable imperfections in such division. In particular, category “Sports” is included in category “Recreation”.

To see relations between different categories more clearly, we plot CDF of page redundancies by category in log-log scale. Due to the fact that results for each search engine have practically the same relative relations between categories, we present only one graph that corresponds to Google (Figure 8). This graph also shows that category “Recreation” has larger redundancy levels compared to all other categories. In addition, it also shows that second high-redundancy category is “Sport.” Another interesting observation is about the other end of the scale. The less redundant pages belong to category “Culture” (URLs sampled from Delicious catalog using tags culture+blog, culture+magazine, culture+inspiration, culture+design, culture+art, culture+writing, culture+technology, culture+music, culture+uk, culture+movies, and culture+development). This means that content on culture-related pages are considerable less interesting to be copied than content on recreation/sports pages. This reflects the general fact that there are a lot of fans who will be happy to copy information about their teams to other websites, and that auditorium of culture-related websites unlikely will do the same things.

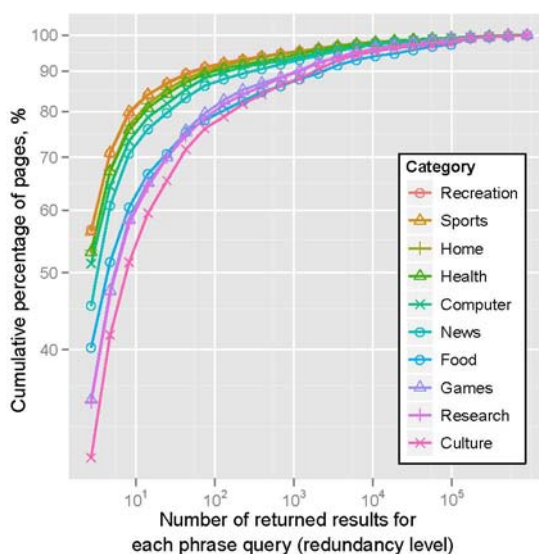


Figure 8: CDF of page redundancies by category in log-log scale for Google  
<sup>(2)</sup>All categories from <http://www.dmoz.org/Recreation/>

Our measurements show that while there is high replication level in some areas (such as area corresponding to “Recreational” category and highly likely in news/ mailing list categories that we have eliminated from our measurements), overall the replication cannot be considered yet a serious problem. In about 40% of cases the content uniquely identifies the source, and in more than 80% cases, a query based on a random phrase from the page yields a very small result set (1-60), which can easily be processed by a human being. However, identification of the original source and information flow (who copied and from where copied, or just from where) would be very useful to arrange results in high-redundancy cases (e.g., for popular categories).

## 4 Related work

There is a large amount of existent work that deals with redundancy. Redundancy has been defined in various ways, based on textual information, internal structure, hashes of the content, external (link) structure, etc.

Brin et al. [3] defined similarity as a significant overlap in the exact sentences (e.g., two documents are considered related if they contain more than 20% of the same sentences). Shiv-akumar and Garcia-Molina [14] employed a vector-space model of the documents and terms to compute asymmetric document closeness measure. Both these approaches require access to the full content of each page, which results in storage and computation scalability issues when applied to the entire corpus Web pages.

To resolve these issues other approaches tried to use the content structure. Among them, the most popular (used by several search engines) is the shingling algorithm [4]. It retrieves every available documents, calculates short syntactic sketches for each document, and then compares these sketches for each pair of documents. The main purpose of this algorithm is to cluster similar documents in big collections. The similar idea was applied by Lin and Ho [9] in design of InfoDiscoverer. On a high level, it partitions pages into several content blocks according to <table> HTML tags and tries to identify informative value (good or redundant) of each content block using calculation of entropies based on the occurrence of the terms (features) in the set of pages. This work mainly focuses on finding the redundant parts in a web page, such as identifying the informative blocks from redundant blocks (like advertisement, banners, navigation panels).

To reduce computation load several proposed algorithms apply hash functions to calculate page similarities. Haveliwala et al. [6] explored a locality-sensitive hashing technique, which for each webpage produces a hash that has high probability to be in a collision with hashes of similar webpages. Similarly, Charikar [5] constructed new locality sensitive hashing schemes using rounding algorithms. Although such techniques provide faster processing, it still requires the whole set of documents to perform actual clustering, which is not realistic for our measurements.

There is also some work to utilize external links and other information to define page similarity. For example, Hou et al. [8] utilized hyperlink transitivity and page importance to measure web page similarity.

One of these algorithms could be ideal tool to find pages with similar content, and thus estimate redundancy. Unfortunately, it is virtually impossible to apply any of these algorithms to the whole Internet: all of them require (at least once) downloading of all the documents. Instead, we employed a more realistic and small-scale approach, where we crawled a small portion of randomly sampled pages and then discovered those which may be duplicate to the sampled pages using popular search engines through the query interface.

As far as we know, there is no much work to measure redundancy level on the real Web. The only work we are aware is the one conducted by Henzinger [7]. He compared the performance of two “state-of-the-art” algorithms developed by popular search engines—Broder’s shingling algorithm [4] and Charikar’s random projection based approach [5])—using a large number (about 1.6B) of web pages. The results show that neither of the algorithms works well for finding near-duplicate pairs on the same sites, while both achieve high precision for near-duplicate pairs on different sites. The measurement work by Henzinger focuses on performance evaluation of the algorithms, whereas our measurements aim to understand characteristics of the Web itself, i.e., the redundancy level on the Web.

## 5 Discussion

### 5.1 Measurement quality

The quality of our measurement depends highly on several factors, and phrase extraction process is one of the most influential. No automated tools can perfectly extract meaningful and identifying phrases from any given random HTML page. Our implementation is not an exception of this rule. In some cases our extraction process makes a “wrong” random decision which phrase to pick from the page. For example, in a small number of cases because of a bad random decision, we extracted phrases like “4. 1. 0. 3. 3. 6. -3. 3.” “4. 3. 0. 1. 15. 4. +11. 9.” “4. 3. 0. 1. 7. 6. +1. 9.” “4. 2. 0. 2. 3. 9. -6. 6.” or “# | A | B | C | D | E | F | G | H | I | J | K | L | M | N |”. These phrases does not have a real meaning

and does not serve identification purposes (second query yields more than 25 billion results in Google).

Other cases when our phrase extraction fails is when there are not much textual information available. For instance, there is a trend in some areas (entertainment, sports, etc.) to have purely Flash- or Silverlight-based websites. In these cases, practically the only textual information available on the page are phrases like “Click here to download the latest version of the Flash Player”. Though phrase queries based on these texts have meaning (some kind of approximation of search for any site with an embedded Flash animations), they are absolutely non-identifying (above mentioned phrase query gives more than 800,000 results in Google, and about 400 millions results in Yahoo and Bing).

Luckily, in most of the cases, providing sufficient amount of textual information available on the page, our algorithm yields quite good results. Our measurements are clear proof of this fact: most of the queries returned a very small number of results—extracted phrases reliably identified the content.

## 6 Conclusion

In this work we performed a small-scale redundancy measurement. We sampled 100,000 links from two Internet services DMOZ and Delicious (link directory and bookmark service) from ten different categories: recreation, sports, home, health, computer, news, food, games, research, culture. We have downloaded HTML pages from all the sample URLs and randomly extracted up to 32-word phrases from these pages. Our actual measurements are based on 100 thousand queries for each of the most popular search engines (Google, Yahoo, and Bing). The results we obtained express a distribution of redundancy similar to a power law (as for each individual category, as well as for across all categories all together). Among all categories, recreational/sports topic has the most number of redundant pages compared to other categories. On the other hand, the least number of redundant pages belong to culture-related pages. The results from each evaluated search engine are consistent (relatively), but Google tends to return higher number of links (estimates) for the same queries, than Bing and Yahoo.

Our initial hypothesis about a bi-modal distribution of the redundancy was not fully confirmed. We did confirm a first portion of this distribution—most of the pages were uniquely identified in the search index by a random 32-word phrase,—but there is no growth in high-redundancy area. However, if we combine all high-frequency pages (say, all pages with redundancy more than 100,000), we will see a small peak. Such combination may be necessary due to a very low precision for these pages (we have to trust estimate and there is no way we can verify it).

## References

- [1] Academic and Research Technologies, Northwestern University. MorphAdomer. <http://morphadorner.northwestern.edu/>, 2009.
- [2] A. Afanasyev, N. Tilley, B. Longstaff, and L. Zhang. BGP routing table: Trends and challenges. In Proceedings of the 12th Youth Technological Conference “High Technologies and Intellectual Systems”, Moscow, Russia, April 2010.
- [3] S. Brin, J. Davis, and H. Garcia-Molina. Copy detection mechanisms for digital documents. In Proceedings of the 1995 ACM SIGMOD international conference on Management of data, pages 398–409. ACM, 1995.
- [4] A. Z. Broder, S. C. Glassman, M. S. Manasse, and G. Zweig. Syntactic clustering of the Web. In Selected papers from the Sixth international conference on World Wide Web, pages 1157–1166, Santa Clara, California, United States, 1997.
- [5] M. S. Charikar. Similarity estimation techniques from rounding algorithms. In In Proc. of 34th STOC, pages 380–388. ACM, 2002.
- [6] T. Haveliwala, A. Gionis, and P. Indyk. Scalable techniques for clustering the web. In WebDB (Informal Proceedings), volume 129, page 134, 2000.
- [7] M. Henzinger. Finding near-duplicate web pages: a large-scale evaluation of algorithms. In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, pages 284–291, Seattle, Washington, USA, 2006.
- [8] J. Hou and Y. Zhang. Utilizing hyperlink transitivity to improve web page clustering. In Proceedings of the 14th Australasian database conference, volume 17, pages 49–57, Adelaide, Australia, 2003.
- [9] S.-H. Lin and J.-M. Ho. Discovering informative content blocks from Web documents. In Proceedings of the eighth ACM SIGKDD international conference on Knowledge Discovery and Data Mining, pages 588–593, Edmonton, Alberta, Canada, 2002.
- [10] A. Ostrow. Bing now powers yahoo search. Online: <http://mashable.com/2010/08/24/bing-powers-yahoo-search/>, August 24, 2010.
- [11] D. Oswald, S. Raha, I. Macfarlane, and D. Walters. HTML Parser. <http://htmlparser.sourceforge.net/>, 2006.
- [12] PostgreSQL Global Development Group. PostgreSQL. <http://www.postgresql.org/>, 2010.
- [13] H. Schildt and J. Holmes. The Art of Java, chapter 6: Crawling the Web with Java, pages 167–216. McGraw-Hill/Osborne, 2003.
- [14] N. Shivakumar and H. Garcia-Molina. SCAM: A copy detection mechanism for digital documents. In Proceedings of the Second Annual Conference on the Theory and Practice of Digital Libraries, 1995.

# РАЗРАБОТКА АЛГОРИТМА АДАПТАЦИИ ТЕХНОЛОГИЧЕСКИХ ПАРАМЕТРОВ СВАРКИ К ИЗМЕНЕНИЮ ГЕОМЕТРИИ СТЫКА ПРИ СВАРКЕ КОРНЕВОГО ШВА

**Шварц М.В.**

*Научный руководитель: к.т.н., доцент Малолетков А.В.*

МГТУ им. Н.Э.Баумана, Москва, Российская Федерация

## DEVELOPMENT OF AN ALGORITHM TO ADAPT THE WELDING PARAMETERS IN RESPONSE TO GROOVE GEOMETRY VARIATION IN WELDING OF ROOT PASS

**Shvarts M.V.**

*PhD, associate professor Maloletkov A.V.*

Bauman Moscow State Technical University, Moscow, Russian Federation

### **Аннотация**

В работе рассматривается процесс орбитальной сварки труб. Разработан алгоритм адаптации технологических параметров сварки к изменению геометрии стыка. Алгоритм представляет собой экспертную модель, для построения которой использовались подходы на основе искусственного интеллекта: нейронные сети, аппарат нечёткой логики, нейро-нечёткие системы.

### **Abstract**

The paper describes an adaptive welding of pipelines. The adaptation algorithm for welding parameters was developed in response for groove geometry variation. Algorithm is an expert model based on artificial intelligence methods: neural networks, fuzzy logic and neuro-fuzzy systems.

При сварке магистральных трубопроводов, предъявляются повышенные требования к точности сборки стыков труб. Это приводит к значительному снижению производительности монтажа трубопровода из-за увеличения временных затрат на обработку кромок, сборку стыка: в среднем, при автоматической сварке около 90% от всего времени идёт на сборку стыка, и лишь 10% на сварку. Одним из способов повышения производительности монтажа трубопроводов является расширение возможностей автомата компенсировать неточности сборки стыка. Другими словами, система управления процессом сварки должна получать информацию о геометрии стыка и соответствующим образом изменять режимы сварки. Так как геометрия стыка может меняться в процессе сварки (от сварочных деформаций), то процесс мониторинга необходимо вести в темпе со сваркой [1].

Такая адаптация сварочного автомата позволит добиться снижения требований по точности сборки стыка, и как следствие, снизить время сборки и обработки труб перед сваркой. При этом повысится качество и производительность процесса укладки трубопровода в целом.

Решение подобных задач в мире проводилось преимущественно с использованием экспертных моделей, которые основаны на построении функции, аппроксимирующей экспериментальные данные.

В настоящее время для аппроксимации функций широко используются подходы на основе искусственного интеллекта: нейронные сети, аппарат нечёткой логики и нейро-нечёткие системы.

Искусственные нейронные сети (ИНС) — сети, в качестве вершин которых выступают искусственные нейроны. ИНС осуществляет преобразование вектора входных сигналов  $X$  в вектор выходных сигналов  $Y$ . В данном случае входными параметрами являются параметры геометрии стыка, а выходными - параметры режима сварки. Связь между ними определяется с помощью набора весовых коэффициентов  $W$ , подбор которых является основной задачей при обучении ИНС [2].

В основе аппарата нечёткой логики лежит база знаний в виде совокупности нечётких правил вида: если  $X$  равно  $A$ , то  $Y$  равно  $B$ . Используя алгоритм нечёткого вывода Мамдани,

для входных параметров  $X$ , которые отсутствуют в базе знаний, вычисляются выходные параметры  $Y$  [3].

Нейро-нечёткая система представляет собой нейронную сеть, каждый слой которой реализует один из этапов нечёткого вывода Сугено. Другими словами, выводы делаются на основе аппарата нечёткой логики, а настройка происходит с использованием алгоритма обучения нейронных сетей [3].

Получение экспериментальных данных, для обучения моделей, проводилось в два этапа. На *первом этапе* проводилась сборка (на прихватках) под сварку стыка труб диаметром 1220 мм и толщиной стенки 17 мм. Затем с помощью системы слежения триангуляционного типа, установленной на автомате УАСТ-1, проводилось сканирование разделки и получение информации о её геометрии (рис. 1.).

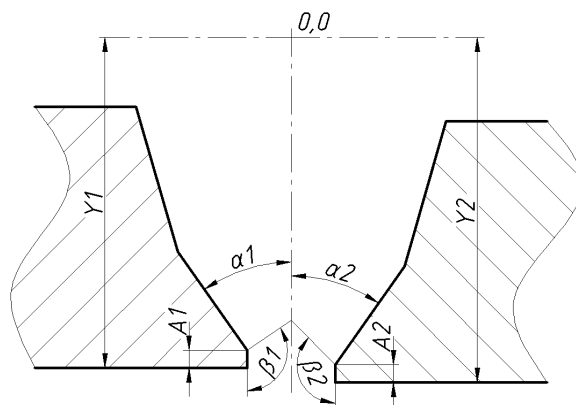


Рисунок 1 - Измеряемые параметры геометрии разделки

На *втором этапе* стык был сварен автоматом УАСТ-1 для сварки плавящимся электродом в защитном газе, при этом квалифицированный сварщик имел возможность в темпе с процессом корректировать режимы сварки, что регистрировалось и сохранялось в памяти компьютера. Таким образом, были получены входные и выходные параметры обучающей выборки, представленные в Таблице 1.

Таблица 1 - Входные и выходные данные обучающей выборки

Входные параметры		Выходные параметры	
Депланация кромок, мм	$Y1$	Скорость сварки, мм/с	$V_{св}$
	$Y2$		
Притупление комок, мм	$A1$	Базовый ток, А	$I$
	$A2$		
Угол разделки, °	$\alpha1$	Скорость подачи проволоки, мм/с	$V_{пп}$
	$\alpha2$		
Наличие зазора/прихватки	$\beta1$	Размах колебаний, мм	$A$
	$\beta2$		

Пример изменения геометрии стыка представлен на рис.2. Здесь номер измерения - это точки окружности трубы, в которых проводилось измерение геометрии.



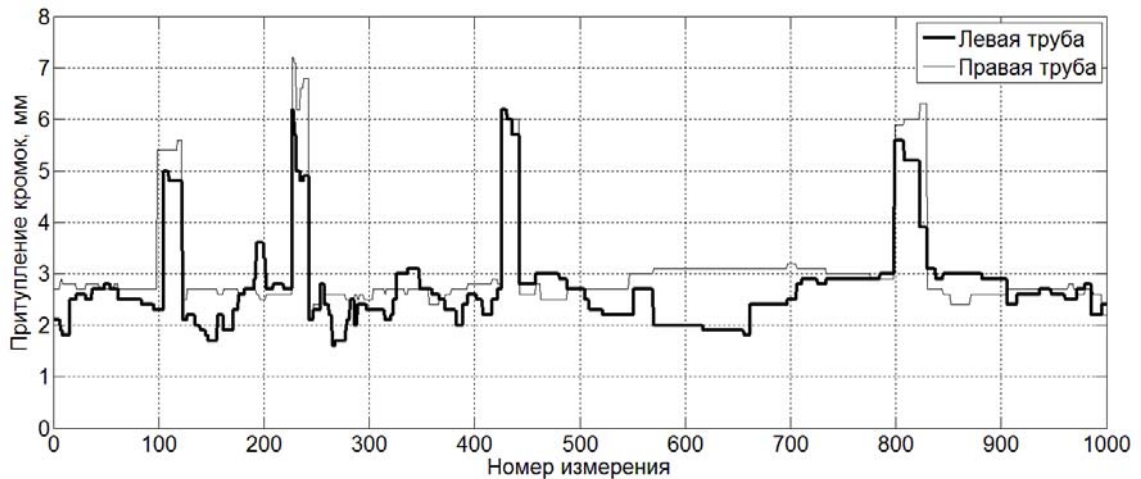


Рисунок 2 - Изменение геометрии стыка

Как было отмечено выше, для аппроксимации функций, а данном случае экспериментальных данных, используются нейронные сети, аппарат нечёткой логики и нейро-нечёткие системы. В качестве нейронных сетей в работе были рассмотрены сети на основе многослойного перцептрона (*MLP-сети, Multi Layer Perceptron*), сети на основе радиально базисных функций (*RBF-сети, Radial-Basis Functions*) и обобщённо - регрессионные нейронные сети (*GRNN - сети, General Regression Neural Networks*). Выбор именно этих сетей обусловлен тем, что для этих сетей существует теорема об универсальной аппроксимации, смысл которой в том что, данные сети способны аппроксимировать любую последовательность с заданной точностью [2, 3].

Исследование проводилось в среде Matlab 7.0, имеющей средства для проектирования, моделирования, обучения и использования множества известных парадигм аппарата искусственных нейронных сетей (*Neural Network Toolbox*). Среда также позволяет создавать нечёткие и гибридные системы (*Fuzzy Logic Toolbox*) [3].

Результаты обучения представлены на рис.3-5. Все экспертные модели обладают достаточным быстродействием и пригодны для применения в темпе с процессом.

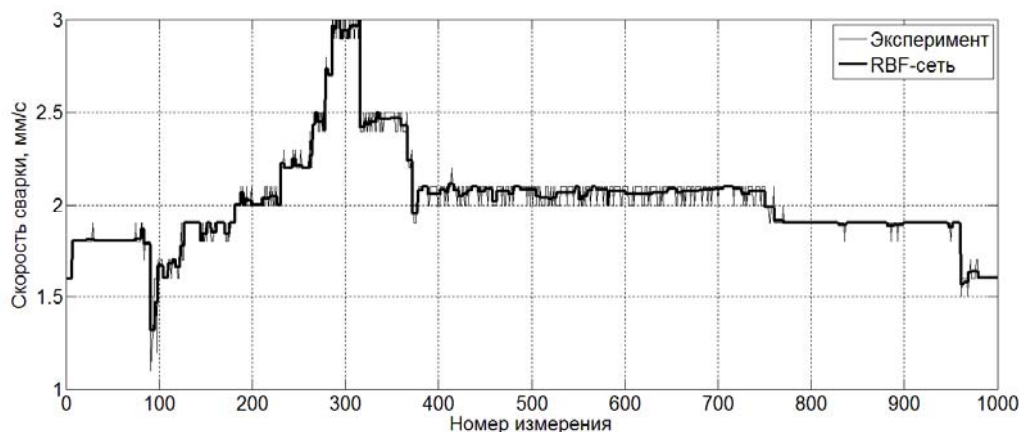


Рисунок 3 - Результаты обучения (нейронная сеть)

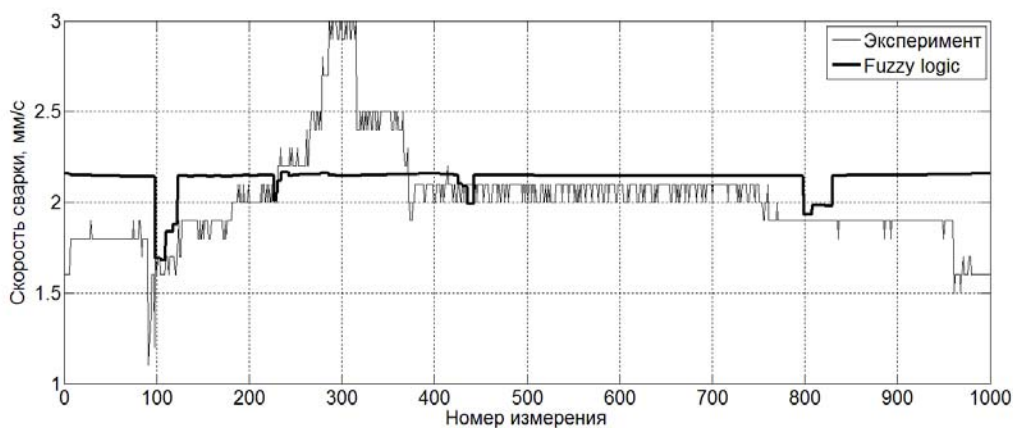


Рисунок 4 - Результаты обучения (аппарат нечёткой логики)

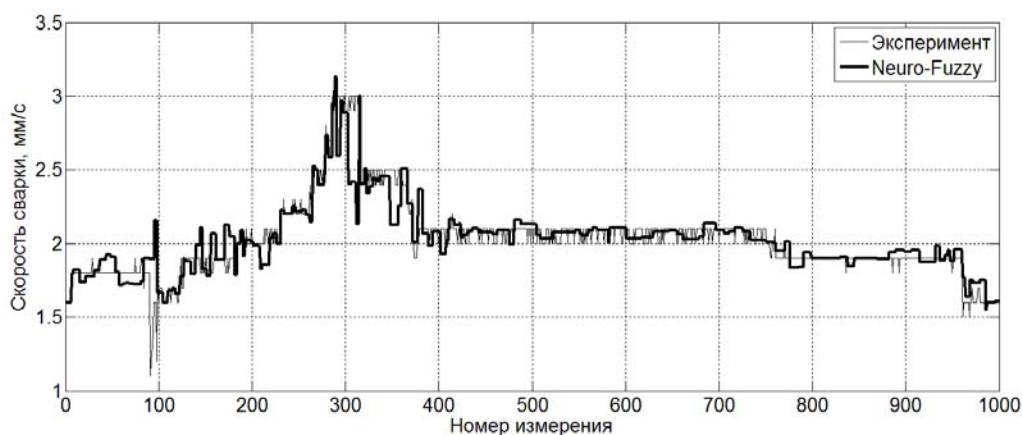


Рисунок 5 - Результаты обучения (нейро-нечёткая система)

Резюмируя вышесказанное, можно сделать следующие выводы.

1. Установлена принципиальная возможность использования экспертных моделей для адаптации сварочного автомата к изменению геометрии стыка.
2. Разработаны алгоритмы адаптации автомата, которые можно условно разделить на три группы:
  - 2.1. алгоритмы, при реализации которых чётко прослеживается связь между изменением геометрии стыка и изменением режимов сварки;
  - 2.2. алгоритмы, воспроизводящие действия опытного сварщика;
  - 2.3. алгоритмы, объединяющие свойства в п.2.1. и п.2.2.

### Литература

1. Э.А.Гладков, Управление процессами и оборудованием при сварке – М.: Издательский центр "Академия", 2006. - 432 с.
2. Хайкин С., Нейронные сети. Полный курс. – М.: ООО "И.Д.Вильямс", 2008.
3. В.В. Круглов, М.И. Дли, Р.Ю.Голунов, Нечёткая логика и искусственные нейронные сети. – М: "Физматлит", 2001.

# ПРИМЕНЕНИЕ МЕТОДА РОЯ ЧАСТИЦ ДЛЯ ЗАДАЧ ПРОЕКТИРОВАНИЯ БУРОВЫХ РАСТВОРОВ

Селиверстов Е.Ю.

*Научный руководитель: д.ф.-м.н., профессор Карпенко А.П.*

Московский государственный технический университет им. Баумана, Москва, Россия

E.U. Seliverstov

*Sc.D., professor A.P. Karpenko*

Bauman Moscow State Technical University, RK6 department, Moscow, Russian Federation

## Аннотация

Рассмотрена задача оптимизации состава бурового раствора. Предложен параллельный метод поведенческой оптимизации роем частиц в составе программного комплекса. Произведены сравнительные оценки эффективности решения.

