

**МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ  
УНИВЕРСИТЕТ  
имени М.В. ЛОМОНОСОВА**

**ФАКУЛЬТЕТ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ И  
КИБЕРНЕТИКИ**

---



**СБОРНИК ТЕЗИСОВ ЛУЧШИХ  
КУРСОВЫХ РАБОТ  
2013 ГОДА**

**МОСКВА 2013**

**ББК 22  
С23**

*Сборник тезисов лучших курсовых работ 2013 года – 29 с.*

Редакционный совет сборника:

**Б. И. БЕРЕЗИН, А.Н. ДАРЬИН, А.Г. ДЬЯКОНОВ,  
А.В. ИЛЬИН, И.В. КАПАЛИН, А.С. КОНУШИН,  
В.В. ФОМИЧЕВ, И.Г. ШЕВЦОВА**

В настоящий сборник вошли тезисы курсовых работ, защищенных студентами факультета Вычислительной математики и кибернетики Московского государственного университета имени М.В. Ломоносова в 2013 году, представленные на конкурс лучших курсовых работ.

© Капалин И.В., Шевцова И.Г.  
Составление, оформление, 2013.

# Оглавление

---

## КАФЕДРА МАТЕМАТИЧЕСКОЙ ФИЗИКИ

---

- Хвостиков Александр Владимирович*  
Обработка текстовых изображений с использованием матрицы смежности . . . . . 5

---

## КАФЕДРА ВЫЧИСЛИТЕЛЬНЫХ ТЕХНОЛОГИЙ И МОДЕЛИРОВАНИЯ

---

- Кислицын Алексей Алексеевич*  
Исследование математической модели стохастической динамики ВИЧ-инфекции . . . . . 6

- Матвеев Сергей Александрович*  
Решение уравнения Смолуховского с применением специальных представлений массивов 7

- Азиатцева Валерия Валерьевна*  
Моделирование вирусной инфекции с помощью детерминированного и стохастического подходов . . . . . 9

- Кузнецов Максим Алексеевич*  
Исследование и развитие технологий автогенерации вычислительного кода на GPU в применении к задачам тензорной аппроксимации. Реализация параллельного алгоритма ALS . . . . . 10

- Харюк Павел Васильевич*  
Параллельная MPI реализация Wavelet Tensor Train (WTT) разложения . . . . . 11

---

## КАФЕДРА НЕЛИНЕЙНЫХ ДИНАМИЧЕСКИХ СИСТЕМ И ПРОЦЕССОВ УПРАВЛЕНИЯ

---

- Роговский Александр Игоревич*  
Исследование свойств относительных порядков для векторных систем . . . . . 12

---

## КАФЕДРА ОБЩЕЙ МАТЕМАТИКИ

---

- Мокроусов Илья Сергеевич*  
Критерий принадлежности классу  $W_l^p$  обобщенного из класса  $L_p$  решения волнового уравнения . . . . . 13

---

## КАФЕДРА СИСТЕМНОГО АНАЛИЗА

---

- Зазаров Евгений Вадимович*  
Задачи оптимального управления с фазовыми ограничениями . . . . . 14

- Маянцев Кирилл Сергеевич*  
Аппроксимация множеств достижимости гибридных систем с линейной структурой и дискретным временем . . . . . 15

---

## КАФЕДРА АВТОМАТИЗАЦИИ СИСТЕМ ВЫЧИСЛИТЕЛЬНЫХ КОМПЛЕКСОВ

---

- Груздев Алексей Михайлович*  
Многомерная фильтрация в прогрессивном синтезе видео . . . . . 17

<i>Старцев Михаил Леонидович</i>	
Применение сенсора Kinect в задачах чтения по губам . . . . .	18
<i>Сигалов Даниил Алексеевич</i>	
Распознавание жестов рук с помощью сенсора Kinect . . . . .	19
<i>Моисеев Борис Дмитриевич</i>	
Генерация синтетических обучающих выборок для классификатора автодорожных знаков . . . . .	20
<i>Чистякова Ксения Юрьевна</i>	
Способ оценки эффективности методов распознавания для задач с многими классами	22
<hr/>	
КАФЕДРА АЛГОРИТМИЧЕСКИХ ЯЗЫКОВ	
<hr/>	
<i>Николаев Евгений Альбертович</i>	
Построение метрики в сети Интернет . . . . .	23
<i>Кулёва Анна Сергеевна</i>	
Модельный проблемно-ориентированный интерпретатор языка Лисп для анализа и синтеза гипертекстовых данных . . . . .	24
<i>Астабацян Карапет Араевич</i>	
Статистические методы определения эмоциональной направленности текста . . . . .	25
<i>Иванов Николай Игоревич</i>	
Применение метода опорных векторов для анализа речевых команд в приложении-помощнике для мобильных устройств . . . . .	27
<hr/>	
КАФЕДРА СИСТЕМНОГО ПРОГРАММИРОВАНИЯ	
<hr/>	
<i>Дудина Ирина Александровна</i>	
Разработка и реализация облачного планировщика, учитывающего топологию коммуникационной среды при высокопроизводительных вычислениях . . . . .	28