## Abstract

Optimization problem for drilling mud research is discussed. Parallel method of global particle swarm optimization as a part of software system is proposed. A comparative study of methods performance is given.

## Введение

Рассматривается подход к решению задач проектирования буровых растворов на основе поведенческих алгоритмов непрерывной оптимизации. Задача относится к классу задач химической кинетики. Процесс сложной химической реакции описывается множеством параллельно выполняющихся элементарных стадий превращений. Математическая модель процесса представляет собой систему дифференциальных уравнений, где переменными являются концентрации веществ, а внешними параметрами, в частности, – константы скорости реакции. В статье рассматривается обратная кинетическая задача, заключающаяся в определении параметров модели по известным экспериментальным данным [1].

## Постановка задачи

Рассмотрим модельную четырехкомпонентную химическую реакцию. Обозначим молярные концентрации веществ как  $X_i$ ,  $i \in [1;4]$ , а константы скоростей реакции  $k_j$ ,  $j \in [1;n]$ ,  $n = 4$ .

Математическая модель реакции представляет собой систему обыкновенных дифференциальных уравнений (ОДУ) в нормальной форме Коши:

$$\begin{cases} \dot{X}_1 = -W_1(t) \\ \dot{X}_2 = 2W_1(t) - W_2(t) \\ \dot{X}_3 = -W_2(t) \\ \dot{X}_4 = W_2(t) \end{cases} \quad (1)$$

где

$$\begin{aligned} W_1(t) &= k_1 X_1(t) - k_2 X_2^2(t) \\ W_2(t) &= k_3 X_2(t) X_3(t) - k_4 X_4(t) \end{aligned} \quad (2)$$

и задан вектор начальных условий  $\mathbf{X}(0) = (X_i(0) = X_i^0, i \in [1;4])$ .

Известны решения системы уравнений (1)  $\mathbf{X}^e(t_p^e)$  в некоторые узловые моменты времени  $t_p^e$ ,  $p \in [1;m]$ . Введем обозначения:  $\mathbf{k}$  –  $n$ -мерный вектор искомых величин  $k_j$ ,  $\mathbf{X}(\mathbf{k}, t) = (X_i(\mathbf{k}, t), i \in [1;4])$ . Множество допустимых значений вектора варьируемых параметров  $D_k$  образуется ограничениями:

$$0 \leq k_j \leq k_j^{\max} \quad k_j^{\max} = 20 \quad (3)$$

$$k_2 \geq k_1 \quad (4)$$

где  $j \in [1; n]$ .

Ставится следующая задача: найти допустимый вектор  $k = k^*$ , который минимизирует невязку

$$\Phi(\mathbf{k}) = \sum_{p=1}^m \|\mathbf{X}^e(t_p) - \mathbf{X}'(t_p)\| \quad (5)$$

Здесь  $\mathbf{X}'(\mathbf{k}, t)$  - нормализованное решение  $\mathbf{X}(\mathbf{k}, t)$ :

$$X'_j(t_p) = \frac{Z_j(t_p)}{\sum_{k=1}^4 Z_k(t_p)} \quad (6)$$

$$Z_k(t_p) = \begin{cases} \square & X_k(t_p) & \exists X_k^e(t_p) \\ \square & 0 & !\exists X_k^e(t_p) \end{cases} \quad (7)$$

Формула (7) учитывает тот факт, что существуют неопределенные величины  $X_k^e(t_p)$ .

Таким образом, поставленная задача представляет собой многопараметрическую задачу глобальной условной оптимизации

$$\min_{k \in D_k} \Phi(\mathbf{k}) = \Phi(\mathbf{k}^*) \quad (8)$$

Отметим, что каждое вычисление значения функции  $\Phi(\mathbf{k})$  сопряжено с высокими вычислительными затратами, поскольку требует интегрирования системы ОДУ (1) (2).

### Метод роя частиц

Для решения задачи обратной кинематики изначально использовался метод полного перебора по дискретной сетке, покрывающей пространство поиска  $D_k$  с фиксированным шагом. Естественно предложить более эффективные методы оптимизации. Из-за высокой вычислительной сложности критерия оптимальности  $\Phi(\mathbf{k})$  используем параллельный метод роя частиц [2]. Существует несколько параллельных моделей этого метода [3]. В работе [4] нами предложена одна из таких моделей и реализации метода для центрального и графического процессоров. В данной работе также используется этот метод. Рассмотрим схему метода.

Рассмотрим задачу глобальной безусловной минимизации целевой функций  $\Phi(Y)$  в  $n$ -мерном арифметическом пространстве  $R^n$ :  $\min \Phi(Y) = \Phi(Y^*)$ ,  $X \in R^n$ . Множество частиц обозначим  $S = \{P_i, i \in [1; L]\}$ , где  $L$  – число частиц в рое (размер популяции). В момент времени  $t = 0, 1, 2, \dots$  координаты частицы  $P_i$  определяются  $n$ -мерным вектором  $Y_{i,t}$ , а скорость –  $V_{i,t}$ . Итерации в методе PSO выполняются по следующей схеме:

$$Y_{i,t+1} = Y_{i,t} + V_{i,t+1} \quad (9)$$

$$V_{i,t+1} = \chi(V_{i,t} + \phi \otimes (Z_{i,t}^g - Y_{i,t})) \quad (10)$$

где

$$Z_{i,t}^g = \frac{U[0, \phi_1] \otimes Y_{i,t}^b + U[0, \phi_2] \otimes Y_{i,t}^g}{\phi} \quad (11)$$

Здесь  $U[a, b]$  представляет собой  $n$ -мерный вектор псевдослучайных чисел, равномерно распределенных в интервале  $[a, b]$ ;  $Y_{i,t}^b$  - вектор координат частицы  $P_i$  с наилучшим значением целевой функции  $\Phi(Y)$  за все время поиска;  $Y_{g,t}$  – вектор координат соседней с данной частицы с наилучшим за время поиска значением целевой функции  $\Phi(Y)$ ;  $\chi, \phi_1, \phi_2$

– константы, свободные параметры алгоритма. Соседство частиц определяется используемой топологией соседства [5].

Разобьем весь рой  $S$  на  $M$  подроев  $S_i$ ,  $i \in [1; M]$ . Количество частиц в каждом рое  $S_i$  обозначим  $M_i$ . Положим размеры подроев одинаковыми:  $M_i = \frac{L}{M}$ . Суброи обрабатываются независимо на протяжении «сезона», состоящего из  $T_{mig}$  итераций. По окончании каждого из сезонов выполняется процедура миграции частиц между суброями, которая может осуществляться с использованием стратегии перемещения и стратегии репликации. Миграция перемещением частиц заключается в обмене наилучшей частицы из одного суброя и наихудшей частицы из другого суброя. Выбор суброя назначения  $S_i$  может происходить случайным образом, либо в соответствии с заданной топологией соседства частиц [5]. В миграции репликацией сначала в каждом из суброев  $S_i$  осуществляется поиск лучшей частицы  $P_{best}^l$  и худшей частицы  $P_{worst}^l$ ;  $l \in [1; M]$ . Выбирается наилучшая частица среди всех роев  $P_{best}^{l*}$ . Затем из суброя  $S_{i^*}$  осуществляется репликация (копирование) частицы  $P_{best}^{l*}$  во все остальные суброи  $S_i$  на место частиц  $P_{worst}^l$ .

Заметим, что миграция перемещением имеет целью исключить наиболее успешную частицу из суброя в случае тенденции его к преждевременной сходимости. При этом уменьшается скорость сходимости алгоритма, но обеспечивается более планомерное исследование пространства поиска. Миграция репликацией, наоборот, преследует цель обеспечить более высокую скорость сходимости итераций. В работе использована стратегия репликации частиц (стратегия обеспечивает быструю сходимость метода).

### Результаты исследования

Задача представляет собой задачу условной оптимизации. Поскольку метод роя частиц предназначен для решения задач безусловной оптимизации, скомбинируем его с методом штрафных и барьерных функций. При нарушении граничных условий (3) (4) численное решение системы ОДУ (1) (2) становится неустойчивым. На этом основании ограничение (3) задается барьерной функцией. Ограничение (4) задается штрафной функцией.

Было выполнено исследование характера целевой функции  $\Phi(\mathbf{k})$  в области  $D_k$ . Применим полный перебор по дискретной сетке, покрыв область равномерной сеткой с шагом  $h=1$  по всем измерениям. Во всех узлах сетки были вычислены значения целевой функции. Решение ОДУ (1) (2) для вычисления функции  $\Phi(k)$  производилось при помощи метода Рунге-Кутты 4-го порядка с постоянным шагом 0.01. Выявлено, что глобальный минимум функции  $\Phi(\mathbf{k})$  в области  $D_k$  примерно составляет  $\Phi_{min}^* = 8.1$ . Такая величина невязки показывает, что в условиях исследования модель (1) (2) не позволяет точно аппроксимировать заданный набор экспериментальных данных.

Кроме того, произведена оценка чувствительности целевой функции к изменениям значений параметров  $k_j$ . Исследование показало, что при фиксированных значениях параметров  $k_1$  и  $k_2$  функция  $\Phi'(x_3, x_4)$  имеет большую область, в которой ее значения не меняются и, таким образом, любое из решений будет оптимальным.

Основное исследование выполнено с использованием следующих методов оптимизации:

- *FULL* – метод перебора на фиксированной сетке;
- *PSO* – последовательный однороевой метод роя частиц;
- *IPSO* – параллельный метод роя частиц на основе островной модели параллелизма.

В методах выбраны *PSO*, *IPSO* использованы стандартные для многопараметрических задач значения параметров [4]. Число частиц в рое принималось равным 16, число роев для *IPSO* равным 4. Поскольку параметры, характеризующие сходимость метода роя частиц, в

значительной мере зависят от начальных условий  $Y^0$ ,  $V^0$ , каждый эксперимент повторялся 20 раз.

Для оценки скорости сходимости методов PSO, IPSO ставилась задача достижения значения целевой функции  $\Phi(\mathbf{k})$ , равного  $\Phi_{min}^*(\mathbf{k})$  исходя из рассмотренных выше результатов метода *FULL*. Результаты данного исследования иллюстрирует таблица (1). Значения  $M_t$  – оценка математического ожидания числа итераций,  $\sigma_t$  – аналогичная оценка среднего квадратичного отклонения.

Метод	Число испытаний $E$	Число итераций, $M_t$	Число итераций, $\sigma_t$
FULL	160000	160000	0
PSO	928	58	17
IPSO	2752	43	8

Таблица 1: Скорость сходимости методов

Таблица 1 показывает, что по сравнению с классическим методом PSO, метод IPSO обеспечивает более высокую скорость сходимости итерационного процесса (благодаря механизму межроевой миграции частиц). При этом общее число испытаний (вычислений значений функции)  $E$  у метода IPSO, естественно, выше. Таким образом целесообразна реализация этого метода на параллельных вычислительных системах.

### Заключение

Для решения обратной кинетической задачи проектирования бурового раствора предложено использование метода оптимизации роем частиц. Выполнено исследование эффективности метода, показывающее перспективность его использования на параллельных вычислительных системах.

### Литература

1. Линд, Ю. Методология параллельных вычислений□ для решения задач химической кинетики и буровой технологии / Ю. Линд, И. Губайдуллин, Р. Мулюков // Системы управления и информационные технологии. — 2009. — №2. — с. 44–50.
2. Kennedy, J. Particle swarm optimization / J. Kennedy, R. Eberhart // In Proceedings of IEEE International conference on Neural Networks. — 1995. — с. 1942–1948.
3. Belal, M. Parallel models for particle swarm optimizers / M. Belal, T. El-Ghazawi // IJICIS. — 2004. — №1. — с. 100–111.
4. Карпенко, А. Глобальная оптимизация методом роя частиц на графических процессорах / А. Карпенко, Е. Селиверстов // Научный сервис в сети Интернет: масштабируемость, параллельность, эффективность: Труды Всероссийской суперкомпьютерной конференции.— Издательство МГУ, 2009. — с. 188–191.
5. Карпенко, А. Обзор методов роя частиц для задачи глобальной оптимизации / А. Карпенко, Е. Селиверстов // Электронное научно-техническое издание «Наука и образование». — 2009. — №3.

# МИРОВЫЕ ТЕНДЕНЦИИ УПРАВЛЕНИЯ БИЗНЕС-ПРОЦЕССАМИ В СФЕРЕ ИНТЕРНЕТ-КОММЕРЦИИ

Баранов А.В.

*Научный руководитель доц., к.т.н. Волосатова Т.М.*  
РК-6 САПР, МГТУ имени Н.Э.Баумана, Москва, Россия

## GLOBAL TRENDS OF BUSINESS PROCESSES IN E-COMMERCE

Baranov A.V.

*The supervisor of studies prof. Volosatova T.M.*  
RK-6 CAD Systems, Bauman Moscow State Technical University, Moscow, Russia

### Аннотация

Электронная торговля может быть использована во многих сферах для поддержки маркетинговой деятельности, включая общение с клиентами, предлагая продукты для продажи и разработку новых маркетинговых стратегий. Многие из малых компаний, как правило, считают, что интернет-коммерция не представляет дополнительных возможностей для развития их бизнеса и более качественного обслуживания клиентов. Дилемма заключается в том, что повседневная жизнь, не дает им время рассмотреть этот вопрос и ознакомиться с состоянием ИТ- это, в свою очередь, оставляет ряд неиспользованных возможностей.

### Abstract

E-commerce can be used in many ways to support marketing activities, including communication with customers, offering products for sale, and developing new marketing strategies. Many of the small, traditional companies, typically those in labour-intensive segments of the sector, believe that ICT does not present a business opportunity for them, neither for improving the products nor for providing a better service to customers. The dilemma is that they are pressured in their daily routines do not give them the time to consider the issue and become knowledgeable about ICT – this, in turn, leaves ICT-based opportunities unexploited.

Оценим готовности зарубежных и российских компаний к переходу к так называемому электронному бизнесу. Согласно определению Организации по экономическому сотрудничеству и развитию (ОЭСР) под электронным бизнесом понимается осуществление автоматизированных бизнес-процессов (внутрифирменных и межфирменных) посредством компьютерных сетей. Следовательно, довольно широко распространенное понимание электронного бизнеса в качестве проекта, реализуемого в сети Интернет, сужает смысл данного понятия.

Аналитики Европейской комиссии, изучающие состояние и тенденции развития электронного бизнеса в странах Западной Европы, уверены в том, что для экономики будущего критичны информационная инфраструктура (внутри компании и внутри государства), информационные услуги и информационно-технологическое обеспечение процессов. Перечисленное выше становится возможным благодаря электронным информационным технологиям, в особенности технологиям коммуникации. Однако основная идея электронного бизнеса заключается не во внедрении технологий и систем, а прежде всего в оптимизации взаимоотношений с клиентами, поставщиками и другими партнерами. Причем взаимоотношения не ограничиваются совершением коммерческих сделок, но подразумевают информационный обмен с партнерами. В то же время электронный бизнес – это оптимизация не только техпроцессов, которые ориентированы во внешнюю среду, но также и внутренних процессов фирмы. Нет смысла концентрировать усилия на внешних коммуникациях, если внутренние процессы неэффективны.

Рассмотрим современное состояние и тенденции развития электронного бизнеса в странах Запада, многие из которых относятся к числу лидеров по показателю готовности бизнеса к использованию возможностей электронных коммуникаций («Networked Readiness Index») [1]. Обратимся к отчету «The European e-Business Report 2009» исследовательской группы «Sectorale-Business Watch», созданной по инициативе Европейской комиссии специально в целях мониторинга электронного бизнеса в Европе [2]. В настоящее время в

странах Евросоюза применение базовых способов электронного бизнеса (таких как коммуникации через веб-сайт; прием и размещение заказов в режиме онлайн) стало практически обязательным для фирм всех размеров и отраслей. Технологии, обеспечивающие такие способы, стали «предметом всеобщего потребления», необходимы всем фирмам.

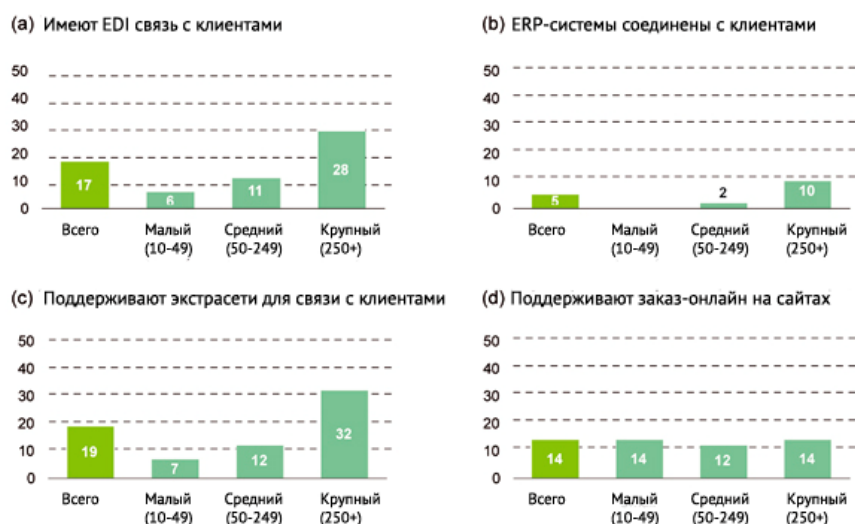


Рисунок 1 - Различные каналы и платформы, используемые для взаимодействия с клиентами в рамках электронной коммерции

Оборотной стороной медали является неспособность этих способов принести стратегические преимущества использующим их компаниям. Более прогрессивные способы электронного бизнеса (такие как торговля через специальные электронные торговые платформы – площадки, непосредственная интеграция информационных систем партнеров) становятся все более необходимы для крупных компаний, тогда как для малых и средних фирм их освоение пока еще не стало насущной необходимостью. Однако именно в небольших фирмах прогрессивные способы электронного бизнеса имеют наибольший стратегический потенциал.

Заслуживает внимания тот факт, что небольшие фирмы всё чаще начинают испытывать давление со стороны крупных партнеров по поводу внедрения тех или иных электронных технологий для оптимизации взаимодействия (Рисунок 1). Впрочем, подобное случалось на протяжении всей истории существования электронной коммерции. Уровень готовности компаний к электронному бизнесу можно примерно оценить по степени освоения конкретных видов деятельности и внедрения тех или иных технологий или систем.

Если ранее основное внимание бизнеса привлекали средства улучшения внутренних коммуникаций, то сейчас на первый план выходят средства и услуги информационной интеграции нескольких компаний [3]. Явно вырисовывающейся целью компаний является переход на высоко автоматизированный электронный обмен, подразумевающий не просто передачу документов через электронные сети, но их автоматическое формирование информационной системой отправителя и автоматическую обработку системой получателя. Конечно, автоматический электронный обмен практически недостижим без предварительной автоматизации внутренних процессов всех участников цепи. Пока невелика доля компаний, в которых управление высоко автоматизировано. Помимо стремления фирм к информационной интеграции в цепочках создания ценности всё более заметным становится стремление к установлению электронных коммуникаций для реализации совместных проектов с тем, чтобы создать качественно новые условия для кооперации. Склонность к межфирменному сотрудничеству стимулирует и появление нового поколения программных средств коммуникации.



Внешне-ориентированные электронные процессы имеет смысл подразделить по сложности их организации. Так, организация электронных коммуникаций, обслуживающих совместную проектную деятельность, очевидно, сопряжена с большими трудностями, чем организация электронной торговли или электронного обмена информацией в цепочках поставок. Проектная деятельность, в которую вовлечены несколько фирм, подразумевает развитие слабо формализованных отношений между их сотрудниками. Эта задача нетривиальна, особенно если участники разрознены территориально и/или культурно. Электронные коммуникации должны не ограничивать неформальные отношения, а стимулировать их активность, придавать им большую насыщенность. Выход на высший уровень взаимоотношений ставит перед партнерами более сложные задачи как технологические (обеспечение надежности и безопасности информационного обмена), так и организационные (формирование атмосферы взаимного доверия и ответственности партнеров).

Большая часть российских промышленных компаний пока не достигли удовлетворительной степени внутренней автоматизации. К полномасштабной автоматизации они приступили сравнительно недавно (2–4 года назад), к тому же, поскольку в отечественной промышленности велика концентрация крупных холдингов, «внутренняя автоматизация» таких компаний сводится к решению гораздо более сложной задачи – интеграции входящих в компанию предприятий. Закономерно, что внедрение информационных систем в масштабах холдинга требует гораздо больше ресурсов и времени, чем автоматизация управления на отдельном предприятии. Таким образом, промышленные компании России пока довольно слабо подготовлены к осуществлению электронных коммуникаций с субъектами внешней среды в приближенных к рыночным условиям.

Безусловно, Россия может получить преимущества от устойчивого и эффективно функционирующего рынка электронной коммерции. Для этого существует ряд предпосылок:

- накоплен достаточный мировой опыт организации электронной коммерции;
- мощная тенденция развития современной телекоммуникационной инфраструктуры;
- наличие вспомогательной инфраструктуры, позволяющей разрабатывать надлежащее программное обеспечение любой сложности, а также информационные продукты для нужд электронной торговли;
- наличие инфраструктуры людских ресурсов, необходимых для информирования делового сообщества о выгодах и преимуществах электронной коммерции, а также подготовки рабочей силы, необходимой для использования этих преимуществ;
- положительная тенденция в развитии правовой и политической инфраструктуры, которая будет оказывать содействие и поддержку электронной торговле.

На фоне стремительного роста электронной коммерции в Америке и Европе данная отрасль находится в России на начальном этапе развития

Расширение доступа в Интернет, формирование правовой базы на основе использования информационно-телекоммуникационных технологий, разработка и внедрение системы государственной поддержки электронной торговли, развитие систем безналичных расчетов являются основными вопросами, разрешение которых будет способствовать более активному развитию электронной коммерции в России.

## Литература

1. Global IT Report. World EconomicForum. [Электронный ресурс] Режим доступа: <http://www.insead.edu/v1/gitr/wef/main/home.cfm>.
2. The European e-Business Report 2009. The Sectoral e-Business Watch. [Электронный ресурс] Режим доступа: <http://www.ebusiness-watch.org>.
3. Adoption of Information & Communication Technologies (ICT) by Business. European Commission, Information Society and Media DG. [Электронный ресурс] Режим доступа: [http://ec.europa.eu/information\\_society/europe/i2010/docs/info\\_sheets/7-2b-i2010-innovationen.pdf](http://ec.europa.eu/information_society/europe/i2010/docs/info_sheets/7-2b-i2010-innovationen.pdf).

# РАЗРАБОТКА ПРОГРАММНОГО СРЕДСТВА ДЛЯ РЕГИСТРАЦИИ ИЗОБРАЖЕНИЙ, ФОРМИРУЕМЫХ ДВУМЯ WEB-КАМЕРАМИ И «ЗАХВАТА» КАДРА В ВИДЕОПОРЯДЕ

**Большакова Д.О., Матвеева К.О.**

*Научный руководитель: к.т.н., доцент каф. РК-6, Волосатова Т. М*  
МГТУ им. Н.Э.Баумана, Москва, Россия

## SOFTWARE DEVELOPMENT FOR RECOGNITION OF IMAGES, WHICH ARE REGISTERED BY TWO WEB-CAMERAS AND «CAPTURING» THE VIDEO FRAME

**Bolshakova D., Matveeva K.**

*Research supervisor, candidate of engineering science, associate professor of CAD/CAE/PLM  
department, Volosatova T.*

Bauman Moscow State Technical University, Moscow, Russia

### **Аннотация**

Одной из наиболее актуальных задач в современном мире информационных технологий является обработка цифровых и видео изображений. В настоящее время разработано большое количество программных средств, нацеленных на решение данной задачи. Подобные программы могут выступать как самостоятельные единицы или быть составной частью программного комплекса. В данной статье рассматривается программное средство для работы со статическими изображениями и видеофрагментами, входящее в состав программного продукта (АРМ), содержащего ряд программных модулей и позволяющего распознавать трехмерные объекты. Дается краткое описание некоторых методов работы с видео и сведения об особенностях форматов воспроизведения.

### **Abstract**

One of the most actual tasks in modern IT sphere is processing and recording of images and videos. Nowadays a lot of programs, aimed at solving this problem are created. Such software tools could be either independent products or a part of software complex. The article is devoted to program that works with static images and videos. Programming product, which allows identifying three-dimensional object, contains several modules and includes this program. Brief description of video processing methods and information about playback format features are given.

Информация – это мера неопределенности в пространстве событий. Ее логическим носителем является сообщение, один из наиболее ценных ресурсов в современном мире. Сообщения кодируются в текстовой, числовой, звуковой, графической, двоичной форме. Для каждого типа кодов должен быть предусмотрен набор программно-аппаратных средств для распознавания, хранения, передачи и воспроизведения сообщений.

Рассмотрим программный продукт для работы с видео и цифровыми изображениями. Он разработан на базе системы среды Matlab и состоит из двух частей: первая – программный интерфейс, объединяющий 8 модулей программного продукта (АРМ) и позволяющий регистрировать изображения, формируемые двумя WEB-камерами, вторая – обработка видео. Выбор системы среды Matlab обусловлен следующими факторами:

- наличие среды GUIDE для создания приложений с графическим интерфейсом пользователя;
- удобство реализации элементов управления и программирования событий;
- широкий набор функций для работы с графикой;
- возможность реализации широкого круга задач.

Программное средство для одновременной регистрации изображений, формируемых двумя WEB-камерами должно обеспечивать:

- Занесение в буфер обмена видеофильмов и отдельных кадров, регистрируемых двумя WEB – камерами, работающих в режиме цифровых фотоаппаратов.
- Просмотр двух видеофильмов.

Под программным интерфейсом подразумевается функциональность, которую некоторый

программный компонент предоставляет другим программным компонентам. Можно различать два вида такой функциональности: та, что используется при создании прикладных программ, и та, что используется при создании системных компонентов. Для регистрации изображений в данном случае используется веб-камера.

Формирование высококачественных видеоданных является одним из необходимых условий для их эффективной обработки с целью получения достоверной информации. Для решения задачи регистрации изображений и видеоданных может использоваться Image Acquisition Toolbox. Это приложение позволяет непосредственно подключать, настраивать и управлять средствами формирования изображений и потокового видео. Вместе с другими приложениями, например с такими как Image Processing Toolbox, который реализует обработку сигналов и изображений, Image Acquisition Toolbox обеспечивает высокий уровень проведения анализа и обработки данных. Приложение Image Acquisition Toolbox работает со следующим широким набором аппаратного обеспечения - платы Matrox, платы Data Translation, платы видеозахвата в форматах WDM (Windows Driver Model) и VFW (Video for Windows), веб-камеры, видеокамеры с интерфейсами USB и FireWire (IEEE-1394), цифровые (DV) видеокамеры.

Стандартные средства разработки позволяют наиболее эффективно работать с аппаратными средствами, создавая объекты захвата и имея возможность редактирования их свойств, предоставлять непосредственный доступ к фреймам изображений объектов. Приложение Image Processing Toolbox предоставляет широкий набор возможностей работы с графической информацией: преобразование форматов представления данных, работа с графическими форматами файлов (чтение/запись), вывод изображений на экран и захват их с экрана, преобразования типов изображений, геометрические преобразования изображений.

При разработке данного ПО считалось, что работа будет вестись либо со статическими объектами, либо, задержка на обработку изображений будет мало относительно скорости движения объекта. В представленном ПО реализуется создание исходных видео объектов obj, формируются свойства объектов, реализуется непосредственный доступ к фреймам изображений frame из исходных видео объектов getsnapshot. Полученные изображения можно сохранять как в формате, удобном для работы MathLab (.mat), так и во многих широко используемых форматах изображений. Например, функция save filename запоминает все переменные в рабочем пространстве MATLAB в MAT-файл. Когда в название файла filename не включено расширение, файл запоминается с расширением .MAT. В рассматриваемом ПО, для удобства дальнейшего использования зарегистрированных изображений, реализована возможность сохранять изображений в формате .jpeg или .bmp, которые являются наиболее популярными форматами хранения изображений. Алгоритм JPEG в наибольшей степени пригоден для сжатия фотографий и картин, содержащих реалистичные сцены с плавными переходами яркости и цвета. С другой стороны, JPEG не подходит для сжатия изображений при многоступенчатой обработке, так как искажения в изображения будут вноситься каждый раз при сохранении промежуточных результатов обработки. BMP позволяет хранить аппаратно-независимые растровые изображения. Глубина цвета в данном формате может быть 1, 2, 4, 8, 16, 24, 32, 48 бит на пиксель.

Неотъемлемая часть современного мира изображений это видеофайлы. Согласно определению, видео - это множество технологий записи, обработки, передачи, хранения и воспроизведения визуального или аудиовизуального материала. Видео формат определяет структуру видео файла. Различают следующие форматы: MPG, AVI, MOV, RealVideo, ASF, DivX, FLV, WMV. Наиболее распространенными из них являются AVI и MOV. Audio-Video Interleaved представляет собой контейнер, который может содержать в себе потоки 4 типов - Video, Audio, MIDI, Text. При этом видео поток может быть только один. Формат MOV, как и AVI, совмещает в себе различные видеокодеки и просматривается большинством проигрывателей. Также используется в фото- и видеокамерах для съемки. Основное отличие этих форматов заключается в способе хранения видео потока. В MOV содержатся ключевые кадры и промежуточные, в результате изменения ключевого меняется ряд промежуточных

до следующего ключевого кадра. В AVI же кадры не подразделяются и равноправны между собой, а следовательно, изменение одного не влечет изменение другого.

В рассматриваемом ПО также реализована регистрация видео в формате .AVI. Запись видео с помощью приложения Image Acquisition Toolbox осуществляется покадрово с последующим переформатированием данных в формат видеофайла .AVI.

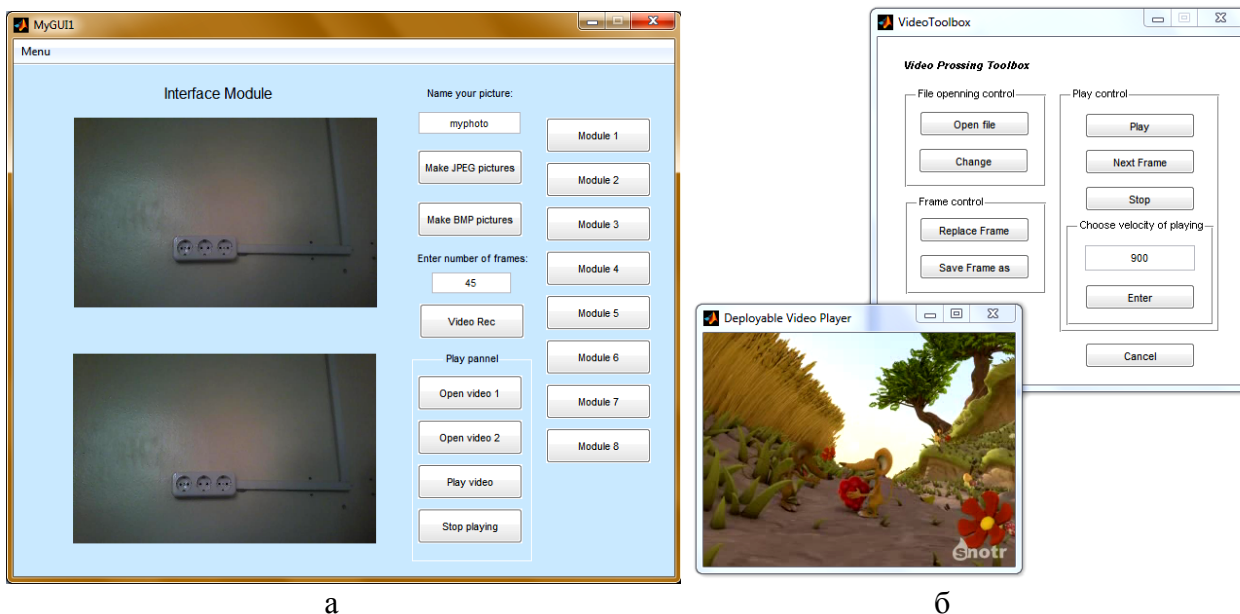


Рисунок 1 - Программный интерфейс а) позволяющий регистрировать изображения и видео, формируемые двумя WEB-камерами б) позволяющий обрабатывать видео файлы

Существует два основных метода обработки видео в системе Matlab: с помощью встроенных функций и системных объектов. Первый способ - это использование метода `mmreader` для открытия файла, метода `read` для чтения кадров и функции `movie` для проигрывания видеоролика. В результате чтения одного кадра получается матрица размером  $Height \times Width \times 3$ , элементами которой являются 8-ми битовые целые числа без знака. В результате чтения диапазона из нескольких кадров получается матрица размером  $Height \times Width \times 3 \times NumOfFrames$ . Соответственно, когда кадр загружен, то с ним можно работать как с обычной матрицей, а так же применять к нему функции для обработки изображений.

Видеофайлы складываются и вычитаются покадрово. Причем, каждый кадр раскладывается по трем цветам. Полученные матрицы цветов складываются/вычитаются. Значение цвета в ячейке представлено числом от 0 до 255. Если при сложении/вычитании получаем число превышающее 255, то принимается значение цвета в ячейке 255. При получении отрицательного числа, заполняем ячейку нулем. Для проигрывания видеоролика необходимо загрузить его в память, причем ролик должен храниться в памяти в формате отличном от того, который получается при чтении видео. Для этого используется переменная, которая хранит массив из структур. Каждая структура имеет поле для хранения битового образа кадра и поля с цветовой палитрой для видео с индексированным цветом.

Записывать можно несколькими способами: покадрово и одним куском. При покадровой записи необходимо сначала выполнить дилатацию каждого кадра, а затем записать его в файл. При использовании записи одним куском нужно модифицировать массив кадров, а именно, представить его в виде массива структур с полями битового образа и палитры.

Однако, работа со встроенными функциями оказывается неудобной при использовании формата видео с ключевыми кадрами, например, MOV, в этом случае предпочтительным является метод `System Objects`. Системные объекты предоставляют объекто-ориентированную реализацию алгоритмов. Они автоматически управляют своим состоянием и хорошо подходят для потоковой обработки информации. Для покадрового чтения видео с помощью системных объектов создается объект для чтения/записи видео, в него заносится

каждый кадр поочередно. Матрица, которая формируется при данном методе, содержит элементы типа single - четырех байтные числа с плавающей точкой. Для записи видео необходимо предварительно провести дилатацию кадров. Существенный недостаток такого подхода в том, что отсутствует произвольный доступ к видео — читать кадры можно только последовательно. Так же, открытие видео и чтение кадров с помощью системных объектов происходит несколько дольше, по сравнению с использованием mmreader и read. С другой стороны, использование системных объектов позволяет существенно сократить код.

Режим замедления реализуется путем изменения скорости проигрывания видео или количества (частоты) кадров в секунду. Чем она больше, тем более плавным и естественным будет казаться движение. Минимальный показатель, при котором движение будет восприниматься однородным — примерно 16 кадров в секунду.

Иногда при записи видео возникают неподходящие или лишние кадры. Их можно заменить или отбросить. Для точного определения такого кадра предусмотрен покадровый последовательный просмотр. Для замены кадра можно использовать статическое изображение любого формата. Оно будет масштабировано в зависимости от разрешения видео и преобразовано в видео фрагмент. Кроме того, любой кадр фильма может быть сохранен и занесен в буфер обмена.

Таким образом, программное средство «захвата» кадра в видеоряде решает следующие задачи:

- Просмотр видеофильма, выбранного пользователем из буфера обмена, в режимах замедления и покадрово.
- Создание файла, содержащего изображение выделенного в видеофильме кадра и занесение его в буфер обмена.
- Замену кадра, выделенного в видеофильме кадром, выбранным пользователем из буфера обмена.
- Покадровое «сложение» и «вычитание» двух видеофильмов, выбранных пользователем из буфера обмена.

Кроме того, данная программа формирует рабочее место и осуществляет первичный ввод графической информации. Она обеспечивает совместную работу модулей 1-8 с изображениями и видео-файлами, позволяя запускать эти модули для работы с зафиксированными или существующими в буфере медиафайлами. Программное средство предоставляет пользователю возможность именовать для хранения все зарегистрированные изображения, осуществлять просмотр регистрируемых и зарегистрированных файлов.

Программный продукт имеет широкие возможности для последующего расширения за счет простоты внедрения в него других модулей обработки медиа-файлов, что делает его незаменимым для оснащения рабочего места специалиста, занимающимся регистрацией и обработкой цифровых изображений и видео-файлов.

## Литература

1. В. П. Дьяконов Matlab 6.5 SP1/7 + Simulink 5/6. Работа с изображениями и видеопотоками. М.: Солон-Пресс, 2005.- 400 стр.
2. Р. Гонсалес, Р. Вудс, С. Эддинс Цифровая обработка изображений в среде MATLAB. М.: Техносфера, 2006,- 616 стр.
3. Дж. Миано Форматы и алгоритмы сжатия изображений в действии М.: Изд-во Триумф, 2003. - 336 стр.
4. Л. Сиденко Компьютерная графика и геометрическое моделирование М.: Питер, 2009. – 224 стр.

# СТРУКТУРИРОВАНИЕ И ВИЗУАЛИЗАЦИЯ ТЕХНИЧЕСКОГО ЛЕКСИКОНА ИНОСТРАННЫХ СТУДЕНТОВ

Рузанова А.А., Шишкунова Е.В.  
*Научный руководитель : к.т.н. Журавлева Л.В.*  
МГТУ им. Н.Э. Баумана, Москва, Россия

## STRUCTURING AND VIZUALIZATION OF TECNICAL GLOSSARY FOR FOREIGN STUDENTS

Ruzanova A.A., Shiskunova E.V.  
*Ph.D. Juravleva L.V.*  
BMSTU, Moscow, Russia

### Аннотация

В данной работе представлено система формирования и визуализации технического лексикона иностранных студентов, обучающихся в Российских вузах. Визуализированный словарь помогает преодолеть высокий языковой барьер при изучении технических дисциплин, используя различные формы представления информации: текстовую, графическую, видеоряды.

### Abstract

In this paper we present a system of formation and visualization of technical vocabulary of foreign students studying in Russian universities. Visualized dictionary helps to overcome the high language barrier in studying technical subjects, using various forms of information: text, graphics, video sequences.

Интеграция стран Евросоюза в области экономики, безвизовый переезд в Шенгенской зоне, признание дипломов об окончании высшего профессионального образования всеми странами Евросоюза способствовали созданию открытого обучения, которое обеспечивает доступ выпускникам многих стран к образованию во всех европейских ВУЗах и широкую академическую мобильность. Тенденция открытого обучения постепенно распространяется и на другие регионы планеты, однако, этот процесс осложняется языковыми барьерами.

Современные образовательные технологии в системе инженерного образования органически включают широкую академическую мобильность. Под академической мобильностью студентов в узком смысле слова понимают свободный выбор студентом индивидуальной образовательной траектории в рамках выбранного направления подготовки. Формирование студентами индивидуальных учебных планов позволяет удовлетворять специфические потребности рынка труда и отдельных потенциальных работодателей, по согласованию с которыми, и все чаще при их финансовом участии, студентам предоставляется возможность получать одновременно с основным второе высшее образование – так называемая подготовка с *двойной концентрацией*. В интернациональном аспекте это имеет отношение к междисциплинарности.

МГТУ им. Н.Э. Баумана, как и любой другой ВУЗ, сталкивается с проблемой обучения иностранных студентов, недостаточно владеющих русским языком. Особую сложность представляет высокий языковой барьер при изучении технических дисциплин, так как для понимания технических терминов требуется определенный словарный запас, которым иностранные студенты также не владеют. На кафедре ИУ4 «Проектирование и технология производства электронных средств» обучаются студенты из Мьянмы, которые получают второе высшее образование. Для ускорения процесса адаптации студентов Мьянмы к русскоязычной среде и овладения предметной областью был разработан визуализированный словарь технических терминов.

Существующие словари структурируются по алфавитному (рис.1) или тематическому (рис.2) принципу.

<p>дует сбросить со счёта; <b>reißend</b> ~ про- даваться нарасхват (о товарах) <b>abgehoben</b> снятый со счёта <b>abgelagert</b> 1. лежалый, залежавшийся (о товаре) 2. выдержанный, созревший, вылежавшийся (о фруктах, овощах) <b>abgelaufen</b> истекший; просроченный (напр. о векселе) <b>Abgeld</b> <i>n</i> бирж., фин. дивизио <b>abgeleitet</b> производный <b>abgelöst</b> см. <b>abgedeckt</b> <b>abgelten</b> возмещать, компенсировать; вознаграждать; оплачивать; погашать (напр. долг) <b>Abgeltung</b> <i>f</i> возмещение, компенсация; вознаграждение; оплата; погашение (напр. долга) <b>Abgeltungsbetrag</b> <i>m</i> размер возмещения [компенсации] <b>Abgeltungsdarlehen</b> <i>n</i> 1. ссуда для уплаты возмещения [компенсации] 2. ссуда для погашения долга <b>abgenutzt</b> изношенный, использованный <b>abgerechnet</b> 1. за вычетом 2. оплаченный 3. удержанный</p>	<p><b>Abgrenzungskonto</b> <i>n</i> бюджетно-распреде- лительный счёт <b>Abgrenzungsmerkmal</b> <i>n</i> разграничитель- ный признак <b>Abgrenzungsposten</b> <i>m pl</i> бухг. разграничи- тельные статьи баланса, статьи баланса, разграничивающие учёт затрат и поступ- лений между смежными отчётными пе- риодами ~, <b>antizipative</b> антиципативные статьи баланса ~ <b>der Jahresrechnung, aktive</b> статьи годо- вого баланса, разграничивающие затра- ты ~ <b>der Jahresrechnung, passive</b> статьи го- дового баланса, разграничивающие по- ступления ~, <b>transitorische</b> статьи баланса, разграни- чивающие расходы и доходы будущих периодов <b>Abgrenzungssammelkonto</b> <i>n</i> бухг. собира- тельно-распределительный [собиратель- ный] счёт <b>Abgrenzungstag</b> <i>m</i> день разграничения процентов</p>
---	--

Рисунок 1 – Алфавитный принцип структурирования словарей

ПОЯСНЕНИЯ К ТЕКСТУ

1. **He only works five days a week.** *Он работает только пять дней в неделю.* Усилительное наречие **only** в предложении стоит, как правило, перед основным глаголом, независимо от того, какое слово оно усиливает. В устной речи то слово, к которому относится **only**, выделяется ударением.

2. **His working day lasts eight hours.** *Его рабочий день продолжается восемь часов.*  
**He usually finishes work at six o'clock.** *Он обычно кончает работать в шесть часов.*

Слово **hour** означает *час* как меру времени. Выражение **o'clock** употребляется для указания часа в ответ на вопрос: *В каком часу? Который час?* Например:

at five o'clock      в пять часов  
at three o'clock    в три часа

ACTIVE WORDS AND WORD COMBINATIONS

<p><b>third</b> <b>a working day</b> <b>a comrade</b> <b>near</b> <b>far from</b> <b>a long way from</b> <b>to walk</b> <b>a walk</b> <b>to go for a walk</b> <b>to go for walks</b> <b>only</b> <b>Monday</b> <b>every Monday</b> <b>on Monday</b> <b>(on) Monday morning</b> <b>(on) Monday afternoon</b> <b>(on) Monday evening</b> <b>next Monday</b> <b>Tuesday</b> <b>Wednesday</b> <b>Thursday</b> <b>Friday</b> <b>Saturday</b> <b>Sunday</b></p>	<p><b>a week-end</b> <b>at the week-end</b> <b>for the week-end</b> <b>over the week-end</b> <b>to last</b> <b>an hour</b> <b>to receive</b> <b>to get</b> <b>to answer</b> <b>an article</b> <b>afternoon</b> <b>in the afternoon</b> <b>to discuss</b> <b>a question</b> <b>to finish</b> <b>work</b> <b>at</b> <b>at six (o'clock) in the evening</b> <b>hard</b> <b>to stay</b> <b>to return</b> <b>to give back</b> <b>to come (get) back</b> <b>to go back</b> <b>how long</b> <b>What time ...?</b> <b>to stop</b></p>
---	---

Рисунок 2 – Тематический принцип структурирования словарей

Для студентов из Мьянмы технический лексикон формировался самими студентами после прослушивания лекционного материала или выполнения практических работ. Студенты отмечали в конспекте лекций неизвестные слова или термины, которые требовали комментария или объяснения с использованием тех слов, значение которых они знают. Каждое слово формализовалось с помощью шаблона (рис.3), который содержал визуализированное изображение термина и его определение. Кроме того, в шаблоне есть

рабочее поле для самостоятельной работы студента. Термины располагаются в соответствии с темами лекций и в алфавитном порядке.

Поясняющий рисунок	Определение
	Для комментария пользователя

Рисунок 3 – Шаблон для формализации визуализированных терминов

Пример формализации визуализированного технического термина представлен на рис.4.

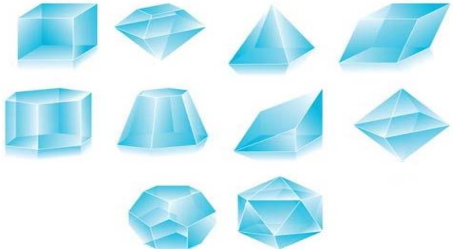
	<b>Геометрическая форма (<u>geometrical shape</u>)</b> детали, конструкции - внешний вид
	Для комментария пользователя

Рисунок 4 – Пример формализации термина "геометрическая форма"

Словарь оформляется на бумажном носителе и постоянно пополняется новыми терминами. Однако, для ускорения поиска необходимых технических терминов, целесообразно разработать электронный словарь.

### Литература

1. Электронный словарь [Электронный ресурс]. Режим доступа: <http://ru.wikipedia.org/wiki/%D0%AD%D0%BB%D0%B5%D0%BA%D1%82%D1%80%D0%BE%D0%BD%D0%BD%D1%8B%D0%B9%D1%81%D0%BB%D0%BE%D0%B2%D0%B0%D1%80%D1%8C> . Проверено 31.01.2011
2. Электронные словари и компьютерная лексикография[Электронный ресурс]. Режим доступа: [http://www.lingvoda.ru/transforum/articles/selegey\\_a1.asp](http://www.lingvoda.ru/transforum/articles/selegey_a1.asp). Проверено 31.01.2011

---

Работа выполнена в рамках реализации фцп «научные и научно-педагогические кадры инновационной россии» на 2009 – 2013 годы ГК №П2333 (НК416).



# ОПТИМИЗАЦИЯ РАЗМЕРА СЕКЦИЙ МНОГОСЕКЦИОННОГО МАНИПУЛЯТОРА ТИПА «ХОБОТ»

Волкоморов С.В.

Научный руководитель д.ф.-м.н., проф., Карпенко А.П.

МГТУ им. Н.Э. Баумана, Москва, Россия.

## SECTIONS SIZE OPTIMIZATION FOR MULTISECTION «TRUNK» MANIPULATOR

S.V. Volkomorov

D. Sc., prof. A.P. Karpenko

MSTU n.a. Bauman, Moscow

### Аннотация

Поставлена задача оптимизации размеров секций многосекционного манипулятора типа «хобот». Рассмотрен подход к ее решению на основе сведения к задаче нелинейного программирования. Сформирован критерий оптимальности и поставлена задача оптимизации. Приводятся результаты численного решения поставленной задачи.

### Abstract

Design task for multisection «trunk» manipulator sections size is posed. Optimization solving approach for it is considered. Optimization criteria is stated and optimization task is posed. Solution is performed.

**Постановка задачи.** «Хобот» манипулятора [1 - 3] представим в виде консольной балки сплошного круглого сечения (рисунок 1), плотность материала которой постоянна и равна  $\rho$ . Вообще говоря, в зависимости от вида нагружения и сечения балки, наряду с нормальными напряжениями существенными могут быть и касательные напряжения. Для балки сплошного сечения последними напряжениями можно пренебречь [4].

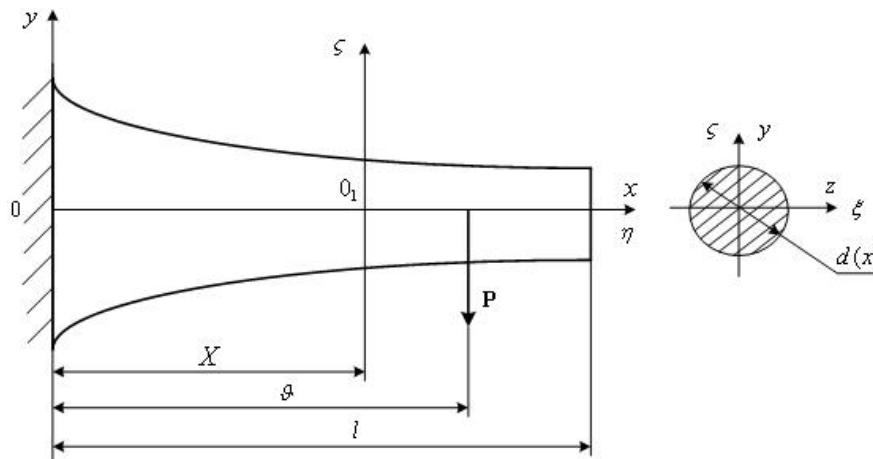


Рисунок 1 - К определению консольной балки равного сопротивления изгибу

Покроем интервал  $[0, l]$  сеткой с узлами  $x_0 = 0, x_1, \dots, x_N = l$ , где  $(N + 1)$  - число узлов сетки. Разобьем консольную балку сечениями, проходящими через узлы указанной сетки и перпендикулярными оси  $Ox$ , на  $N$  секций. Обозначим  $l_i$  длину  $i$ -ой секции, левая граница которой проходит через узел  $x_{i-1}$ , а правая - через узел  $x_i$  (рисунок 2).

Легко видеть, что для  $x$ -координаты начала  $i$ -ой секции имеет место равенство

$$x_{i-1} = \sum_{j=1}^{i-1} l_j, \quad i \in [1 : N]. \quad (1)$$

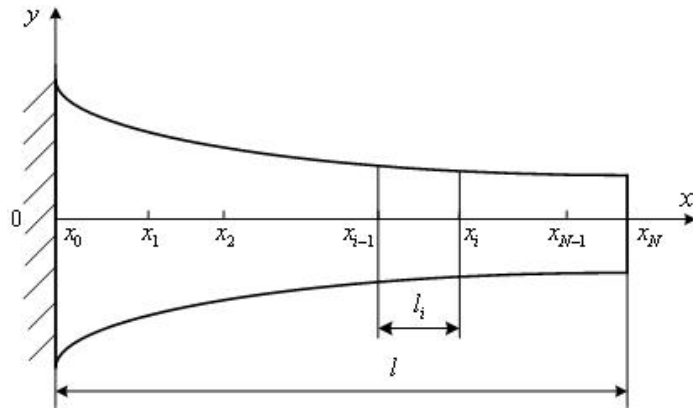


Рисунок 2 – Секции манипулятора

Положим, что упругие свойства материала каждой из секций одинаковы и что податливость (величина, обратная жёсткости) пропорциональна длине этой секции и обратно пропорциональна площади ее левого поперечного сечения [4]. Потребуем, чтобы так определенная податливость всех секций была одинакова, т.е. потребуем выполнения равенств

$$\frac{l_1}{\pi r^2(x_0)} = \frac{l_2}{\pi r^2(x_1)} = \dots = \frac{l_N}{\pi r^2(x_{N-1})} = \bar{b}, \quad \sum_{i=1}^N l_i = l$$

или, что то же самое, равенств

$$\frac{l_1}{r^2(x_0)} = \frac{l_2}{r^2(x_1)} = \dots = \frac{l_N}{r^2(x_{N-1})} = b, \quad \sum_{i=1}^N l_i = l, \quad (2)$$

где  $b = \pi \bar{b}$ , а координаты узлов  $x_i = x_i(l_1, l_2, \dots, l_{i-1})$ ,  $i \in [1 : N]$ ,  $l_0 = 0$  определяются выражением (1).