## Обработка текстовых изображений с использованием матрицы смежности

*Работа удостоена диплома I степени  
Хвостиков Александр Владимирович  
Кафедра математической физики  
e-mail: xubiker@gmail.com*

*Научный руководитель: д.ф.-м.н., проф. Крылов Андрей Серджемвич*

Применение аппарата матриц смежности для анализа и обработки изображений страниц печатного текста, позволяет рассматривать текст как единую текстуру и анализировать ее со статистической точки зрения, что может быть полезным в системе предварительной обработки изображений.

Целью работы была разработка алгоритма для обработки изображений, получаемых с камер портативных устройств. Так пользователь телефона при помощи специального приложения может делать снимки текста (оцифровывать книгу), при этом в режиме реального времени приложение осуществляет выравнивание страниц, определяет различные характеристики текста и оценивает степень размытости картинки. В случае, если качество изображения не является достаточно хорошим (изображение смазано или находится не в фокусе), приложение просит пользователя сделать снимок повторно. Кроме того, в ряде случаев программа может самостоятельно улучшить качество изображения.

Матрица смежности [1] (Grey Level Co-occurrence Matrix) строится для полутоновых изображений по фиксированному вектору  $\vec{d} = (dr, dc)$ , задающему смещение между пикселем яркостью  $i$  и пикселем яркостью  $j$ :

$$C_d [i, j] = |\{(r, c) \mid I[r, c] = i, I[r + dr, c + dc] = j\}|$$

Для компактного описания GLCM использовались числовые признаки [2,3], посчитанные для нормированной матрицы смежности  $N_d$ :

- Энергия:  $ASM = \sum_i \sum_j N_d^2 [i, j]$ ;
- Энтропия:  $ENT = - \sum_i \sum_j N_d [i, j] \log_2 N_d [i, j]$ ;
- Контрастность:  $CON = \sum_i \sum_j (i - j)^2 N_d [i, j]$ ;
- Однородность:  $HOM = \sum_i \sum_j \frac{N_d [i, j]}{1 + |i - j|}$ ;
- Корреляция:  $COR = \sum_{i,j} \frac{(i - \mu_i)(j - \mu_j)}{\sqrt{\sigma_i^2 \cdot \sigma_j^2}} \cdot N_d [i, j]$ ,  
 $\mu_i = \sum_j i \cdot N_d [i, j]$ ,  $\sigma_i^2 = \sum_j (i - \mu_i)^2 \cdot N_d [i, j]$ ;

В качестве исходных данных были выбраны изображения русского и английского текста с разными типами и размерами шрифтов. Общий объем выборки составил 550 изображений. В выборке присутствовали изображения разного качества: от «идеальных» (переведенная в растровый формат электронная книга) до «плохих» (снимки с телефона при слабом освещении).

По изображениям строились GLCM, вычислялись текстурные коэффициенты, анализировалось их поведение, исходя из которого делались выводы о различных параметрах изображения. Похожая методика использовалась в работе [4].

### Результаты

На основе поведения характеристик GLCM был разработан алгоритм, позволяющий по изображению страницы с текстом определить угол поворота текста, найти средние значения высоты кегля, толщины линии кегля, оценить размытость изображения, а также контролировать автоматическую корректировку качества изображения (используя метод *Unsharp Mask*).

Было разработано приложение для мобильной ОС Android, использующее ускоренную версию алгоритма. На данный момент в приложении реализованы функции нахождения угла поворота страницы и последующего выравнивания, нахождения высоты кегля, нахождения

толщины линии кегля. Приложение написано с использованием Java, C++, OpenCV и технологии JNI.

Тестирование показало, что приложение работает корректно и достаточно стабильно. Для обработки одного снимка требуется в среднем 2-4 секунды. Предварительные результаты исследования были представлены на XV Международной конференции «Цифровая обработка сигналов и её применение DSPA-2013» [5].