Система равенств (2) представляет собой систему  $N$  нелинейных уравнений относительно неизвестных  $l_i$ ,  $i \in [1 : N]$ .

**Методы решения задачи.** Аналитическое решение системы уравнений (2) не удастся. Можно предложить несколько подходов к численному решению этой системы. Рассмотрим подходы, основанные на сведении этой задачи к задаче однопараметрической и многопараметрической оптимизации.

*Сведение к задаче однопараметрической оптимизации.* Сформируем критерий оптимальности

$$F_1(b) = \left( l - \sum_{i=1}^N l_i(b) \right)^2$$

и поставим задачу одномерной глобальной условной оптимизации

$$\min_{b>0} F_1(b) = 0. \quad (3)$$

Заметим, что при каждом фиксированном значении параметра  $b$  значение функции  $l_i(b)$ ,  $i \in [1 : N]$  определяется из решения уравнения

$$l_i(b) = b r^2(l_1(b), l_2(b), \dots, l_{i-1}(b)). \quad (4)$$

Поэтому однократное вычисление значения критерия оптимальности  $F_1(b)$  требует решения  $N$  уравнений (4).

Задача (3) может быть решена известными методами решения задач одномерной глобальной условной оптимизации [5].

*Сведение к задаче многопараметрической оптимизации.* Введем критерий оптимальности

$$F_2(L) = \sum_{i=2}^N \left( \frac{l_i}{r^2(l_1, l_2, \dots, l_{i-1})} - \frac{l_{i-1}}{r^2(l_1, l_2, \dots, l_{i-2})} \right)^2,$$

где  $L$  -  $(N \times 1)$ -вектор с компонентами  $l_1, l_2, \dots, l_N$ . Поставим задачу многомерной глобальной условной оптимизации

$$\min_L F_2(L) = 0, \quad L \in D_L, \quad (5)$$

где

$$D_L = \left\{ L \mid l_i > 0, i \in [1 : N]; \sum_{i=1}^N l_i = l \right\}.$$

Однократное вычисление критерия оптимальности  $F_2(L)$  не требует, в отличие от задачи (4), решения вложенной системы уравнений.

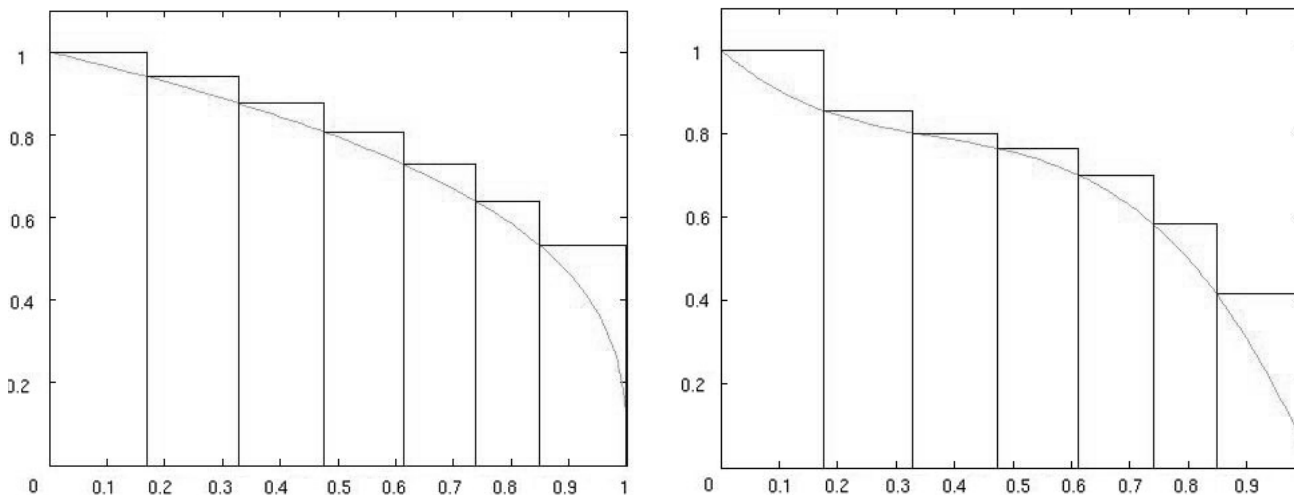
Задача (5) может быть решена методом штрафных функций, где функция штрафа имеет вид

$$U(\Lambda, \mu, L) = \sum_{i=1}^N \lambda_i \bar{l}_i^{-2} + \mu \left( \sum_{i=1}^N l_i - l \right)^2.$$

Здесь  $\Lambda = (\lambda_1, \lambda_2, \dots, \lambda_N)$  - вектор неотрицательных коэффициентов штрафа, отвечающих за выполнение условий  $l_i > 0, i \in [1 : N]$ ;  $\mu$  - также неотрицательный коэффициент штрафа, призванный обеспечить выполнение условия  $\sum_{i=1}^N l_i = l$ ;

$$\bar{l}_i = \begin{cases} 0, & l_i > 0, \\ l_i, & l_i < 0. \end{cases}$$

**Результаты.** Результаты решения задачи методом сведения к задаче многопараметрической оптимизации для семисекционного манипулятора ( $N = 7$ ) иллюстрируют рисунки 3, 4. В качестве формы продольного сечения манипулятора взяты решения, полученные для случая балки равного сопротивления изгибу в работе [6].



а) нагружение сосредоточенной силой  $P = 1$

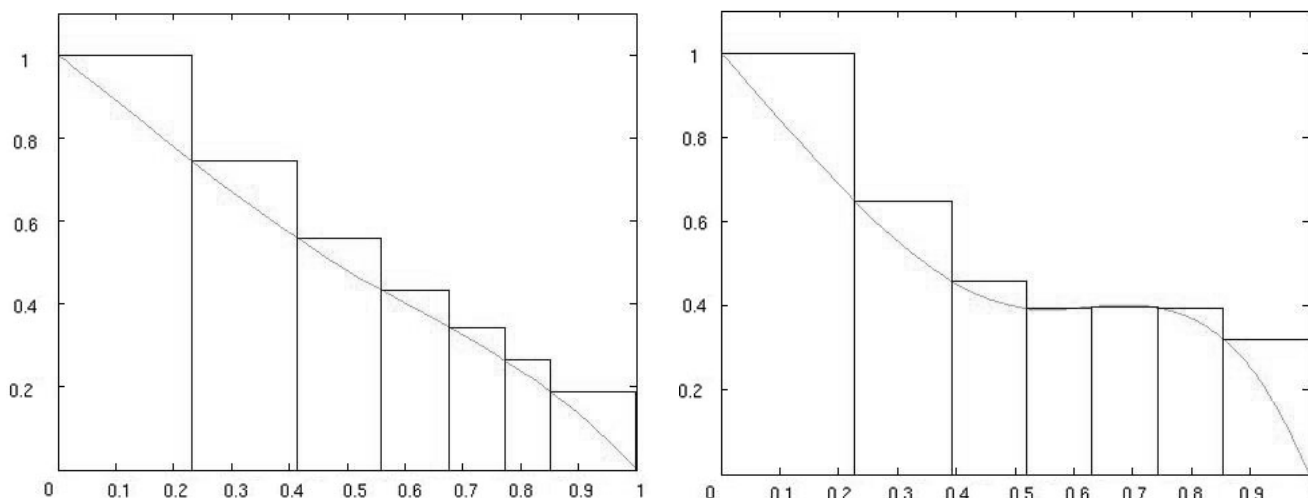
б) нагружение весом балки;  $\rho = 1$

Рисунок 3 - Оптимальные размеры секций манипулятора:  $l = 1.0$ ;  $N = 7$ ; статическая постановка задачи

Для вариантов решения задачи, представленных на рисунках 3, 4 оптимальные относительные длины секций манипулятора указаны ниже.

Вариант 3а: 0,1684; 0,1589; 0,1487; 0,1375; 0,1253; 0,1113; 0,1500;

Вариант 3б: 0,1755; 0,1539; 0,1449; 0,1384; 0,1274; 0,1081; 0,1518;  
 Вариант 4а: 0,2319; 0,1824; 0,1446; 0,1177; 0,0969; 0,0778; 0,1486;  
 Вариант 4б: 0,2280; 0,1643; 0,1263; 0,1114; 0,1122; 0,1118; 0,1459.



в) нагружение моментом инерции балки;

$$\rho = 1; \dot{\varpi} = 1$$

г) нагружение силой лобового  
сопротивления;  $C_A = 1; \varpi = 1$

Рисунок 4 - Оптимальные размеры секций манипулятора:  $l = 1.0$ ;  $N = 7$ ; динамическая постановка задачи

Работа выполнена в рамках аналитической ведомственной целевой программы «Развитие потенциала высшей школы (2009 – 2010 годы)», проект 2.1.2/1509.

### Литература

1. Глазунов В.А., Колискор А.Ш., Крайнев А.Ф. Пространственные механизмы параллельной структуры.- М.: Наука, 1991.- 94 с.
2. Каганов Ю.Т., Карпенко А.П. Математическое моделирование кинематики и динамики робота-манипулятора типа «хобот». 1. Математические модели секции манипулятора, как механизма параллельной кинематики типа «трипод» // Наука и образование: электронное научно-техническое издание, 2009, 10, (<http://technomag.edu.ru/doc/133262.html>).
3. Каганов Ю.Т., Карпенко А.П. Математическое моделирование кинематики и динамики робота-манипулятора типа «хобот». 2. Математические модели секции манипулятора, как механизма параллельной кинематики типа «гексапод» // Наука и образование: электронное научно-техническое издание, 2009, 11, (<http://technomag.edu.ru/doc/133731.html>).
4. Феодосьев В.И. Сопротивление материалов Учебник для вузов.- М.: Изд-во МГТУ им. Н.Э. Баумана, 2004.- 591 с.
5. Грешилов А.А. Математические методы принятия решений.- М.: Изд-во МГТУ им. Н.Э. Баумана, 2006.- 584 с.
6. Волкоморов С.В., Каганов Ю.Т., Карпенко А.П. Моделирование и оптимизация некоторых параллельных механизмов // Информационные технологии, Приложение, 2010, №5, с. 1-32.

# РАЗРАБОТКА ПРОГРАММНОГО СРЕДСТВА ДИНАМИЧЕСКОГО АНАЛИЗА 3D ИЗОБРАЖЕНИЯ С ИСПОЛЬЗОВАНИЕМ СТРУКТУРИРОВАННОЙ ПОДСВЕТКИ В СРЕДЕ MATLAB

Бочаров Ф.А., Савелов А.С.

*Научный руководитель: к.т.н. доцент Волосатова Т.М.*

Кафедра РК6 МГТУ им. Н.Э. Баумана, Россия

## DEVELOPMENT OF SOFTWARE FOR DYNAMIC ANALYSIS OF 3D IMAGES USING STRUCTURED LIGHT IN MATLAB ENVIRONMENT

Bocharov P.A., Savelov A.S.

*The scientific chief: Ph.D. senior lecturer Volosatova T.M.*

Department RK6 BMSTU, Russia

[rhfphd@mail.ru](mailto:rhfphd@mail.ru)

### Аннотация

Данная работа посвящена задаче обработки цифровых изображений. Разработано универсальное программное средство спектрального анализа цифрового изображения, которое позволяет получить и исследовать Фурье- и вейвлет-образы и вести их управляемый просмотр. В данной работе использована среда Matlab для создания графического приложения, выполняющего шифрацию и анализ изображения.

### Abstract

This work is devoted to the problem of digital image processing. A universal software tool of spectral analysis of digital images, which allows you to receive and investigate Fourier and wavelet transforms, and they managed to show. In this paper we used Matlab environment for creating graphical application that performs encryption and image analysis.

MATLAB («Matrix Laboratory») — пакет прикладных программ для решения задач технических вычислений, использующий свой собственный язык программирования. MATLAB используют более 1 000 000 инженерных и научных работников, он работает на большинстве современных операционных систем, включая Linux, Mac OS, и Microsoft Windows.

На сегодняшний день цифровая обработка изображений получила широкое распространение. Промежуточным действием в процессе решения практически каждой задачи в этой области является получение и анализ спектра входного цифрового изображения.

Традиционным математическим аппаратом для получения спектра входного сигнала является преобразование Фурье. Однако, наряду с ним используется и сравнительно недавняя разработка – вейвлет-преобразование. Оно имеет ряд существенных преимуществ по сравнению с преобразованием Фурье, главное из которых – это возможность абсолютно точного представления спектра входного сигнала (без погрешностей).

### Получение Фурье-образа цифрового изображения

Данное программное средство разработано при помощи комплекса математического моделирования Matlab, в рабочем пространстве которого цифровые изображения представляются в виде дискретных отсчетов  $s(k,l)$ . Поэтому для машинного синтеза Фурье-образа применяется математический аппарат дискретного преобразования Фурье (ДПФ). Формула (1) служит для вычисления элементов матрицы Фурье-образа  $\{F(r,s)\}$  по матрице чисел  $\{s(k,l)\}$ :

$$F(r,s) = \sum_{k=0}^{N_x-1} \sum_{l=0}^{N_y-1} s(k,l) \exp\left[i2\pi\left(\frac{kr}{N_x} + \frac{ls}{N_y}\right)\right], \quad (1)$$

где  $N_x$  и  $N_y$  – это количество отсчетов для каждой размерности массива, которое выбирается в соответствии с теоремой Котельникова. Основным методом визуального анализа преобразования Фурье заключается в вычислении его спектра (т.е. абсолютной величины

$F(r, s)$ ) и отображения на экране. В данном средстве пользователю предлагается диаграмма энергетического спектра  $P(r, s)$ , который вычисляется согласно формуле (2):

$$P(r, s) = |F(r, s)|^2 = R^2(r, s) + I^2(r, s) \quad (2)$$

Также средство при необходимости выводит на экран фазовый спектр и изображение, полученное при помощи алгоритма обратного ДПФ.

### Получение вейвлет-образа цифрового изображения

Вейвлеты – это функции типа маленькой волны (всплески), которые порождают базисы пространства  $L^2(R)$ , удобные для обработки сигналов.

Построим в пространстве  $L^2(R)$  ортонормированный базис, который определяется на основе некоторой финитно-определенной функции  $\varphi(x)$ . Рассмотрим систему функций  $\varphi_{j,n}(x)$ , полученную из  $\varphi(x)$  целочисленными сдвигами:

$$\varphi_{j,n}(x) = \sqrt{2^j} \varphi(2^j x - n), \quad (3)$$

Это ортонормированная система функций, все функции системы получаются из  $\sqrt{2^j} \varphi(2^j x)$  сдвигами на  $n/2^j$  по оси  $Ox$ . Имеем бесконечную систему вложенных подпространств  $V_j$ , порожденных системой функций  $\varphi_{j,n}(x)$   $V_j \subset L^2(R)$ :

$$\dots \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \dots \subset V_j \subset \dots$$

Если пространство  $V_0$  образовано сдвигами функции  $\varphi(x)$ , а пространство  $V_1$  - сдвигами  $\varphi(2x - n)$ , то для включения  $V_0 \subset V_1$  необходимо, чтобы  $\varphi(x) \in V_1$ , т.е. чтобы  $\varphi(x)$  могла быть представлена в виде линейной комбинации функций  $\varphi(2x - n)$ . Далеко не каждая функция обладает таким свойством.

Функция  $\varphi(x) \in L^2(R)$  называется масштабирующей, если она может быть представлена в виде:

$$\varphi(x) = \sqrt{2} \sum_{n \in \mathbb{Z}} h_n \varphi(2x - n) \quad (4)$$

Масштабирующая функция Добеши (db1) имеет вид в соответствии с рисунком 1:

Функция удовлетворяет следующему масштабирующему уравнению:

$$\varphi(x) = \frac{1 + \sqrt{3}}{4} \varphi(2x) + \frac{3 + \sqrt{3}}{4} \varphi(2x - 1) + \frac{3 - \sqrt{3}}{4} \varphi(2x - 2) + \frac{1 - \sqrt{3}}{4} \varphi(2x - 3)$$

При этом ее фильтр есть множество:  $\{h_0 = \frac{1 + \sqrt{3}}{4\sqrt{2}}, h_1 = \frac{3 + \sqrt{3}}{4\sqrt{2}}, h_2 = \frac{3 - \sqrt{3}}{4\sqrt{2}}, h_3 = \frac{1 - \sqrt{3}}{4\sqrt{2}}\}$

Для каждого  $j \in \mathbb{Z}$  рассмотрим оператор ортогонального проектирования пространства  $L^2(R)$  на подпространство  $V_j$ :

$$P_j(f) = \sum_{n \in \mathbb{Z}} (f, \varphi_{j,n}) \varphi_{j,n}(x) \quad (5)$$

Если функция  $f(x)$  достаточно хорошая, например, равномерно непрерывная, то ее проекции  $P_j(f)$  будут равномерно приближаться к  $f(x)$ . Поскольку равномерно непрерывные функции плотны в  $L^2(R)$ , а операторы  $P_j : L^2(R) \rightarrow V_j$  непрерывны, то для любого  $f \in L^2(R)$  имеем:

$$P_j(f) \xrightarrow{L^2} f, \text{ при } j \rightarrow \infty \quad (6)$$

Проекция  $P_j(f)$  являются приближениями функции  $f(x)$ , все более точными при увеличении  $j$ . Рассмотрим ортогональное разложение

$$V_{j+1} = V_j \oplus W_j \quad (7)$$

Ортонормированный базис пространства  $W_j$  образуют вейвлет-функции  $\psi_{j,n}(x)$ :

$$\psi_{j,n}(x) = \sqrt{2^j} \psi(2^j x - n) \quad (8)$$

В соответствии с разложением  $V_{j+1} = V_j \oplus W_j$  оператор проектирования  $P_{j+1}: L^2(R) \rightarrow V_{j+1}$  раскладывается в сумму  $P_{j+1}(f) = P_j(f) + P_j^W(f)$ . Первое слагаемое есть более грубое приближение в пространстве  $V_j$ , а второе – отражает детали, которыми  $P_j(f)$  отличается от  $P_{j+1}(f)$ .

В соответствии с разложением  $P_{j+1}(f) = P_j(f) + P_j^W(f)$  на  $(j+1)$ -ом уровне приближения получаем

$$P_{j+1}(f) = \sum_{k \in Z} a_{j,k} \varphi_{j,k}(x) + \sum_{k \in Z} d_{j,k} \psi_{j,k}(x), \quad (9)$$

где  $a_{j,k}$  - коэффициенты аппроксимации  $j$ -го уровня приближения:

$$a_{j,k} = (f, \varphi_{j,k}) = \int_R f(x) \overline{\varphi_{j,k}(x)} dx;$$

$$d_{j,k} - \text{детализирующие коэффициенты: } d_{j,k} = (f, \psi_{j,k}) = \int_R f(x) \overline{\psi_{j,k}(x)} dx.$$

Продолжая до бесконечности, получаем разложение  $V_{j+1}$  на ортонональные подпространства вейвлетов. При  $j \rightarrow \infty$  получается разложение всего пространства  $L^2(R)$  в ортогональную сумму подпространства вейвлетов

В каждом подпространстве  $W_j$  ортонормированный базис образует система  $\{\psi_{j,n}(x)\}$ ,  $n \in Z$ , поэтому вся система функций образует базис пространства  $L^2(R)$ . Обозначим:

$$cA_0 = \{a_{j,n}\}, \quad a_{j,n} = (f, \varphi_{j,n}) = \int_R f(x) \overline{\varphi_{j,n}(x)} dx,$$

$$cA_1 = \{a_{j-1,k}\}, \quad cD_1 = \{d_{j-1,k}\},$$

где  $cA_0$  - коэффициенты аппроксимации глубины разложения 0,  $cA_1, cD_1$  - коэффициенты аппроксимации глубины разложения 1.

По построению коэффициенты  $cA_1$  и  $cD_1$  в совокупности дают ту же информацию, что и первоначальный набор коэффициентов  $cA_0$ . Преимущество коэффициентов  $cA_1$  и  $cD_1$  заключается в том, что они несут еще и качественную информацию. А именно коэффициенты  $cA_1$  описывают более грубое (сглаженное) приближение функции  $f(x)$  в пространстве  $V_{j-1}$ , а коэффициенты  $cD_1$  характеризуют колебания функции  $f(x)$  относительно  $P_{j-1}(f)$ .

Второе преимущество набора коэффициентов  $cA_1$  и  $cD_1$  заключается в том, что данные  $cA_1$ , представляющие сглаженную функцию  $P_{j-1}(f)$ , можно еще раз разложить на еще более сглаженную часть  $P_{j-2}(x)$  и детализирующую часть  $P_{j-2}^W(f)$ . Вейвлет-анализ состоит из следующих шагов:

Выбираем достаточно высокий уровень разрешения  $j$ , такой, что аппроксимация  $P_j(x)$  достаточно точно отражает функцию  $f(x)$ . Практически здесь проблем не возникает, поскольку функция  $f(x)$  обычно задается массивом значений. Эти значения и берутся за начальное приближение  $P_j(f)$ .

Выбираем глубину разложения  $N$  и находим коэффициенты разложения:

$$\{cA_N, cD_N, cD_{N-1}, \dots, cD_1\}$$

Анализируем эти коэффициенты и изменяем их в случае необходимости.

Восстанавливаем функцию  $f(x)$ , используя, возможно измененные, коэффициенты разложения:

$$f(x) = \sum_{k \in Z} a_{j-N,k} \varphi_{j-N,k}(x) + \sum_{k \in Z} d_{j-N,k} \psi_{j-N,k}(x) + \dots + \sum_{k \in Z} d_{j-1,k} \psi_{j-1,k}(x) \quad (10)$$

Алгоритм разложения изображений аналогичен тому, что применяется в одномерном случае. Двумерный сигнал  $s(x, y)$  раскладывается по базисным в  $L^2(R^2)$  функциям  $\varphi_{j,n}(x)\varphi_{j,m}(y)$ ,  $\varphi_{j,n}(x)\psi_{j,m}(y)$ ,  $\psi_{j,n}(x)\varphi_{j,m}(y)$  и  $\psi_{j,n}(x)\psi_{j,m}(y)$ . Соответствующие коэффициенты принято называть следующим образом.

Аппроксимирующие коэффициенты  $cA_j$  получаются как коэффициенты разложения по вейвлет-базису  $\varphi_{j,n}(x)\varphi_{j,m}(y)$ . Горизонтальные детализирующие коэффициенты  $cH_j$  получаются как коэффициенты разложения по вейвлет-базису  $\varphi_{j,n}(x)\psi_{j,m}(y)$ . Вертикальные детализирующие коэффициенты  $cV_j$  получаются как коэффициенты разложения по вейвлет-базису  $\psi_{j,n}(x)\varphi_{j,m}(y)$ . Диагональные детализирующие коэффициенты  $cD_j$  получаются как коэффициенты разложения по вейвлет-базису  $\psi_{j,n}(x)\psi_{j,m}(y)$ .

Практически входной сигнал задается матрицей  $s$ . При разложении этого сигнала получаются указанные выше четыре типа коэффициентов. Схему разложения можно изобразить в виде:

$$C_0 \rightarrow \{cA_1, cH_1, cV_1, cD_1\} \rightarrow \{cA_2, cH_2, cV_2, cD_2, cH_1, cV_1, cD_1\} \rightarrow \dots \quad (11)$$

Поскольку массив начальных коэффициентов двумерный, то более естественно схему разложения сигнала  $C_0$  изобразить в (принятом в Matlab) виде в соответствии с рисунком 2:

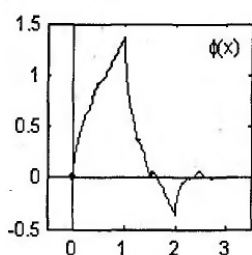


Рисунок 1 - Масштабирующая функция Добеши

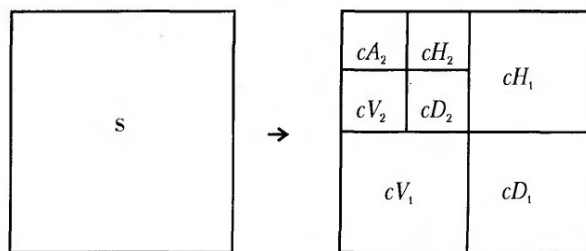


Рисунок 2 - Схема разложения двумерного сигнала

Данное программное средство использует масштабированную функцию Добеши (db1) и позволяет вывести на экран набор масштабированных или детализирующих коэффициентов, а также восстановленное изображение по заданному набору коэффициентов, для уровней разложения от 0 до 2.

### Литература

1. Залманзон Л.А. Преобразование Фурье, Уолша, Хаара и их применение в управлении, связи и других областях. 1989 год. 496 стр.
2. В.П. Дьяконов. Matlab. Полный самоучитель. М.: ДМК Пресс, 2010 768 стр.
3. В.Т. Фисенко, Т.Ю. Фисенко. Компьютерное распознавание и обработка изображений. Уч. пособие. 2008 год. 182 стр
4. Р. Гонсалес, Р. Вудс. Цифровая обработка изображений. 2005 год. 1071 стр.
5. В.Ф. Кравченко, В.Л. Рвачёв «Алгебра логики, атомарные функции и вейвлеты», М.: «Физматлит», 2006;
7. Л.П. Ярославский, Н.С. Мерзляков «Методы цифровой голографии», М.: «Наука», 1977;



# АНАЛИЗ ЭФФЕКТИВНОСТИ АЛГОРИТМОВ ПОИСКА РАЦИОНАЛЬНОГО ВАРИАНТА ПОДКЛЮЧЕНИЯ ВОЗРОСШЕЙ НАГРУЗКИ К ЭНЕРГОСЕТИ

Кузьмина И.А.

*Научный руководитель: д.т.н., профессор, заведующий кафедрой РК-6 И.П. Норенков*  
Московский Государственный Технический Университет им. Н.Э.Баумана, Москва, РФ

## PERFORMANCE ANALYSIS OF SEARCHING RATIONAL VARIANT CONNECTION OF NEW CONSUMERS TO THE ELECTRONETWORK

Kuzmina I.A.

*Scientific adviser: Ph.D, professor, head of department RK-6 I.P. Norenkov*  
Moscow State Technical University named after Bauman, Moscow, Russian Federation

### Аннотация

В данной работе рассмотрены алгоритмы нахождения оптимального количества и места положения трансформаторных подстанций при подключении возросшей нагрузки к сети, алгоритмы поиска варианта подключения возросшей нагрузки к существующей сети энергоснабжения и новым узлам. Произведен сравнительный анализ указанных алгоритмов на примере решения задач различной размерности.

### Abstract

In this paper the algorithms of searching optimal numbers and locations of transformer substation and algorithms of searching connection of increase power consumption to the electronetwork variant are given. Execute the benchmark analysis of these algorithms.

Энергосистемы, как известно, относятся к так называемым большим системам. Планирование развития таких систем, для которых характерна многокритериальность, неполнота информации и неоднозначность решений, требует применения нестандартных и нетривиальных подходов.

Решение данной задачи рационально производить, применив метод декомпозиции – представив исходную задачу в виде ряда подзадач, последовательное решение которых приведет к поиску оптимального решения исходной задачи.

Решение задачи можно представить в виде решения следующих подзадач:

1. Определение оптимального количества и места положения подстанций, строительство которых необходимо для обеспечения возросшей нагрузки; Получение варианта подключения возросшей нагрузки к сети энергоснабжения и новым узлам сети, обеспечивающего критерий минимальной длины кабельных линий и строительства новых подстанций;
2. Нахождение структуры сети энергоснабжения, удовлетворяющей заданному критерию оптимальности.

Указанные подзадачи относятся к классу комбинаторно-оптимизационных задач, то есть для них существует множество допустимых решений, среди которых необходимо отыскать такое, для которого целевая функция достигает оптимального значения.

Кроме того, данные задачи являются NP-трудными, то есть для них не найдено точных алгоритмов поиска решения, работающих за полиномиальное время. Для относительно небольшого количества узлов время поиска точного решения будет приемлемо, однако для реальной задачи применение точных методов не представляется возможным. Решение данного типа задач осуществляется методами поиска приближенного решения.

В рамках данной статьи рассмотрим алгоритмы решения первой подзадачи.

Для решения указанной задачи применим подход, состоящий из двух последовательно реализованных этапов: на первом этапе алгоритма производится попытка подключения возросшей нагрузки к существующим узлам сети; на втором этапе алгоритма принимается решение о количестве и месте возведения новых узлов сети (подстанций), и присоединения к ним узлов нагрузки, подключение которых на первом этапе алгоритма не возможно.

При этом на первом этапе алгоритма подключение возросшей нагрузки производится по принципу минимизации суммарного количества кабельных линий, необходимых для подключения возросшей нагрузки к сети энергоснабжения. Так как строительство и эксплуатация подстанций сети энергоснабжения являются наиболее затратными, на втором этапе решения задачи количество возводимых подстанций также должно быть минимальным.

Для реализации описанных выше процессов были разработаны и реализованы следующие варианты алгоритмов:

**Первый этап:** Последовательное подключение узлов нагрузки с предварительной сортировкой их в порядке возрастания расстояния до ближайшей подстанции.

- 1.1. С попыткой одного варианта подключения узла к сети на первой итерации, без последующей пересортировки;
- 1.2. С попыткой одного варианта подключения узла к сети на первой итерации, с последующей пересортировкой;
- 1.3. С проверкой всех вариантов подключения узла к сети на первой итерации.

Блок-схемы данных алгоритмов приведены на рисунках 1 – 3 соответственно.

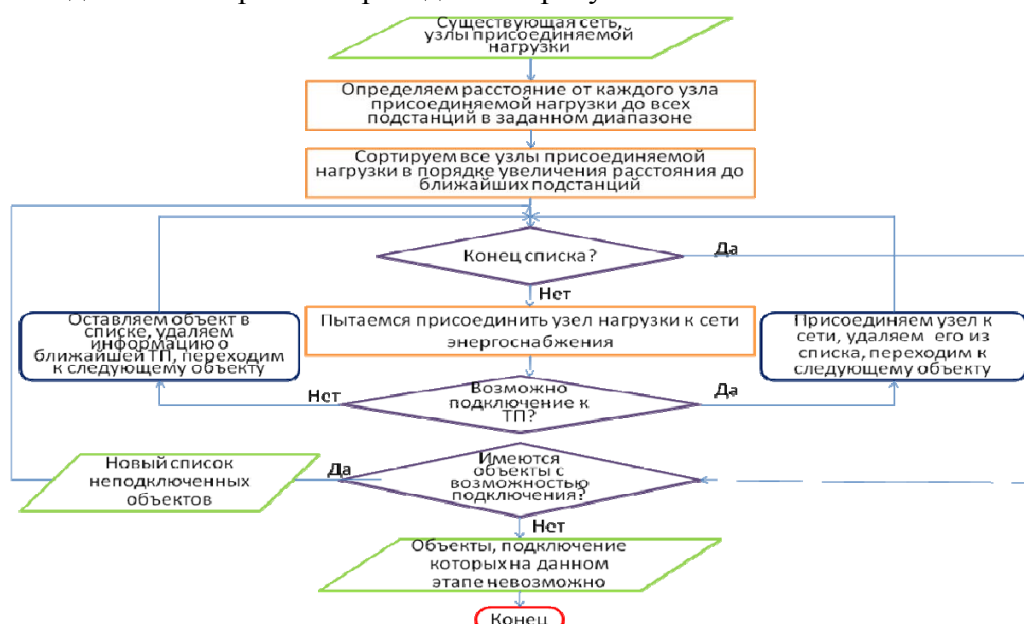


Рисунок 1 - Алгоритм 1.1

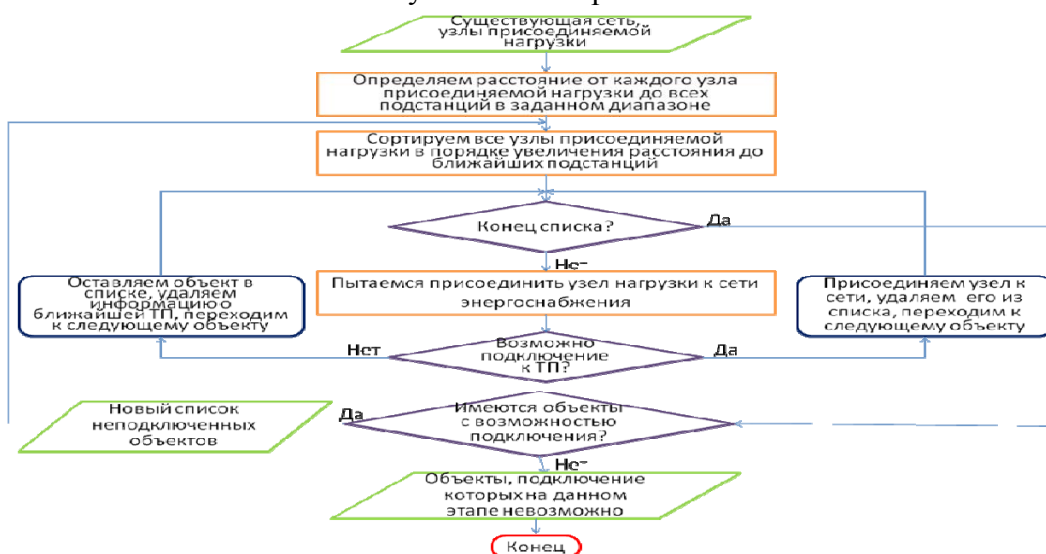


Рисунок 2 - Алгоритм 1.2

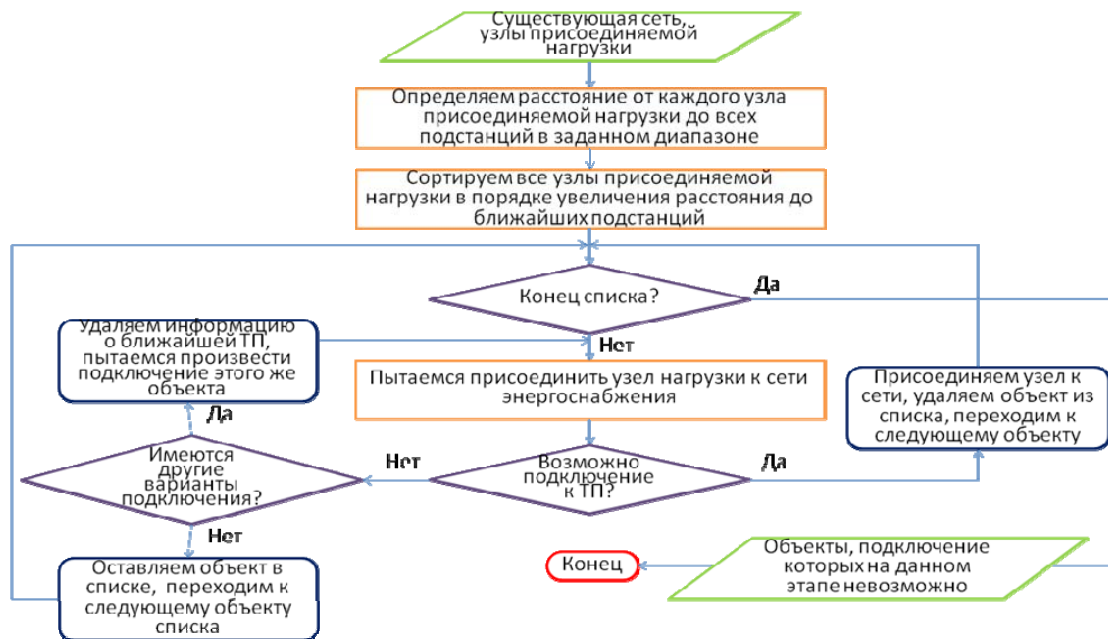


Рисунок 3 - Алгоритм 1.3

**Второй этап:** Определение количества и места возведения новых узлов сети (подстанций), и варианта присоединения к ним узлов нагрузки, подключение которых на первом этапе алгоритма не представляется возможным.

- 1.1. Последовательное возведение подстанций с максимально возможным количеством подключаемых потребителей;
- 1.2. Равномерное разбиение области с предварительным расчетом минимального количества требуемых подстанций.

Блок-схемы данных алгоритмов приведены на рисунках 4 и 5 соответственно.

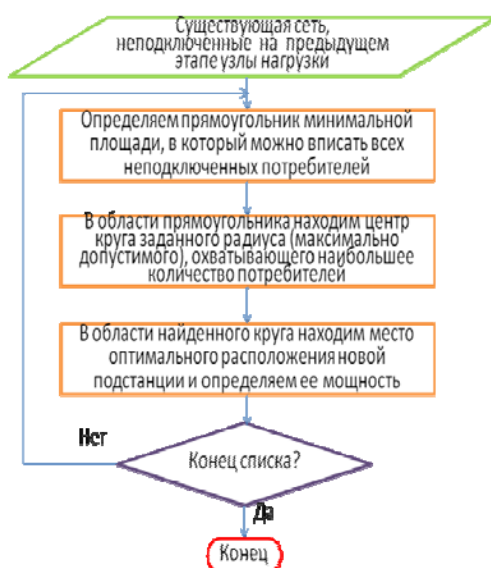


Рисунок 4 - Алгоритм 2.1

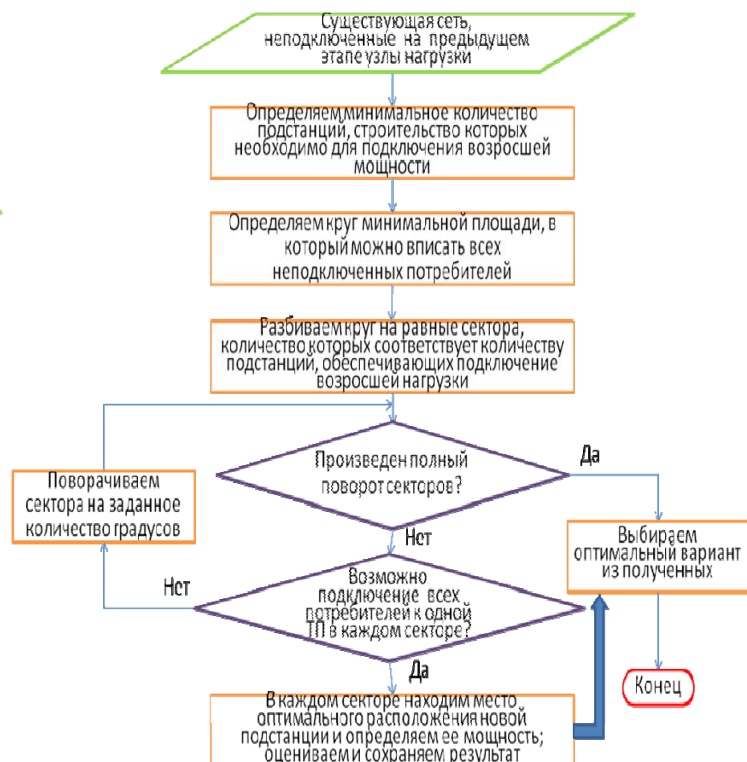


Рисунок 5 - Алгоритм 2.2

Для оценки эффективности шести возможных комбинаций данных алгоритмов были проведены эксперименты на картах размерностью 100, 200, 300, 400 и 500

трансформаторных подстанций. Количество подключаемых вводов было в 2 раза меньше, равно и в два раза больше количества подстанций.

Результаты проведенных экспериментов представлены на рисунке 6.

Для расчета стоимости строительства новых элементов сети примем среднее значение прокладки 1 метра кабельной линии равным 4000 рублей, а строительства новой подстанции – 6 500 000 рублей.

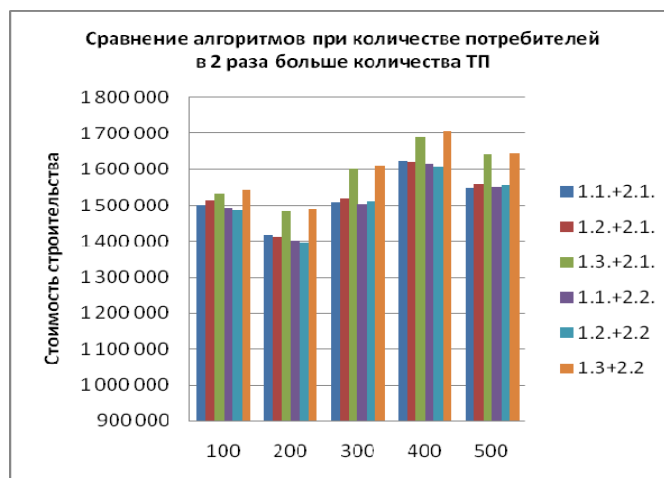


Рисунок 6 - Результаты экспериментальных вычислений

Анализ представленных графиков показывает, что на задачах с небольшим количеством присоединяемых потребителей наилучшие результаты дает комбинация алгоритмов 1.3+2.1 и 1.3+2.2, однако с увеличением числа подключаемой нагрузки, указанные комбинации алгоритмов показывают наихудшие результаты. При этом комбинации алгоритмов 1.2+2.2 показывают наиболее постоянные хорошие результаты.

На основании вышеизложенного можно сделать вывод, что для решения поставленной задачи наиболее оправданным является применение комбинации алгоритмов 1.2+2.2, так как она дает закономерный нехудший результат. Применение иных комбинаций алгоритмов может привести к получению далекого от оптимального результата.

### Литература

1. Оптимизационные задачи электроэнергетики: Учебное пособие. В.Н. Костин – СПб.: СЗТУ, 2003 - 120 с.
2. Теория и практика решения NP-трудной задачи структурного синтеза программными средствами. А.В. Анисимов, Т.Н. Романова – журнал «Вестник МАИ», №2, 2009г.

# АВТОМАТИЗАЦИЯ ПОСТРОЕНИЯ СЕЧЕНИЙ ЛОПАСТЕЙ КОМПРЕССОРОВ В СИСТЕМЕ UNIGRAPHICS

Грушин М.А.

Научный руководитель: к.т.н., доцент, Мартынюк В.А.  
МГТУ им. Н.Э.Баумана, кафедра РК-6, Москва, Россия

## AUTOMATION OF COMPRESSOR BLADES CROSS-SECTIONS CREATING IN THE UNIGRAPHICS SYSTEM

M.A. Grushin

Ph.D., associate professor, V.A. Martynjuk  
MSTU n.a. N.E.Bauman, RK-6 Department, Moscow, Russia

### Аннотация

В практике построения твердотельных моделей лопастей компрессоров по их сечениям зачастую большое количество времени отнимает процесс получения требуемых сечений. В докладе предложен один из возможных подходов к автоматизации построения сечений, основанный на применении кривых Безье, а также процесс его реализации для системы Unigraphics NX 2.

### Abstract

In compressor blades solid models reception practice the demand cross-section creation process is taking a considerable amount of time. This article offers one of possible approaches to automation of cross-section creation process based on Bezier curves application, and also process of its realization for Unigraphics NX 2 system.

В практике моделирования лопаточных венцов компрессоров газотурбинных двигателей (ГТД) построение твердотельной модели пера лопатки обычно ведётся по её плоским сечениям. При этом сечения получают путём изменения геометрии выбранного стандартного исходного симметричного профиля в соответствии с газодинамическими и геометрическими требованиями, предъявляемыми к лопатке (рис. 1.1). Плоский исходный профиль в свою очередь представлен координатами точек, обмерянных с поверхности его реального прототипа.

Получение сечения сводится к вычислению новых значений координат точек исходного профиля, полученных после всех геометрических преобразований.

После получения координат всех точек сечения начинается этап построения сечения в среде САПР. В данной работе построение ведётся в среде Unigraphics NX. По полученным точкам строится сплайн, затем он сопрягается с окружностями входных и выходных кромок.

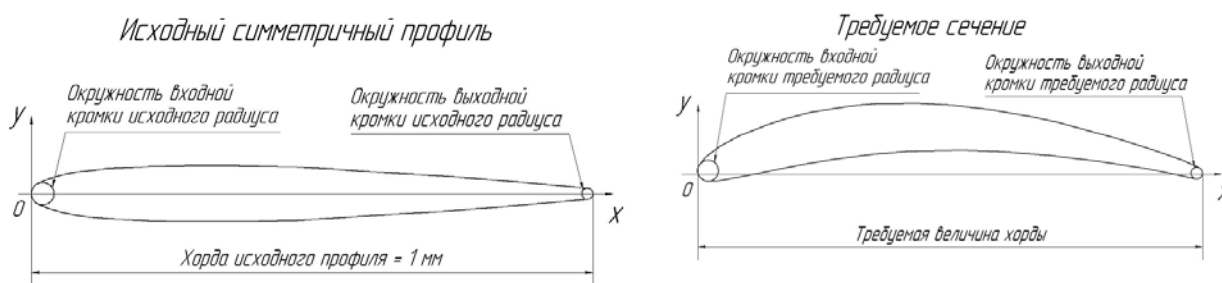


Рисунок 1.1 - Получение требуемого сечения путём изменения геометрии исходного

Инструментальные погрешности при обмере точек исходного профиля приводят к тому, что сплайн, впоследствии проведённый через эти точки, не является гладким и имеет локальные неровности и перегибы, особенно заметные в области сопряжения с кромками на входе и выходе. По этой причине в ходе моделирования лопатки довольно много времени уходит на «выглаживание» сплайна и построение промежуточных кривых, сопрягающих поверхность лопатки с окружностями входной и выходной кромок.

В таком случае приемлемым решением проблемы является аппроксимация точек исходного симметричного профиля гладкой кривой невысокой степени. В данной работе рассмотрен метод аппроксимации с помощью кривой Безье третьего порядка. Сам процесс аппроксимации сводится к применению методов оптимального поиска, в данном случае метода градиентного спуска, для минимизации функции отклонения кривой от аппроксимируемых точек. Метод, использованный для аппроксимации, гарантирует отсутствие перегибов и локальных осцилляций кривой, а также гладкость сопряжения с окружностями входной и выходной кромок.

### Кривые Безье третьего порядка

В параметрической форме кубическая кривая Безье описывается следующим уравнением:  $\mathbf{B}(t) = (1-t)^3\mathbf{P}_0 + 3t(1-t)^2\mathbf{P}_1 + 3t^2(1-t)\mathbf{P}_2 + t^3\mathbf{P}_3$ ,  $t \in [0, 1]$ .

Четыре опорные точки  $P_0, P_1, P_2$  и  $P_3$ , заданные в 2-х или 3-мерном пространстве определяют форму кривой.

Кроме гладкости кривой третьего порядка кривые Безье обладают следующими полезными свойствами.

1. Основа формы кривой повторяет очертания определяющего многоугольника.
2. Первая и последняя точки совпадают с соответствующими точками определяющего многоугольника.
3. Векторы касательных к кривой в первой и последней точках по направлению совпадают соответственно с первой и последней сторонами многоугольника.

При известном радиусе входной и выходной кромок (соответственно  $R$  и  $r$  на рис.2.2) и известных углах сопряжения поверхности профиля с ними ( $\alpha_1$  и  $\alpha_2$  на рис. 2.2) можно точно определить координаты лежащих на соответствующих окружностях начальной и конечной точек кривой Безье. С ними совпадут точки  $P_0$  и  $P_3$  определяющего многоугольника (в соответствии со свойством 2). В этих точках кривая будет сопрягаться с окружностями. Но для гладкого сопряжения необходимо достичь непрерывности по касанию. Таким образом, совпасть должны не только координаты точек на окружностях и кривой Безье, но и направления векторов первой производной в них. Анализируя свойство 3, приходим к выводу, что за направление вектора первой производной в начальной и конечной точках кривой отвечают соседние с ними точки  $P_1$  и  $P_2$ . Направление вектора первой производной в точке на окружности нам известно – это направление прямой, касательной к окружности в этой точке. Значит, для обеспечения равенства направлений векторов опорные точки  $P_1$  и  $P_2$  должны лежать на соответствующих касательных прямых  $l_1$  и  $l_2$  (см. рис. 2.1). Более того, так как поверхность профиля должна быть выпуклой и не иметь перегибов, то в соответствии со свойством 1, определяющий многоугольник также должен быть выпуклым, и эти точки должны лежать выше точек  $P_0$  и  $P_3$ .

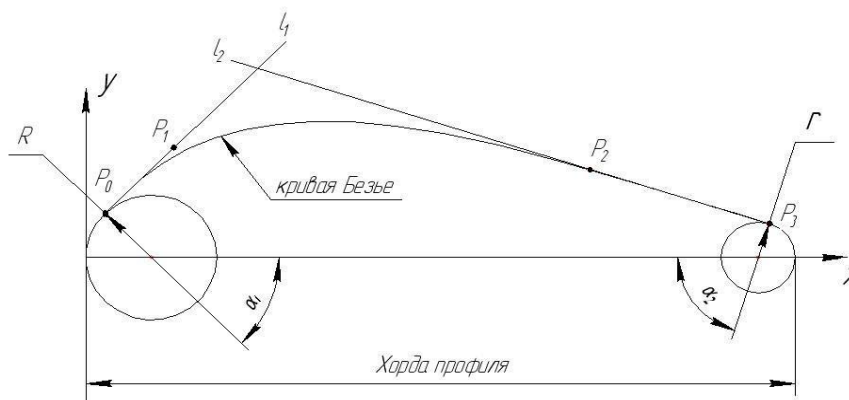


Рисунок 2.1 - Поверхность исходного профиля

Итак, на точки наложены все необходимые ограничения для обеспечения нужной формы кривой. Следующий этап – аппроксимация точек исходного профиля.

## Задача оптимизации

Как нам уже известно, форму кривой целиком определяют её опорные точки, следовательно, задача получения кривой определённой формы сводится к вычислению координат опорных точек, при которых такая форма будет получена. Если установить зависимость отклонения кривой от аппроксимируемых точек от координат её опорных точек, то с помощью методов оптимизации можно определить такие значения координат опорных точек, при которых отклонение будет иметь минимальную величину.

Целевая функция отклонения кривой от точек имеет вид  $E = \frac{1}{2} \sum_{i=1}^N (y_i - y_{bi})^2$ , где  $N$  – количество исходных точек,  $y_i$  – ордината  $i$ -ой исходной точки,  $y_{bi}$  – ордината соответствующей точки на кривой Безье.

Отметим, что только координаты точек  $P_1$  и  $P_2$  доступны для варьирования в целях приближения. Также отметим, что эти точки всегда лежат на прямых, касательных к окружностям в точках сопряжения с кривой. Тогда координаты точек  $P_1$  и  $P_2$  могут быть получены из параметрических уравнений этих прямых как точки, принадлежащие им и заключённые между точками касания с окружностями и точкой пересечения прямых. Следовательно, целевую функцию можно представить как функцию от параметров касательных прямых  $t_1$  и  $t_2$ .

Для вычисления значения целевой функции каждой  $i$ -ой исходной точки определяется такая точка на кривой Безье, которая имеет равную с ней абсциссу, затем вычисляется разность их ординат. За выбор точек на кривой отвечает параметр  $t$ . Так как для каждого  $i$  нам известна абсцисса  $B_x$ , то параметр  $t$  определяется решением кубического уравнения кривой Безье. Для получения аналитического решения уравнения третьей степени воспользуемся формулами Кардано.

Итак, теперь можно однозначно определить значение параметра  $t$ , которому будет соответствовать точка с координатой  $x_i$ . В таком случае уравнение целевой функции

$$\text{окончательно принимает вид } E(t_1, t_2) = \frac{1}{2} \sum_{i=1}^N (B_{yi}(t_1, t_2) - y_i)^2.$$

### Оптимизация целевой функции

В данной работе для поиска минимума выбран градиентный метод оптимизации с дроблением шага.

Суть градиентного метода состоит в итерационном изменении значений варьируемых параметров функции в антиградиентном направлении для поиска минимума. На каждой итерации значение параметра вычисляется по формуле  $X^{r+1} = X^r + \lambda^r S^r$ , где  $X^r$  – текущее значение параметра,  $\lambda^r$  – текущее значение шага в направлении  $S^r$ ,  $S^r$  – вектор направления антиградиента целевой функции в точке  $X^r$ .

Итерации повторяются до достижения заданной точности, т. е. достижения целевой функцией определённого значения, либо до тех пор, пока изменение значения функции по отношению к предыдущему значению не будет меньше некоторого значения  $\varepsilon$ .

Итого, после достижения оптимизационным алгоритмом заданной величины отклонения, опорные точки расположены таким образом, что кривая имеет наименьшее достижимое отклонение от точек исходного профиля. При практическом применении алгоритму оптимизации требуется в среднем 600 итераций на достижение среднего отклонения от точек, равного 0.001 мм.

## Создание модуля пользовательского приложения в среде NX

Процесс построения сечения в среде САПР можно полностью автоматизировать вместо построения вручную по точкам, полученным из аппроксимации. Сам процесс аппроксимации также можно интегрировать в модуль построения сечения. От пользователя потребуется только задать требуемые параметры сечения. Для решения подобных задач Unigraphics располагает достаточно развитыми возможностями модуля UG/Open API (Application Program Interface).

UG/Open API (User Function) реализует взаимодействие Unigraphics и программ пользователя, написанных на языке C/C++. Заголовочные файлы (.h) соответствуют требованиям стандарта ANSI C и поддерживают разработку программ с использованием языков C/C++. Файл с программным кодом создается в среде Microsoft Visual C++ с использованием шаблона для создания модулей Unigraphics NX. Шаблон поставляется вместе с дистрибутивом программы NX. Документация по функциям библиотек UG/Open поставляется с дистрибутивом Unigraphics.

### Заключение

Рассмотренный метод аппроксимации кривыми Безье третьей степени показал хорошие результаты при практическом применении. Простота модификации и малое количество вычислений для построения, а также инвариантность аффинных преобразований делают этот тип кривых довольно удобным инструментом при решении задач аппроксимации с наложенными ограничениями на форму кривой. Точность аппроксимации можно повысить, применяя кривые более высоких степеней и более точные методы оптимального поиска.

Автоматизация процессов построения сечений средствами пользовательских модулей Unigraphics позволяет сократить время моделирования лопаток и повысить эффективность труда инженера-конструктора.

### Литература

1. М. Краснов, Ю. Чигишев. Unigraphics для профессионалов. – М: Лори, 2004. - 320с.
2. Роджерс Д., Адамс Дж. Математические основы машинной графики, 2-е издание. – М.: МИР, 2001. – 604 с.
3. Карпенко А.П. Методы оптимизации. <http://bigor.bmstu.ru> МГТУ им. Н.Э. Баумана. 2006. URL <http://bigor.bmstu.ru/?cnt/?doc=МО/ch0701.mod/?cou=МО/base.cou> (дата обращения 15.03.10)



# ПОСТРОЕНИЕ ГРАНИЦ ОБЛАСТИ ДОСТИЖИМОСТИ ЛЕТАТЕЛЬНОГО АППАРАТА НА ГРАФИЧЕСКОМ ПРОЦЕССОРЕ

Домашнев В.К.

д.ф.-м.н., проф. Карпенко А.П.

МГТУ им. Н.Э.Баумана, Москва, Россия

## MODELING ATTAINABILITY DOMAIN OF AIRCRAFT USING GRAPHICS PROCESSING UNIT

Domashnev V.K.

PhD, prof. Karpenko A.P.

BMSTU, Moscow, Russia

### Аннотация

Рассматривается использование технологии *CUDA* при построении области достижимости летательного аппарата.

### Abstract

This article describes application of *CUDA* architecture for modeling attainability domain of dynamic system (aircraft).

### Введение

В настоящее время при использовании моделей объектов в виде обыкновенных дифференциальных уравнений (ОДУ) часто возникает задача построения области достижимости соответствующей динамической системы. Эту задачу часто приходится решать в режиме реального времени, поэтому для построения этой области используют численные методы, т.е. строят некоторую дискретную аппроксимацию области достижимости или ее границ.

Графические процессоры являются мощными вычислительными системами, изначально рассчитанными на выполнения огромного числа микроопераций. Они могут быть использованы для решения целого ряда сложных вычислительных задач.

*CUDA* (*Compute Unified Device Architecture*) представляет собой технологию *GPGPU* (*General-Purpose computing on Graphics Processing Units*), которая позволяет реализовывать на упрощённом языке программирования Си алгоритмы, выполнимые на графических процессорных устройствах (ГПУ) *GeForce* восьмого и старшего поколений, а также на ГПУ *Nvidia Quadro* и *Tesla* компании *Nvidia*. Технология разработана компанией *nVidia*.

**Постановка задачи.** Рассмотрим летательный аппарат, движение центра масс которого в земной неподвижной системе координат  $Oxyz$  описывается системой нелинейных дифференциальных уравнений [1].

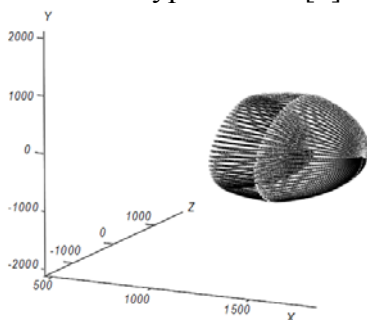


Рисунок 1 - Дискретная аппроксимация области достижимости

$$\begin{cases} \dot{v} = g \cdot (n_T - \sin \Theta), \\ \dot{\Theta} = \frac{g}{v} (n \cos \gamma - \cos \Theta), \\ \dot{\Psi} = -gn \cdot \sin \gamma / v \cdot \cos \Theta, \\ \dot{x} = v \cdot \cos \Theta \cdot \cos \Psi, \\ \dot{y} = v \cdot \sin \Theta, \\ \dot{z} = -v \cdot \cos \Theta \cdot \sin \Psi. \end{cases} \quad (1)$$

Обозначим  $X$  - шестимерный вектор, компонентами которого являются фазовые переменные системы (1):  $X = (v, \Theta, \Psi, x, y, z)$ . Движение летального аппарата будем рассматривать на интервале времени  $[0, T]$ .

Областью достижимости  $D_X = D_X(T, X^0)$  динамической системы (1) назовем множество всех возможных значений вектора  $X(T)$ , которые достигаются на решениях данной системы при начальных условиях  $X^0$  и выполнении следующих ограничений на управления:

$$u_T^{\min} \leq u_T \leq u_T^{\max}; \quad |u| \leq u^{\max}, \quad |\gamma| \leq \pi.$$

Требуется построить границы область достижимости  $D_Y$  при следующих допущениях.

- 1) Имеет место плоское движение ЛА, так что  $\gamma = const$ .
- 2) В области достижимости летательного аппарата (ОДЛА) отсутствуют препятствия.
- 3) Известна структура оптимальных управлений  $u_1, u_2, \gamma$ , приводящих летательный аппарат на границы ОДЛА [1]:

- дальняя граница  $\Omega_f$  достигается при управлениях

$$u_T = const = 1, \quad u = \begin{cases} \pm 1, & t \leq \tau_f, \\ 0, & t > \tau_f; \end{cases} \quad (2)$$

- ближняя граница  $\Omega_n$  - при управлениях

$$u_T = const = -1, \quad u = \begin{cases} \pm 1, & t \leq \tau_n, \\ 0, & t > \tau_n; \end{cases} \quad (3)$$

- боковая граница  $\Omega_s$  - при управлениях

$$u_T = \begin{cases} -1, & t \leq \tau_s, \\ 1, & t > \tau_s; \end{cases} \quad u = const = 1. \quad (4)$$

### Метод мультифиниша

Для приближенного построения границ ОДЛА используем метод мультифиниша [2]. Запишем систему (1) в векторной форме

$$\dot{X} = F(X, U), \quad X(0) = X^0, \quad t \in [0, T], \quad (5)$$

где  $U = (u_1 = u_T, u_2 = u, u_3 = \gamma) \in D_U$  - вектор управления, а  $D_U$  - множество допустимых управлений, которое формируют ограничения (2) – (4).

Покроем множество  $D_U$  некоторой сеткой с узлами

$$U_j = U_j(t) = \{u_{i,j}(t), i \in [1:m], j \in [1:M]\},$$

где  $m = 3$  - размерность вектора управлений,  $M$  - общее число узлов сетки. Поставим в соответствие системе (5) совокупность  $M$  систем ОДУ с указанными управлениями

$$\left\{ \begin{array}{l} \dot{X}_1 = F(t, X_1, U_1), X_1(0) = X^0, \\ \dots \\ \dot{X}_M = F(t, X_M, U_M), X_M(0) = X^0. \end{array} \right. \quad (6)$$

Метод мультифиниша заключается в отыскании множества точек  $\{X_j(T), j \in [1:M]\}$ , представляющих собой дискретную аппроксимацию границы области  $D_X$ , путем интегрирования совокупности систем ОДУ (6).

## Схема распараллеливания вычислений

Общая схема распараллеливания вычислений при построении границы ОДЛА методом мультифиниша на  $N$ -процессорной вычислительной системе имеет вид:

- 1) процессор  $P_i$ ,  $i \in [1:N]$  получает от *host*-процессора векторы управлений  $U_{(i-1)r+1}, U_{(i-1)r+2}, \dots, U_{ir}$ ;
- 2) интегрирует при каждом из этих управлений систему ОДУ (5) - находит координаты множества точек  $\{X_j(T), j \in [(i-1)r+1:ir]\}$ ;
- 3) передает координаты полученных точек *host* - процессору;

Здесь  $r = \left\lceil \frac{M}{N} \right\rceil$  - число векторов управления, обрабатываемых процессором  $P_i$ ;  $\lceil * \rceil$  - символ ближайшего целого большего.

Поскольку графический процессор *NVidia* является *SIMD*-системой, целесообразно воспользоваться равномерной декомпозицией точек переключения [2, 3]. Рассмотрим схему распараллеливания на примере задачи построения дальней границы ОДЛА.

Покроем интервал  $[0, T]$  равномерной сеткой с шагом  $\Delta t = \frac{T}{M}$  и узлами  $t_j$ ,  $j \in [1:M]$ .

Положим, что шаг  $\Delta t$  кратен шагу, с которым осуществляется интегрирование системы ОДУ (5). Пусть также моменты времени  $\tau_{f,j}$  переключения управления  $u$  совпадают с узлами указанной сетки.

В принятых соглашениях естественно принять, что  $U_j = (u_T, u(t, \tau_{f,j}), \gamma)$ . При этом процессор  $P_1$  выполняет интегрирование системы ОДУ (1) при управлениях  $U_1, \dots, U_r$ , на процессоре  $P_2$  - при управлениях  $U_{r+1}, \dots, U_{2r}$  и т.д. до процессора  $P_N$ , который выполняет интегрирование при управлениях  $U_{(N-1)r+1}, \dots, U_M$ . Вычисления на процессоре  $P_i$  организуем по следующей схеме.

*Шаг 0* (этап «разгона»). Исходя из начальных условий  $X^0$ , выполняем интегрирование системы (5) при управлении  $(1, 1, \gamma)$  от момента времени 0 до момента времени  $t_{ir-1}$  и сохраняем в памяти процессора  $P_i$  значения компонентов векторов  $X(t_{(i-1)r+1}) = X^0_{(i-1)r+1}$ ,  $X(t_{(i-1)r+2}) = X^0_{(i-1)r+2}, \dots$ ,  $X(t_{ir-1}) = X^0_{ir-1}$ .

*Шаг 1*. Исходя из начальных условий  $X^0_{(i-1)r+1}$ , выполняем интегрирование системы (5) при управлении  $(1, -1, \gamma)$  от момента времени  $t_{(i-1)r+1}$  до момента времени  $T$ .

*Шаг 2*. Исходя из начальных условий  $X^0_{(i-1)r+2}$ , выполняем интегрирование системы (5) при управлении  $(1, -1, \gamma)$  от момента времени  $t_{(i-1)r+2}$  до момента времени  $T$ .

...

*Шаг  $r$* . Исходя из начальных условий  $X^0_{ir-1}$ , выполняем интегрирование системы (5) при управлении  $(1, -1, \gamma)$  от момента времени  $t_{ir-1}$  до момента времени  $T$ .

Поскольку *CUDA* осуществляет динамическую балансировку загрузки процессоров ГПУ, нити вычислительного процесса целесообразно проектировать максимально легковесными [4]. Поэтому каждой нити поставлено в соответствие одно значение управления  $\gamma$ . Оптимальное число точек переключения  $t_i, \dots, t_{iL-1}$  управления  $u$ , назначаемых каждой нити, экспериментальным образом подбирается таким образом, чтобы обеспечить максимально ускорение вычислений.

## Результаты исследования

Исследование выполнено при следующих значениях величин  $u_T^{\min}$ ,  $u_T^{\max}$ ,  $u^{\max}$ , определяющих множество допустимых управлений:

$$u_T^{\min} = -1.6, \quad u_T^{\max} = 0.6; \quad u^{\max} = 8.$$

Начальные условия системы ОДУ (1), (2) приняты равными

$$v(0) = 330,0 \text{ м/с}, \quad \Theta(0) = 0, \quad \psi(0) = 0, \quad x(0) = 0 \text{ м}, \quad y(0) = 10000 \text{ м}, \quad z(0) = 0 \text{ м}.$$

Длительность рассматриваемого интервала времени  $[0, T]$  равна 5 с.

Вид соответствующей области достижимости иллюстрирует рисунок 1. ОДЛА на рисунке получена для случая, когда сетка содержит 128 точек по компоненте управления  $\gamma$  и 2048 точек – по компоненте  $u$ .

Выполнено исследование эффективности параллельного построения ОДЛА на ГПУ. В качестве меры эффективности использовано ускорение – отношение времени решения задачи на *host*-процессоре ко времени решения задачи на ГПУ.

*Host*-процессор использованной вычислительной системы представляет собой персональный компьютер со следующими значениями основных параметров: *CPU Intel® Core™2Duo E4500*, 2.21 ГЦ; 2 ГБ ОЗУ.

В качестве ГПУ использована видеокарта *NVidia GeForce 8800GT*, имеющая 14 *SIMD*-мультипроцессоров. Мультипроцессор состоит из 8 суперскалярных потоковых процессоров, каждый из которых имеет 8КБ 32х-битных регистров. Мультипроцессор включает в себя модуль разделяемой памяти, объем которой составляет 16КБ.

Число точек переключения управления  $u$ , приходящихся на одну нить, варьировалось от двух до 16. Экспериментальным путем установлено, что максимальное быстроедействие при решении задачи на указанном ГПУ достигается в ситуации, когда нити соответствует четыре точки переключения указанного управления. При этом время решения задачи составило 0,953 секунды, в то время как время последовательного решения равно 330,031 секунды. Таким образом, достигнуто ускорение, равное  $\sim 348$ .

Авторы благодарят Воронова Е.М. и Карпунина А.А. за постановку задачи и многочисленные плодотворные обсуждения подходов к ее решению.

## Литература

1. Е.М. Воронов, А.А. Карпунин. Алгоритм оценки границ области достижимости летательного аппарата с учетом тяги // Вестник МГТУ. Сер. Приборостроение.- 2007.- №4(69).- С. 81-99.
2. Е.М. Воронов, А.П. Карпенко, О.Г. Козлова, В.А. Федин. Численные методы построения области достижимости динамической системы // Наука и образование: электронное научно-техническое издание, 2010, №1. (<http://technomag.edu.ru>).
3. Е.М. Воронов, А.П. Карпенко, В.А. Федин. Параллельное построение множества достижимости высокоманевренного летательного аппарата методом «мультифиниша» // Параллельные вычислительные технологии (ПаВТ'2010): Труды международной научной конференции. – Челябинск: Издательский центр ЮУрГУ, 2010. – с. 113-120.
4. А. Адинец, Вл.В Воеводин. Графический вызов суперкомпьютерам // Открытые системы. -2008. -№ 4. -С. 35–41.

## ИСПОЛЬЗОВАНИЕ ВОЗМОЖНОСТЕЙ ГРАФИЧЕСКИХ СИСТЕМ ДЛЯ ОТРАБОТКИ НОВЫХ УПРАЖНЕНИЙ СПОРТСМЕНА

**Дроздов К.В.**

*Научный руководитель: Доцент кафедры РК6 , к.т.н. Мартынюк В.А.*  
МГТУ им. Баумана кафедра РК6, Москва, Россия

**DrozdoV K.V.**

*Associate Professor, PhD, lecturer RK6 Martyniuk V.A.*  
MSTU n.a. Bauman, Moscow, Russia

### **Аннотация**

Рассматривается использование графических систем для моделирования комплексов физических упражнений.

Известно, что тяжелые графические системы (CATIA, UNIGRAPHICS, Pro-Engineer), и современные системы PDM и PDL (ENOVIA, DELMIA и др.) имеют в своем составе различные средства, позволяющие имитировать поведение рабочих исполнителей или пилотов на своих рабочих местах (на конвейере, у станков, в кабине самолета, в салоне автомобиля и пр.). Такая имитация позволяет, например, полнее определить эргономические характеристики спроектированного рабочего места пилота, или наглядно продемонстрировать возникновение возможных очередей на обработку в процессе сборки на конвейере и т.д. Эти средства стали уже достаточно привычными в системах CAD и, как правило, сведены в разделе проектирования эргономики будущего рабочего места исполнителя в проектируемом изделии.

Если таким рабочим местом является, например, кабина самолета, то в этом случае в перечень эргономических вопросов входит расчет углов обзора пилота, доступность различных кнопок и рычагов управления, освещенность рабочего места, возможные позы, которые принимает пилот, различные нагрузки, которые он испытывает во время полета и многое другое. Естественно, что предоставляемые инструменты для отработки подобных вопросов включают в себя и возможность имитировать достаточно сложные перемещения пилота. Вплоть до возможности имитировать движения любого из его пальцев на каждой руке. В общем, для имитации движения исполнителя на его рабочем месте в перечисленных выше графических системах средств – более чем достаточно.

Поэтому оказалось возможным применить этот аппарат для имитации гораздо более сложных движений спортсмена в ситуации, когда он вместе с тренером только приступает к созданию совершенно нового движения или упражнения. Речь идет о ситуации, когда тренер придумывает совершенно новые элементы в спортивной гимнастике, художественной гимнастике, прыжках в воду, в фигурном катании, различных новых современных видах зимнего спорта (например, halfpipe) и др.



Рисунок 1



Рисунок 2

В таких ситуациях спортсмен ещё никогда не выполнял данного упражнения, и «заснять» учебный фильм не с чего, и не с кого. В таких ситуациях, конечно, можно разъяснить спортсмену его новые движения и элементы «на пальцах». Но гораздо нагляднее смоделировать новое движение на некоем манекене (рис.1-4) в реальной обстановке, рядом со спортивным снарядом, в реальном времени, с возможностью останавливать имитацию упражнения, рассматривать её с разных точек зрения, возвращать назад и пр. Именно такой инструмент и обсуждается в данной статье.



Рисунок 3



Рисунок 4

Нужно подчеркнуть, что речь идет не только о просмотре сгенерированных AVI файлов, в которых можно только останавливать и повторять движения спортсмена. Речь идет о полномасштабной имитации упражнения спортсмена с помощью соответствующей графической системы. В такой системе возможно:

- Полное обновление упражнения.
- Частичная его корректировка.
- Просмотр имитации с разных точек зрения.
- Повторы, возвраты назад, остановки и пр.
- Добавление элементов спортивного инвентаря и др.

Конечно, очевидный недостаток данной методики понятен. Если сравнивать имитацию движения спортсмена, выполненную по предлагаемой методике с помощью манекена, с последующим учебным фильмом, снятым по ходу движения реального спортсмена, то учебный фильм с живым человеком безусловно выиграет. В имитации движения манекена сразу бросается в глаза некоторая искусственность и неестественность поз и движений. Но повторим, что данная методика предлагается только для первоначальной, проработки совершенно нового упражнения. И в этом случае некоторая неестественность человеческих поз и движений манекена допустима.

Конечно, первое, что приходит в голову при такой постановке задачи, - это графические системы типа 3D Studio Max и Autodesk Maya. Но мы выбрали одну из подсистем тяжелой графической системы CATIA. Не будем утверждать, что это самый оптимальный и удачный выбор. Скорее всего, в нашем случае сказалась приверженность системе CATIA по предыдущим промышленным проектам. Для нас оказалось проще и дешевле применить одну из подсистем уже освоенной большой системы, чем приобретать и осваивать меньшую новую графическую систему. Повторим - мы не настаиваем на том, что наш выбор – самый наилучший. Возможно, что имитация движения манекена с помощью системы Autodesk Maya, окажется эффективнее. Мы старались показать, что имитация достаточно сложного движения человека таким способом в принципе возможна. Кроме этого, хотелось лишний раз показать, что применение тяжелых графических систем часто оказывается слишком поверхностным. Многие их приложения практически не используются. Поэтому данную статью можно еще рассматривать и как иллюстрацию применения достаточно развитого раздела Ergonomics Design & Analysis (рис.5) в графической системе CATIA.

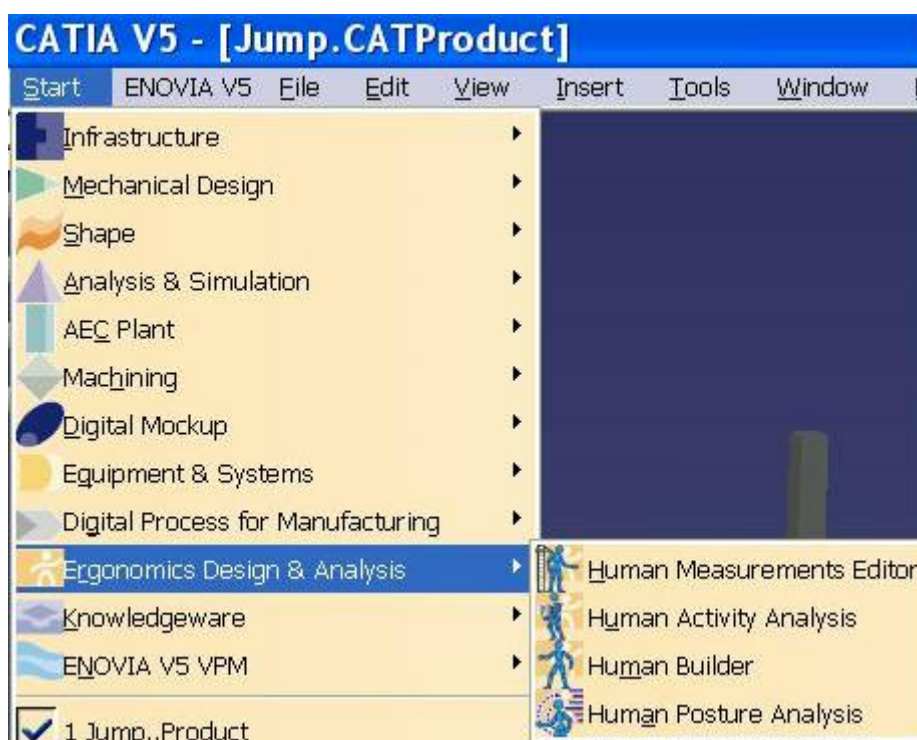


Рисунок 5

Естественно, что разработка нового упражнения спортсмена начинается с объемного моделирования соответствующей спортивной площадки, или спортивного инвентаря. Например, для разработки новых прыжков и переворотов в таком зимнем виде спорта как Halfpipe придется смоделировать объект, показанный на рис. 6.

При имитации прыжка в высоту (конечно, ничего нового в этом упражнении нет, и мы говорим о таком прыжке только в качестве примера) придется разметить сектор для

прыжков, смоделировать стойки и планку (рис.7). Желательно особо указать в секторе те места, на которые прыгун будет становиться ногой во время разбега.

Вообще, еще до имитации движений манекена важно как можно четче представлять себе не только его будущие позы, но и положения в пространстве, в которых спортсмен эти позы будет принимать. Вплоть до указания координат X, Y, и Z центра тяжести спортсмена. Применительно к прыжку в высоту важно знать: как сильно спортсмен подпрыгивает на каждом шаге разбега, как далеко или близко он ставит опорную ногу относительно стоек, как близко он подлетает к планке и пр. Все неточности, допущенные во время такого планирования будущего упражнения, явно проявятся в его итоговой имитации.

Конечно, для имитации известных упражнений очень пригодились бы так называемые фильмограммы, пример одной из которых показан на рис. 8. Но в нашей ситуации таких фильмограмм нет, и тренеру придется самому прикидывать, или рассчитывать положение спортсмена и его позы в каждый момент времени будущего упражнения. Это непросто, и может приводить к понятным ошибкам.

Конечно, выявив все ошибки и неестественности после первого же «проката» полученной имитации, вы тут же сможете эти ошибки исправить.

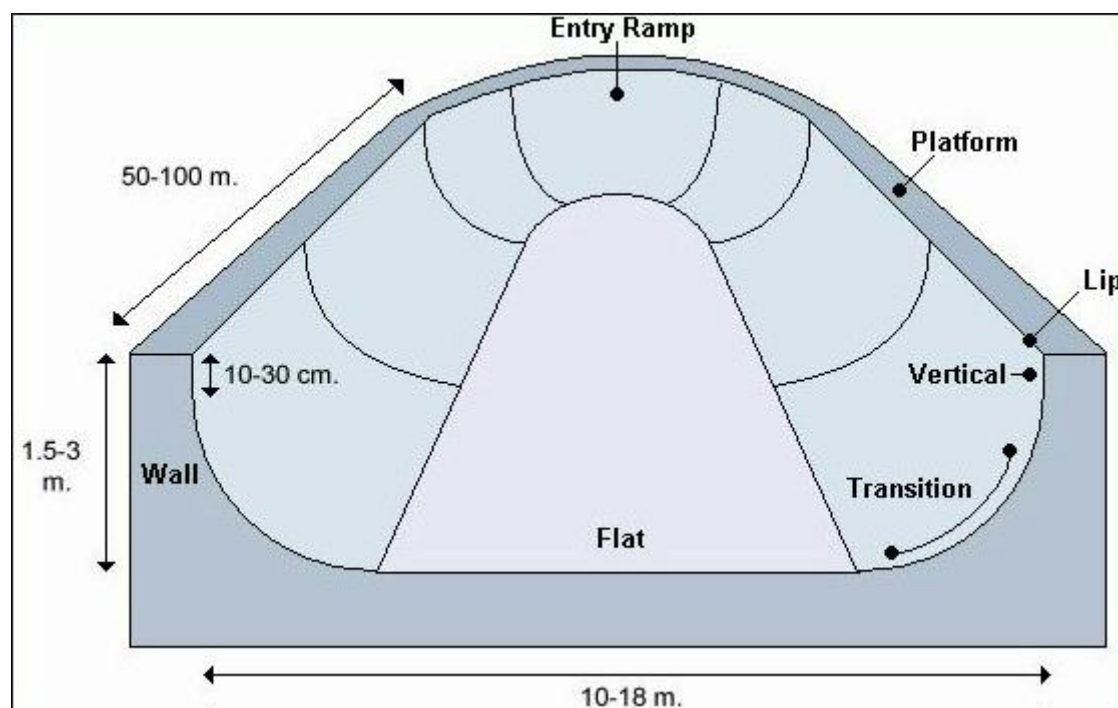


Рисунок 6

Итак, проектирование нового упражнения спортсмена начинается с объемного моделирования его окружения. Нужно сказать, что окружение спортсмена, спортивные снаряды, его реквизит предварительно создаются как отдельные детали, которые впоследствии организуют итоговую сцену, в которой и выполняется новое упражнение.

После того, как вы создали окружение, нужно сгенерировать в этом проекте изображение человеческого манекена.



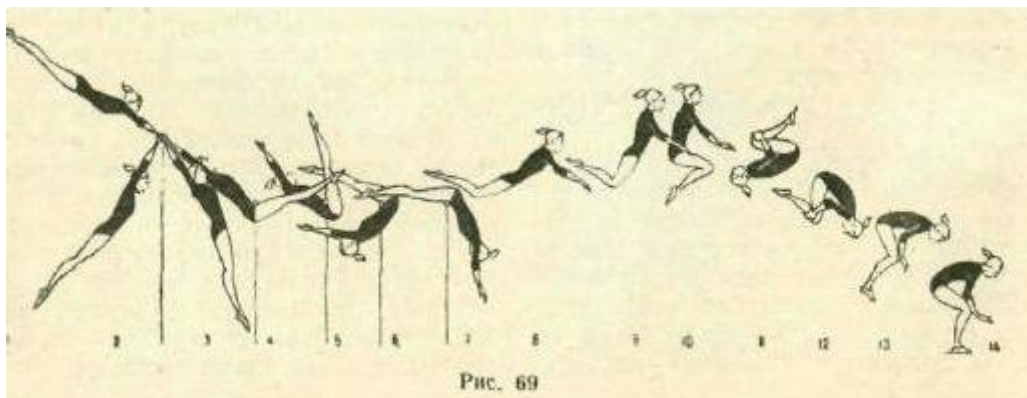


Рисунок 8

Конечно, мы не будем подробно описывать технику выполнения этого этапа. Скажем только, что при этом возможно указать пол, вес манекена, принадлежность его к какой-либо расе. Кроме этого, можно указать точку привязки манекена, за которую вы сможете впоследствии перемещать его внутри сцены и пр.

В Autodesk Maya нет готового манекена человека, его нужно создавать из отдельных геометрических тел и для придания манекену подвижности необходимо привязать к телу кости. В САТИА на это нет необходимости тратить время, однако, при создании манекена вручную с помощью Maya есть возможность усложнить движения, добавить ему большей подвижности, ограниченная только возможностями пользователя.

На рис.9 представлены образцы манекенов разной величины в САТИА.

Система предоставляет разнообразные инструменты, с помощью которых можно менять позу манекена самым произвольным образом (рис.10). В частности, можно по отдельности менять положение его рук, ног, головы, ступней, локтей, плеч, и даже каждого из пальцев на руках (рис.11).

Но главное в предлагаемой методике - это построение не отдельной позы, а создание целой последовательности кадров, в каждом из которых важно по возможности точно смоделировать и позу спортсмена, и его положение относительно окружающей среды. Последовательность таких кадров впоследствии и создаст впечатление движущегося гимнаста, фигуриста, прыгуна и т.д.

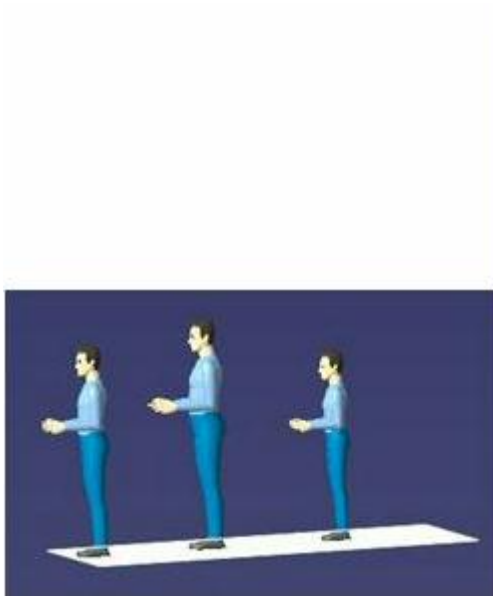
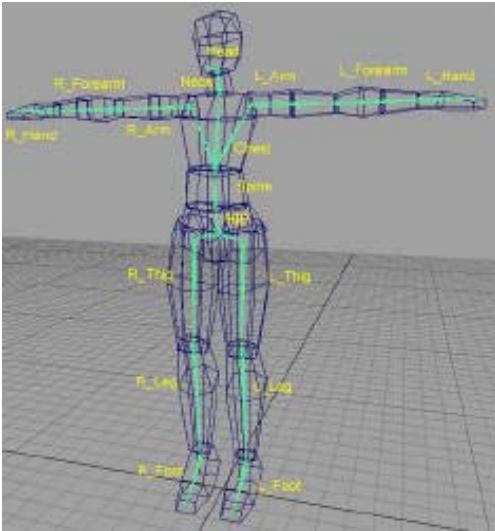


Рисунок 9



Рисунок 10

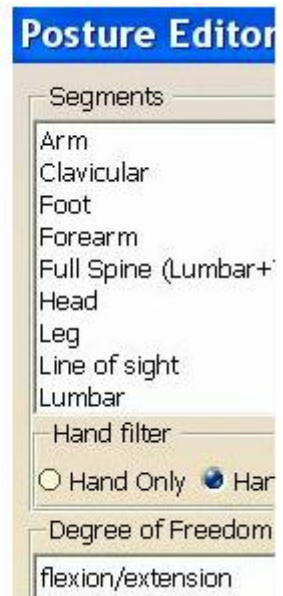


Рисунок 11

Поэтому предварительно приходится осуществить примерную раскадровку поведения манекена в будущем упражнении. Это означает, что придется оговорить некоторое число различных поз, которые по ходу действия будет принимать манекен, и рассчитать его местоположение для каждой из таких поз в каждом будущем кадре относительно окружающих спортивных снарядов. Каждую из сгенерированных поз нужно сохранить в соответствующем каталоге, что позволяет впоследствии многократно их использовать и редактировать. На рис. 12 показано содержимое раздела каталога, в котором хранятся различные позы манекена. Аналогично хранятся и соответствующие пространственные положения манекена.

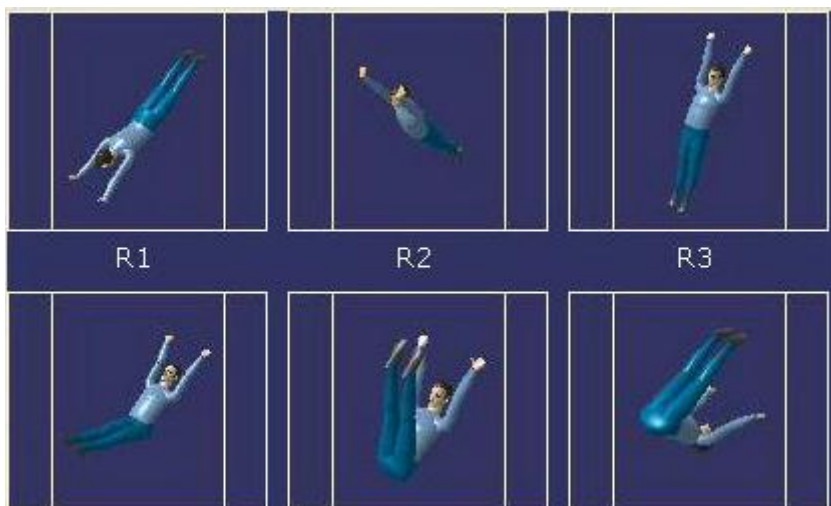
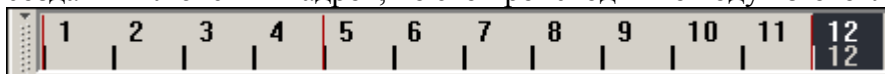


Рисунок 12

Принцип создания анимации манекена состоит в том, что сначала вы «вручную» создаете несколько ключевых кадров и положений манекена, а потом система дополняет их промежуточными подкадрами, которые в соответствии с заданным числом вставляются между ключевыми кадрами. Таким образом, создается последовательность ключевых и промежуточных кадров, которая в совокупности и позволит имитировать требуемое движение манекена.

В Autodesk Maya процесс создания анимации устроен проще. Он так же основан на создании ключевых кадров, но это происходит по ходу изготовления анимации.



Выбирается положение и поза манекена и сохраняется в качестве ключевого кадра (отмечены красным) на линии времени. Система сама дополнит анимацию промежуточными кадрами. Имеется возможность быстрого редактирования промежуточных кадров для придания реалистичности движениям. При этом он становится ключевым, но количество промежуточных кадров не изменится, в отличие от CATIA, где при создании анимации заранее выбирается количество промежуточных кадров между ключевыми.

Нужно подчеркнуть, что имитация движения манекена в CATIA состоит из двух частей: подготовительной и основной.

На подготовительном этапе мы заготавливаем все необходимые позы манекена и фиксируем соответствующие им положения манекена в пространстве. Вся эта информация запоминается в соответствующем каталоге.

На основном этапе мы формируем последовательность ключевых кадров, дополняем их промежуточными подкадрами, и определяем время имитации.

Нельзя утверждать, что описываемая технология имитации движения спортсмена полностью удовлетворяет поставленным требованиям. В частности, еще не решен вопрос наложения геометрических ограничений на манекен и элементы его реквизита. Например, во время сложных переворотов спортсмена - сноубордиста важно, чтобы доска сноуборда плотно прилегала к его ногам. Еще сложнее организовать свободное перемещение, например, обруча, который подбрасывает, а потом опять ловит гимнастка. Эти вопросы еще предстоит решить. Конечно, при первоначальной разработке нового упражнения можно обойтись и без таких частностей, но представляет интерес сама технология наложения геометрических ограничений и свободное перемещение предметов.

Важным компонентом предложенной методики являются многократные исправления уже полученных имитаций движения спортсмена. Самыми распространенными исправлениями являются:

- Добавление в готовую имитацию новых дополнительных кадров

- Замена в готовой имитации отдельных кадров. Причем менять приходится и позу манекена, и его положение относительно окружающей среды.
- Изменение скорости перемещения манекена на отдельных участках его траектории. Если на подготовительном этапе вы неправильно выполнили раскадровку будущего упражнения, и неравномерно сформировали ключевые кадры по всей траектории движения манекена, то в итоге на отдельных участках он будет двигаться где-то быстрее, а где-то медленнее.

В Maya процесс исправления организован проще, ключевые кадры можно перемещать на линии времени, добавлять промежуточные кадры без изменения скорости на отдельных участках. Существуют так же специальные инструменты редактирования скорости воспроизведения в разных временных промежутках. Например, несложно изменить положение объекта в ключевом кадре — достаточно активировать ключевой кадр, переместить объект и создать новый ключ анимации, в отличие от САТІА, в которой для изменения положения требуется создание анимации с начала.

Следует отметить, что некоторые исправления в системе САТІА не всегда удобны. Поэтому приходится разрабатывать собственные технологии отладки. Однако возможностей этой системы вполне достаточно для отработки новых упражнений спортсмена.

### Литература

1. Басов К.А. САТІА V5. Геометрическое моделирование / К.А. Басов.- СПб.: Питер, 2008.- 280 с.
2. Мартынюк В.А. САТІА\_ Начало работы [Электронный ресурс] 2007 //, (<http://bigor.bmstu.ru>), № 0320702622. Проверено 12.05.2010
3. Шляхтина С. Первые шаги в Maya. Урок 5. Введение в анимацию Web-журнал Компьютер Пресс'12 2006 (<http://www.compress.ru/>)

# СЖАТИЕ ТАБЛИЦ РЕШЕНИЙ С ОГРАНИЧЕННЫМ ВХОДОМ. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ

**Жейкарс А.Р.**

*к.т.н., доц. Божко А.Н.*

МГТУ им. Н.Э. Баумана, Москва, Россия

## COMPRESSION OF DECISION TABLES WITH LIMITED INPUT. SOFTWARE IMPLEMENTATION

**Zheykars A.R.**

*Ph. D., senior lecturer, Bozhko A.N.*

Moscow State Technical University n.a. N.E. Bauman, Moscow, Russia

### Аннотация

В работе рассматривается аппарат представления знаний на основе таблиц решений с ограниченным входом. Описаны основные проблемы, связанные с обработкой таких таблиц. Особое внимание уделено проблеме сжатия таблиц решений. Описана программная реализация этой задачи.

### Abstract

The paper explores knowledge representation technique using decision tables with limited input. Main problems associated with this model are described. The main attention is paid to the problem of compression. Also the software implementation is described.

### Введение

Табличные модели используются в информационных технологиях уже достаточно продолжительное время. Видимо, их первое значительное применение связано с описанием логики программ и предложено в рамках совместного проекта General Electric, Sutherland Corporation и BBC США. В этой разработке обсуждался частный, но очень важный вид табличных моделей, получивший название таблиц решений (ТР).

Со временем область применения ТР существенно расширилась, а тематика исследований сместилась в область разработки методик использования ТР для представления знаний в интеллектуальных системах поддержки принятия решений (ИСППР).

Таблица решений представляет собой прямоугольную форму, состоящую из четырех обязательных разделов или секций (см. рис. 1).

Условие	Значения условий
Действия	Тело действия

Рисунок 1 - Структура таблицы решений

В левой верхней секции перечисляются имена различных условий. В правой верхней секции отмечаются различные состояния условий. В левой нижней секции перечисляются возможные действия. В правой нижней секции отмечаются действия, которые можно выполнить при соответствующем состоянии всех условий.

Например, на рисунке 2 приведен фрагмент ТР для выбора токарных автоматов. В правой верхней секции ячейка помечена крестиком, если соответствующее условие выполняется. Крест в ячейке правой нижней секции означает, что при такой комбинации значений условий, которая указана в данном столбце в правой верхней секции, должен быть выбран станок, модель которого записана в строке, соответствующей ячейке.

	1	2	3	4	5	6	7
1 DR ≤ 12		X					X
2 DR < 18		X	X				X
3 DR ≤ 24		X	X	X	X	X	X
4 D2 ≤ 10		X	X	X			X
5 D2 ≤ 12		X	X	X	X		X
6 D2 ≤ 22		X	X	X	X	X	X
7 LZ ≤ 60		X	X	X	X		
8 LZ ≤ 90		X	X	X	X	X	X

	1	2	3	4	5	6	7
1 1B112		X	X				
2 1B118				X	X		
3 1A124							X
4 1B136						X	

Рисунок 2 - Фрагмент ТР для выбора токарных автоматов

Таблицы решений получили широкое распространение благодаря следующим качествам:

- Любые типы двухвходовых таблиц (кроме алгоритмических), которые в изобилии встречаются в проектной практике инженеров, технологов и архитекторов, можно представить в унифицированной форме – в виде ТР;
- ТР – удобный способ формализации любых одношаговых процессов принятия решений. Одношаговыми называются такие ситуации принятия решений, когда комбинация выполнимых условий детерминирует выбор множества валидных действий;
- ТР – хороший языковый интерфейс между специалистами в прикладных областях и программистами.

На рисунке 3 приведена диаграмма, показывающая области применения таблиц решений.

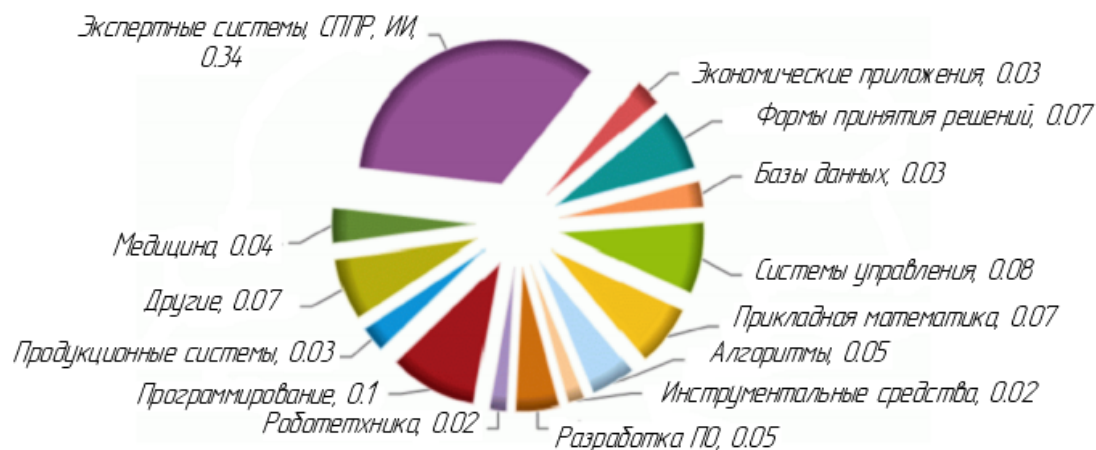


Рисунок 3 - Применение таблиц решений (по данным [6])

Таблицы решений, описывающие реальные проектные ситуации, могут иметь очень высокую размерность. Так, при технологической подготовке механосборочного производства технологи нередко сталкиваются с таблицами, размеры которых достигают нескольких сотен строк и нескольких тысяч столбцов. По этой причине задача генерации сжатой компактной формы ТР является важной и актуальной. В работе рассматривается метод сжатия частного класса таблиц решений – таблиц с ограниченным входом. Таковыми являются ТР, все условия которых принимают значения в бинарном алфавите. Возможные значения условий толкуется как их истинность или ложность. Таблицы с многозначными, неопределенными или вероятностными условиями не рассматриваются.

Пусть дана некоторая многозначная таблица решений (см. рис. 4). Рассмотрим процедуру приведения многозначной таблицы к стандартной форме, т.е. к полной таблице с ограниченным входом. Таблица называется полной, если в секции значений условий перечислены все возможные кодовые комбинации.

Материал	B90	Шx15	Ст10
Резьба	да	нет	да
Станок 1	X		X
Станок 2	X	X	

Рисунок 4 - Многозначная ТР

Процедура приведения таблицы к стандартной форме включает следующие этапы:

1. Выбор способа кодирования и представление небинарных условий в двоичном виде.
2. Пополнение таблицы всеми недостающими кодовыми комбинациями.
3. Включение в таблицу фиктивного действия.
4. Связывание всех добавленных кодовых комбинаций с фиктивным действием.

На рисунке 5 изображена ТР, приведенная к стандартному виду.

Материал	0	0	1	0	0	1	1	1
	0	1	0	0	1	0	1	1
Резьба	1	0	1	0	1	0	0	1
Станок 1	X		X					
Станок 2	X	X						
Empty				X	X	X	X	X

Рисунок 5 - ТР стандартного вид

Стандартный вид ТР обладает важным преимуществом: в таком виде таблицу можно рассматривать, как таблицу истинности булевой функции. Это дает нам право применить технику минимизации булевых функций для сжатия ТР.

### Сжатие таблиц решений

Для сжатия таблиц решений с ограниченным входом предлагается использовать следующий алгоритм:

1. Разложить таблицу на подтаблицы. В каждую из подтаблиц включить столбцы, выбирающие одно и то же действие. Столбцы, за которыми закреплены несколько действий включаются в разные подтаблицы.
2. Сжать каждую из подтаблиц.
  - 2.1. Поставить таблице решений в соответствие булеву функцию представленную в виде СДНФ.
  - 2.2. Найти все простые импликанты полученной булевой функции.
  - 2.3. Построить импликантную таблицу.
  - 2.4. Найти минимальное столбцовое покрытие импликантной таблицы.
  - 2.5. Построить по минимальному покрытию минимальную ДНФ булевой функции.
  - 2.6. По минимальным ДНФ восстановить сжатые подтаблицы.
3. Объединить сжатые подтаблицы в единую таблицу.

Очевидно, что сжатие таблиц решений сводится к задаче нахождения минимальной ДНФ булевой функции. В вышеописанной схеме алгоритма пункты 2.1. – 2.5. повторяют шаги классического алгоритма нахождения минимальной ДНФ Квайна-МакКласки.

В качестве кандидатов были рассмотрены несколько методов минимизации булевых функций в классе ДНФ. Среди них метод неопределенных коэффициентов, метод Блейк-Порецкого, метод минимизации с помощью карт Карно. Из рассмотренных методов для программной реализации на ЭВМ нецелесообразно использовать методы неопределенных коэффициентов и метод карт Карно. Главное преимущество метода Блейка-Порецкого перед методом Квайна-МакКласки заключается в том, что метод Блейка-Порецкого не требует

приведения исходной формулы к СДНФ. Но для задачи сжатия таблицы решений это не существенно, т.к. соответствующая булева функции будет задана в СДНФ (в случае таблиц решений с ограниченным входом и при отсутствии заданных изначально состояний « - » для условий).

Метод Квайна-МакКласки на одном из заключительных этапов требует решения задачи минимального покрытия (ЗМП) для импликантной таблицы. В рамках работы осуществлялся выбор метода решения ЗМП. Были рассмотрены жадный алгоритм, минимаксный алгоритм, точный алгоритм. Выбор пал на минимаксный алгоритм решения задачи ЗМП. Данный алгоритм быстр и эффективен. Разумеется, он работает дольше жадного алгоритма, но дает лучшие результаты. К недостаткам алгоритма стоит отнести его приближенность, следовательно, в общем случае результат может быть не оптимальным. Предпочтение минимаксному алгоритму в сравнении с точным было дано по причине экспоненциального времени работы последнего. Начиная с определенного размера таблицы решений, пользователю потребовалось бы ждать многие часы для сжатия его таблицы. Это неприемлемо. Хотя в перспективе развития программного продукта планируется адаптивный выбор точного или приближенного алгоритма исходя из анализа размерности и концентрации единиц матрицы Квайна.

### Программная реализация

Программа написана на языке C++ с использованием библиотек Qt 4.7 и boost 1.45. Объем исходных текстов 70 Кбайт. Для работы программы необходима операционная система Windows XP или выше или Linux с установленной библиотекой Qt 4. Примерное количество строк кода – 1500. Программа обладает графическим интерфейсом. В разработанной программе реализована возможность сохранения и загрузки таблицы решений из файла формата XML. С целью повторного использования и валидации разработанной структуры XML-файла было разработано ее описание на языке XML Schema (XSD). Генерация и чтения XML-файлов в программе осуществляется средствами библиотеки Qt, а именно с использованием класса QDomDocument, реализующим стандартный программный интерфейс для работы с XML-файлами Document Object Model.

Проводилось автоматизированное тестирование реализованного метода Квайна-МакКласки. При этом использовался модуль QTest библиотеки Qt.

### Литература

1. А.Д. Закревский, Ю.В. Поттосин, Л.Д. Черемисинова. Логические основы проектирования дискретных устройств. – М.:ФИЗМАТЛИТ, 2007.
2. Э. Хамби. Программирование таблиц решений. – М.:МИР, 1976.
3. Божко А.Н. Сжатие таблиц решений. //Наука и образование. [Электронный ресурс]. URL: <http://technomag.edu.ru/doc/48528.html>.
4. Виноградов О.В. Методы и программные средства представления знаний на основе нечетких таблиц решений и их применение в интеллектуальных системах: Дис. канд. тех. Наук 05.13.11 / Московский энергетический институт. М., 2009. 243 с.
5. Башлыков А.А., Еремеев А.П. Экспертные системы поддержки принятия решений в энергетике / Под. ред. А.Ф. Дьякова. М.:Издательство МЭИ, 1994.
6. Garcia A.M., Verhelle M., Vanthienen J. An overview of decision table literature 1982 – 2000 // Fifth International conference on AI and emerging technologies in accounting, finance and tax, 2000, Huelva, Spain.
7. Шлее М. Qt4. Профессиональное программирование на C++. – СПб.: БХВ-Петербург, 2007. – 880 с.



## НЕКОТОРЫЕ ПРОБЛЕМЫ ФИЛОСОФИИ ИНФОРМАТИЗАЦИИ

Шахнов Д.В.

*Научный руководитель: докт. ист. наук., проф. Малышева О.Г.*  
Российская академия государственной службы при Президенте РФ

## SOME PROBLEMS OF INFORMATIZATION PHILOSOPHY

Shakhnov D.V.

*The supervisor of studies Professor Malysheva O/G/*

### Аннотация

Рассмотрены некоторые философские проблемы самой динамично развивающейся отрасли деятельности человека: информационных технологий. Уделено внимание взаимодействию человека и ЭВМ, влиянию компьютерной «революции» на социальную сферу, значению интеллектуализации техники и использования виртуальной реальности и др.

В последние десятилетия во всём мире неуклонно происходит увеличение числа людей, работающих в информационной сфере. Так, например, в США сто лет назад в информационной сфере было занято 5 процентов работающих, в производственной - 95 процентов. В 1980 году это соотношение было уже 45 и 55 процентов, причём подобное перераспределение людей продолжается теми же темпами.

Массовая компьютеризация информационной сферы влечет за собой использование ЭВМ широкими кругами непрограммирующих специалистов, в связи с этим перед разработчиками средств информатизации стоит основная задача: обеспечить машинам такие свойства, чтобы с ними можно было обращаться без специалистов-посредников: математиков, программистов, операторов. В связи с этим усилился интерес к искусственному интеллекту, к его программно-аппаратным средствам, получившим название «комплексный диалоговый интерфейс, или просто «интерфейс». К сожалению, до сих пор не изжито представление о том, что возможно создать искусственный интеллект как синоним искусственного разума – об этом и речи быть не может по целому ряду принципиальных соображений.

Возьмём шахматы. Чем руководствуется шахматист, делая тот или иной ход, - прецедентами, прошлым опытом, умением, интуицией, догадкой, просмотром на сколько-то ходов вперёд? Мы не знаем. Но зато мы в точности знаем, как это делает ЭВМ, так как человек составил для неё программу – инструкцию, позволяющую за достаточно короткий промежуток времени проанализировать тысячи вариантов очередного хода, количественно оценить ту или иную ситуацию. На машине играет в шахматы не программа, а всё тот же человек, который сумел формализовать шахматную игру и составить программу.

В настоящее время среди всех систем, ориентированных на знания, особое внимание уделяется так называемым экспертным системам. Мы хорошо знаем, что во всяком использовании ЭВМ существует триада: модель, алгоритм, программа (если не учитывать при этом «железо», то есть саму машину). Для того, что бы использовать ЭВМ, мы должны обязательно иметь модель в виде уравнения или других математических категорий и далее построить алгоритм решения на основе этой модели и запрограммировать его. Мы живём в мире правил и окружены правилами, или, другими словами, продуктами. В медицине – это правила диагностики и лечения, накопленные тысячелетиями. Вспомним грамматические правила, правила дорожного движения. Все правила укладываются в формулу «если – то». Как видно, в продукции есть левая часть – ситуация и правая часть – действие. Если взять статьи Уголовного Кодекса, то левая часть – это диспозиция, а правая – санкция. Набор продуктов из какой-либо области знаний образует базу знаний экспертной системы и в зависимости от состояния системы в диалоге с пользователем определяется по левым частям та или иная продукция, которая изменяет состояние системы.

С помощью названных выше формализмов, особенно производственных систем, в последние годы интенсивно развиваются экспертные системы искусственного интеллекта и построенные на этой основе так называемые мягкие модели. Примером применения давно привычных жёстких моделей могут служить некоторые системы автоматизированного проектирования – САПР. Однако многое, в частности опыт проектировщика, может быть отражено лишь в мягких моделях. Синтез жёстких и мягких моделей, переход к так называемым гибридным экспертным системам, резко повысит эффективность тех же САПР.

### **1. *Человек и ЭВМ в решении творческих задач.***

Никакое совершенствование информационно-логических программ не заменит творческого потенциала человека. Причина, в частности, в том, что нет универсального алгоритма открытия и формирования новых алгоритмов. Ставка только на совершенствование алгоритмов и программ представляет собой явное преувеличение роли машинного фактора и недооценку человеческого фактора в решении творческих задач. Перечислю важнейшие не формализуемые на сегодня творческие компоненты: постановка задачи или реализация проблемной ситуации; самостоятельная выработка критериев отбора нужных, приводящих к решению операций; генерация догадок и гипотез в процессе поиска основной идеи решения (это научная, художественная, техническая фантазия, не сводимая к комбинаторике и генерации случайных состояний); интерпретация формального решения; понимание и др.

### **2. *Компьютеризация и социальные структуры.***

Компьютеризация, как и связанная с нею роботизация производства, приводит к существенному изменению не только характера труда, но и многих социальных отношений и структур.

Начнём с того, что многие виды трудовой деятельности уже сегодня в принципе могут быть полностью автоматизированы. Какие же следствия влечёт за собой этот процесс? Прежде всего исчезают традиционные единства, представлявшие ранее непременной характеристикой трудового процесса, в первую очередь пространственное единство субъектов и трудового процесса. Дистанционное управление производством разрывает это единство, открывается возможность, например, если не ликвидировать полностью, то существенно сократить ежедневные перевозки людей к месту работы и обратно.

Та же судьба, видимо, ожидает и единство времени. Компьютеризированное производство, работающее в автоматическом режиме, не требует непременного совпадения по времени производственного процесса и человеческой деятельности, обеспечивающей этот процесс.

### **3. *Противоречия информатизации.***

Задумываясь о соотношении искусственного и естественного интеллекта, необходимо учитывать очевидное противоречие – то, что становится искусственным и передаётся машине, перестаёт быть интеллектом, а то, что подлинно интеллектуально, остаётся вне функций компьютера. Причём мы не можем получить выигрыш в искусственном интеллекте, не проиграв чего-то в естественном. Подобное уже бывало – когда мы что-то приобретали в техническом отношении, то и что-то теряли, например, в экологическом плане. Мы очень активно начинаем развивать искусственный интеллект, весьма мало ещё зная естественный. Мы, например, не знаем, как возникает качественный скачок в человеческом информационном процессе, а ведь в нём главная ценность творчества.

При любом росте электронизации остаётся вечный вопрос – как продуцировать новую информацию. Ещё идут споры, даёт ли ЭВМ принципиально новую информацию, или она лишь перерабатывает данные человека, переставляет их элементы. Признание существенных ограничений возможностей ЭВМ практически полезно; нельзя перегружать корабль чрезмерными надеждами. Машины за нас не решают человеческих проблем, и это важно сказать сейчас, когда происходит такое увлечение компьютерами. Дети часто даже полагают,

что таблицу умножения сейчас учить не обязательно, раз счёт автоматизирован. Если мы не задумаемся над противоречиями компьютеризации, а будем только говорить о её плюсах, то рискуем упустить из вида объективный ход вещей.

Становление информатики как бы обошло трудности, которые связаны с определением природы информации, и это остаётся большой проблемой для философов. Уже в рамках самих информационных процессов остаётся важным понимание соотношения информации формальной и информации семантической.

Следует отметить, что анализ закономерностей развития информационной техники целесообразно начать с определения её сущности. Например, ныне забываемый в широких кругах общества известный философ начала прошлого века В. И. Ульянов писал: «...закон и сущность - понятия однородные (однопорядковые) или вернее, одностепенные, выражающие углубление познания человеком явлений, мира...». Можно сказать, что закономерность является формой выражения сущности.

#### **4. *Природа «интеллектуальной» техники.***

Природу «интеллектуальной» техники можно раскрыть через её характерные черты. Применимо к технике прилагательное «интеллектуальная» намерено взято в кавычках, о чём пойдёт речь в последующем. Компьютерная техника – это искусственные средства человеческой деятельности, усиливающие возможности человека для аналитических и наукоёмких работ; созданные исключительно в интересах и предназначенные для достижения определённых знаний и оценки большого количества фактов. В какой-то мере – это овеществлённая сила знаний об окружающей действительности; овеществлённый труд работников материального производства, овеществлённые возможности экономики.

Компьютерная техника органически связана с разными уровнями, сферами развития общества. В самом широком общесоциальном плане к ним относят политику, экономику, науку. Кроме того, в качестве относительно самостоятельной выделяется сфера эксплуатации данного вида техники – это область оборонной деятельности. Нельзя не упомянуть применение компьютеров в ежедневной деятельности человека. Этот вид функционирования ЭВМ – самый распространённый. Сюда включаются и компьютерные игры, и доступ к информационным ресурсам и многое другое.

#### **5. *Общесоциальные закономерности развития информатизации.***

Как ни парадоксально это звучит, но, на мой взгляд, основными социальными причинами развития вычислительной техники, резкого технологического прорыва, организации самого капиталоемкого производства за всю историю человечества являются лень и любопытство, присущие каждому человеку в той или иной степени. Никому не хочется заниматься рутинными вычислениями и сбором информации по крупицам, всегда у человека есть желание переложить часть своей работы на кого угодно, лишь бы не делать самому. Пусть «вкалывают» роботы – homo sapiens должен быть счастлив. В итоге мы получили миллионную армию инженеров, которые смутно представляют себе ход технологического процесса, а вся работа сводится к анализу двух - трёх цифр и выдаче очевидного результата. Сам компьютер в итоге превратился в подобие детского конструктора, когда любой в силах из общественного списка деталей собрать ЭВМ и наслаждаться суперграфикой последних игр.

Глубинными экономическими предпосылками возникновения и функционирования вычислительной и телекоммуникационной техники явились антагонистские противоречия в способе производства материальных благ. Казалось бы, что внедрение «интеллектуальной» техники призвано повысить эффективность технологического производства в несколько раз. Теоретически казалось (существовали даже научные обоснования), что это должно привести к отторжению человека из сферы производства и полной замены человека роботом. Доказывалось, что социум ждёт небывалые потрясения, новые восстания «луддитов». На деле это оказалось не так. Действительно, там где раньше с работой справлялись группа из двух бухгалтеров со счётами (или на худой конец – с калькуляторами), теперь появился

целый бухгалтерский отдел, каждый специалист которого разбирается в одной – двух программах. К этому необходимо приплюсовать системного администратора, который управляет компьютерной сетью отдела, инженера, который решает аппаратные проблемы, возникающие в процессе эксплуатации техник. Сюда же необходимо добавить работу специалистов, которые разработали вышеуказанные программы и регулярно исправляют различного рода сбои в программном обеспечении. Подобного рода ситуацию можно обнаружить во многих сферах деятельности человека.

#### **6. *Виртуальная реальность – этап развития информатизации.***

Приоритетным в последние годы стало развитие не информационных, а имитационных технологий – технологий виртуальной реальности. В результате наращивания оперативной памяти и быстродействия компьютеров, а также создания нового программного обеспечения возникают не только качественно новые формы передачи и обработки данных, но и, в первую очередь, достигается всё большее сходство между работой на компьютере и управлением реальными объектами, а также сходство коммуникаций в режиме online с общением в реальном пространстве – времени.

Киберпротезирование институциональных форм является характерной особенностью виртуального взаимодействия – виртуальных сообществ, виртуальных корпораций, виртуальных развлечений, виртуальных преступлений и виртуального же отпущения грехов.

Сегодня с помощью компьютера, имеющего доступ в интернет, можно обсуждать политиков, поп-звезд, погоду или вести досужие беседы с виртуальными друзьями или соседями – участниками чата или социальной сети. Можно заработать деньги, принимая заказы на размещение рекламы на виртуальных щитах-баннерах, выиграть деньги в виртуальных казино или украсть те же деньги, взломав виртуальные замки электронной системы учёта какого-нибудь банка. Обличить пороки и покаяться в содеянном можно в виртуальных проповедях и блогах на web-страницах, открываемых священниками в качестве виртуальных приходов.

Столь интенсивное в последние годы использование технологий виртуальной реальности имеет социальный смысл – замещение социальной реальности её компьютерными симуляциями. Этот социальный аспект развития компьютерных технологий явно превалирует над техническим аспектом. Именно поэтому наращивание быстродействия процессора и объёма оперативной памяти практически без остатка конвертируется в совершенствование графики и звучания компьютерных симуляций и не сопровождается ощутимыми функциональными изменениями. Обнаружение социального смысла развития технологий виртуальной реальности с необходимостью приводит социологов к идее использовать понятие виртуальности для объяснения общественных изменений.

#### **Литература**

1. Клаус Г. Кибернетика и философия. М., «Иностранная литература», 1983.
2. [http://philsci.univ.kiev.ua/biblio/FIL\\_XX/15.html](http://philsci.univ.kiev.ua/biblio/FIL_XX/15.html) - Г. Блумберг и др. «Философия техники XX века».

# НАХОЖДЕНИЕ ОПТИМАЛЬНОГО ПУТИ ЛЕТАТЕЛЬНОГО АППАРАТА МЕТОДОМ НЕЙРОДИНАМИЧЕСКОГО ПРОГРАММИРОВАНИЯ

Козлова О.Г.

Научный руководитель: д.ф.-м.н., профессор Карпенко А.П.

МГТУ им. Баумана, кафедра РК6, Москва, Россия

## AIRCRAFT OPTIMAL PATH FINDING VIA NEURO-DYNAMIC PROGRAMMING

O.G. Kozlova

Scientific adviser: professor A.P. Karpenko

BMSTU, Moscow, Russia

### Аннотация

В работе рассматривается задача поиска наикратчайшего пути летательного аппарата из начального положения в заданную точку. Исследуется подход к решению задачи на основе нейродинамического программирования с применением приближенного Q-обучения, в котором в качестве универсального аппроксиматора используется многослойный перцептрон.

### Abstract

In this paper the problem of finding the aircraft shortest path from initial position to given point is considered. A solution methodology based on neuro-dynamic programming approach is proposed. For this purpose we apply Q-learning algorithm which uses multilayer perceptron as universal approximator.

Одним из аспектов решения задачи траекторной безопасности является нахождение оптимального по быстродействию пути летательного аппарата (ЛА) до заданной точки в пространстве с необходимостью облета динамических запрещенных зон, которые в том числе могут создавать другие ЛА.

### Постановка задачи

Пусть движение ЛА задается системой уравнений:

$$\dot{X} = F(t, X, U), X(0) = X^0, t \in [0, T] \quad (1)$$

где  $X = (x_1(t), x_2(t), \dots, x_n(t))$  –  $n$ -мерный вектор фазовых переменных системы,  $U = (u_1(t), u_2(t), \dots, u_m(t))$  –  $m$ -мерный вектор управлений,  $X^0$  –  $n$ -мерный вектор начальных условий,  $t \in [0, T]$  – анализируемый период времени. Должны быть заданы также условия на конце траектории системы, например, в виде  $X(T) = X^T$ . Задано множество допустимых управлений  $D_U$ , так что  $U \in D_U$ , а также множество допустимых значений вектора фазовых переменных  $D_X$ ,  $X \in D_X$ .

В задаче оптимального быстродействия критерий качества управления  $J_0(X, U)$  соответствует времени  $T$ , и задача ставится как нахождение траектории ЛА, удовлетворяющей условию:

$$J_0(X, U) = T \rightarrow \min_{U \in D_U}. \quad (2)$$

### Метод нейродинамического программирования

Классические методы динамического программирования применимы к задачам с относительно небольшим числом состояний. В противном случае они весьма ресурсоемки и требуют слишком большого числа вычислений. В подобных случаях приходится отказаться от поиска оптимальной стратегии и искать субоптимальную. В контексте данной статьи интерес представляет субоптимальная процедура, которая включает в себя использование нейронных сетей с целью аппроксимации функции стоимости перехода  $f^*(X)$  для всех состояний  $X$  динамической системы [1].

Одним из методов определения оптимального управления посредством нейродинамического программирования является Q-обучение. Пусть  $U$  – управление, которое из состояния  $X$  приводит систему (1) в состояние  $X'$  при затратах  $g(X, U, X')$ . Тогда вычисление оптимального Q-фактора [2] можно определить как двухшаговую версию уравнения оптимальности Беллмана

$$Q^*(X, U) = g(X, U, X') + \min_{U' \in D_U} Q^*(X', U'). \quad (3)$$

На каждой итерации алгоритм Q-обучения резервирует Q-фактор для одной пары состояние-управление, а именно для текущего наблюдаемого состояния и фактически применяемого управления. Как только оптимальные значения Q-факторов становятся известны, оптимальное управление может быть определено без особо трудоёмких вычислений [2]:

$$U^* = \arg \min_{U \in D_U} Q^*(X, U). \quad (4)$$

Сходимость процесса Q-обучения к оптимальной стратегии предполагает представление Q-факторов в виде справочных таблиц. Однако если входное пространство, состоящее из пар состояние-управление, является достаточно большим или если входные переменные непрерывны, использование справочных таблиц может быть недопустимо затратным вследствие требования громадных объемов памяти. В такой ситуации для аппроксимации функций можно использовать нейронные сети [3]. В этом случае алгоритм приближенного Q-обучения может быть записан следующим образом:

1. Начинаем с произвольного вектора весов  $w_0$ , на основе которого получаем Q-фактор  $Q(X_0, U_0, w_0)$ . Вектор весов  $w$  относится к нейронной сети, используемой для осуществления аппроксимации.

2. Для итераций  $i = 1, 2, \dots$  выполняем следующие действия.

2а. Для данного вектора весов  $w$  нейронной сети определяем оптимальное управление

$$U_i = \min_{U \in D_U} Q_i(X_i, U, w).$$

2б. Определяем целевой Q-фактор

$$Q_i^{\text{цел}}(X_i, U_i, w) = g(X_i, U_i, X'_i) + \min_{U' \in D_U} Q_i(X'_i, U', w).$$

2в. Применяем вектор  $(X_i, U_i)$  в качестве входа нейронной сети, которая генерирует выходной сигнал  $Q_i(X_i, U_i, w)$ , являющийся аппроксимацией целевого Q-фактора  $Q_i^{\text{цел}}(X_i, U_i, w)$ . Слегка изменяем вектор весов  $w$  так, чтобы приблизить аппроксимацию  $Q_i(X_i, U_i, w)$  к целевому значению  $Q_i^{\text{цел}}(X_i, U_i, w)$ .

2г. Возвращаемся к шагу 2а и повторяем вычисления.

Таким образом, если алгоритм не расходится, вектор весов  $w$  обеспечивает хранение аппроксимированных Q-факторов в обучаемой нейронной сети.

**Пример.** Рассматриваются траектории полета ЛА, лежащие в вертикальной плоскости. В этом случае математическая модель, описывающая движения центра масс ЛА в нормальной земной системе координат, описывается следующей системой нелинейных дифференциальных уравнений [4]:

$$\begin{cases} \dot{v} = g \cdot (u_1 - \sin \theta); \\ \dot{\theta} = g/v \cdot (u_2 - \cos \theta); \\ \dot{x} = v \cdot \cos \theta; \\ \dot{y} = v \cdot \sin \theta. \end{cases}$$

Здесь  $v$  – скорость ЛА,  $\theta$  – угол наклона траектории,  $x, y$  – координаты центра масс ЛА,  $u_1$  – тангенциальная перегрузка,  $u_2$  – нормальная перегрузка,  $g$  – ускорение свободного падения.

Покроем временной интервал  $[0, T]$  равномерной сеткой  $t_k$ ,  $k = [0, N]$  с шагом  $\tau$  (рисунки 1).

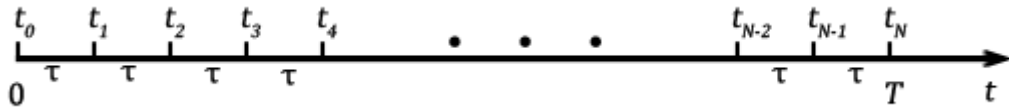


Рисунок 1 – Равномерная временная сетка на интервале  $[0, T]$

Зададим постоянные управления на промежутках  $[t_k, t_{k+1}]$  для всех  $k = [0, N - 1]$ . Таким образом, значение Q-фактора на  $k$ -ой итерации будет определяться вектором параметров состояния ЛА  $[v_p, \theta_p, x_p, y_p]$  и управлением  $[u_{1,i}, u_{2,i}]$ , которое применяется на протяжении следующих  $\tau$  секунд.

Был проведен вычислительный эксперимент, в котором входные величины принимают следующие значения:  $v_{\text{вв}} \leq v \leq 1.9 \cdot v_{\text{вв}}$ ,  $|\mu_1| \leq 0.6$ ,  $|\mu_2| \leq 8$ ,  $T = 5 \text{ с}$ ,  $\tau = 0.5 \text{ с}$ . Здесь  $v_{\text{вв}} = 33146 \text{ м/с}$  – скорость звука в воздухе. В качестве аппроксиматора использовался многослойный перцептрон, состоящий из 32 нейронов.

В качестве примера на рисунке 2 приведено сопоставление оптимальной траектории и траектории, найденной с помощью нейронной сети. На рисунке 3 представлено сравнение найденного управления с оптимальным. Полученное управление приводит ЛА в заданную точку за 4.93 секунд, тогда как оптимальное время составляет 4.9 секунд.

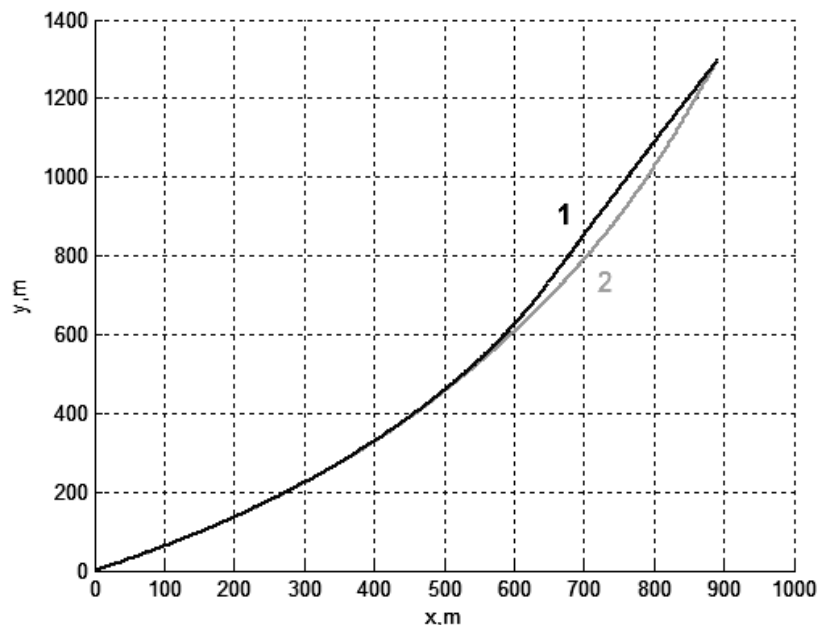


Рисунок 2 – Траектории полета ЛА при начальных условиях  $v_0 = v_{\text{вв}}$ ,  $\theta_0 = \pi/6$ ,  $x_0 = 0 \text{ м}$ ,  $y_0 = 0 \text{ м}$  с оптимальным управлением (1) и управлением, найденным с помощью обученной нейронной сети (2)

### Заключение

Эксперименты показали, что метод нейродинамического программирования применим к поиску оптимального по быстродействию управления ЛА. Хорошая масштабируемость данного метода с увеличением размерности задачи и его возможности к распараллеливанию делают перспективными дальнейшие эксперименты с усложнённой трёхмерной моделью ЛА.

В дальнейшем предполагается реализация данного алгоритма для ситуации, когда в области достижимости ЛА имеются движущиеся препятствия (запрещенные зоны), которые могут создаваться, в том числе, другими ЛА.

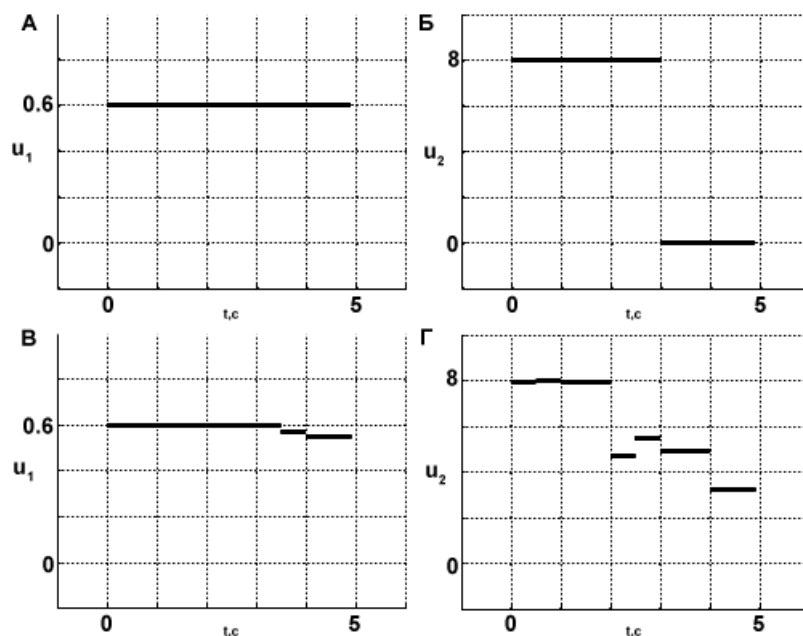


Рисунок 3 – Оптимальные управления (А,Б) и управления, найденные нейронной сетью (В,Г)

Авторы выражают благодарность Воронову Е.М. и Карпунину А.А. за постановку задачи и многочисленные плодотворные обсуждения подходов к ее решению.

### Литература

1. Саймон Хайкин. Нейронные сети: полный курс. 2-е издание. — Вильямс, 2006. — С. 778-790.
2. Bertsekas D.P., Tsitsiklis J.N. Neuro-Dynamic Programming. — Athena Scientific. Belmont, MA. — 1996, — Pp. 59-75.
3. Bauso D., Giarre L., Pesenti R. Multiple UAV cooperative path planning via neuro-dynamic programming // 43rd IEEE Conference on Decision and Control. — Vol. 1. —2004. — Pp. 1087-1092.
4. Воронов Е.М., Карпунин А.А. Алгоритм оценки границ области достижимости летательного аппарата с учетом тяги // Вестник МГТУ. Сер. Приборостроение. — 2007. — №4(69). — С. 81-99.