#### Литература

1. Шапиро Л., Стокман Дж. *Компьютерное зрение*. Изд-во: Бинوم. Лаборатория знаний, 2006
2. Haralick R., Shanmugam K., Dinstein I. *Textural Features for Image Classification* // IEEE Transactions on Systems, Man, and Cybernetics, 1973. Vol. SMC-3, No. 6, p. 610–621
3. Suresh A., Shunmuganathan K. *Image Texture Classification using Gray Level Co-Occurrence Matrix Based Statistical Features* // European Journal of Scientific Research, 2012. Vol. 75, No. 4, p. 591–597
4. Datta A., Dutta S. *Texture Analysis of Turned Surface Images using Grey Level Co-occurrence* // Advanced Materials Research, 2012. Vol. 365, p. 38–43
5. Хвостиков А. *Обработка текстовых изображений с использованием матрицы смежности* // Доклады XV Международной конференции «Цифровая обработка сигналов и её применение DSPA-2013». Москва, 2013. т. 2, с. 41–44

## Исследование математической модели стохастической динамики ВИЧ-инфекции

*Кислицын Алексей Алексеевич*

*Кафедра Вычислительных Технологий и Моделирования*

*e-mail: alexey.kislitsyn@gmail.com*

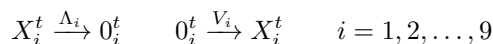
*Научный руководитель: д.ф.-м.н., в.н.с. ИВМ РАН Бочаров Геннадий Алексеевич*

В настоящее время Россия переживает одну из самых масштабных ВИЧ эпидемий в Европе, поэтому моделирование развития ВИЧ инфекции является одной из актуальных задач. Большинство математических моделей репликации вируса иммунодефицита человека не учитывают случайность процессов, протекающих в клетке во время от проникновения вируса в клетку до сборки и высвобождения вирусов. В данной курсовой работе эта проблема, перехода от детерминированного к стохастическому описанию, была решена для недавно разработанной модели внутриклеточной репликации ВИЧ, сформулированной в виде системы ОДУ [1].

Была воспроизведена детерминированная модель репликации вируса иммунодефицита человека в Т-клетках по результатам измерения концентраций 7991 маркеров экспрессии генов в клетке, описанные в недавно опубликованной работе зарубежных авторов [1,2]. Были промоделированы 9 этапов вирусной репликации в клетке, характеризующиеся активностью показателей Early RT (концентрация первичных компонент обратной транскриптазы), Late RT (концентрация поздних компонент обратной транскриптазы), 2LTR (концентрация компонент регулирующих транскрипционный цикл), Integrated (концентрация компонент интеграции), Multiply Spliced (концентрация многократно расщепляемых компонент), Singly Spliced (концентрация однократно расщепляемых компонент), Unspliced (концентрация нерасщепляемых компонент транскрипции), GFP-MFI (концентрация компонент зеленого флуоресцентного белка средней интенсивности), Release (концентрация компонент высвобожденного вируса из клетки). Уравнения динамики вирусных компонент описывались, с помощью системы ОДУ: где  $k, \theta$  – векторы констант, отвечающие за увеличение и уменьшение соответствующих концентраций.

Однако, процесс репликации вируса характеризуется относительно небольшим значением биохимических компонент и эффекты случайности дискретных численностей могут сказываться на динамике процесса. Для того чтобы учесть это в курсовой работе, была построена и

реализована стохастическая версия модели. Ее расчеты происходили с использованием стохастического алгоритма Гиллеспи [4]. Для стохастической модели репликации ВИЧ необходимо было определить ряд элементарных реакций, которые в данном случае имеют вид:



Расчеты производились в системе Matlab, с использованием алгоритма SSA [5]. Для каждого этапа репликации ВИЧ были построены 95% доверительные интервалы для траекторий стохастической модели. Для этапа Release (продукции вирусных частиц) в стохастической реализации модели была доказана устойчивость кинетики исхода вирусных частиц.

В дальнейшем планируется построить модель репликации вируса иммунодефицита человека высокого разрешения, с расширением класса рассматриваемых этапов. Также будут использованы большие возможности для внедрения параллелизма в алгоритм, что позволит повысить эффективность решения стохастической модели.

Работа была выполнена при поддержке гранта Программы Президиума РАН “Фундаментальные науки - медицине”.

### Литература

1. Mohammadi et al. *24 hours in the life of HIV-1* // PLoS, 2013.
2. Bartha I., Mohammadi P., Ciuffi A., Telenti A. *Patterns of Expression and Analysis of Clusters of HIV/Host interactions (PEACHi)* Available: <http://www.peachi.labtelenti.org>.
3. Dormand J. R., Prince P. J. *A family of embedded Runge-Kutta formulae* // J Comp Appl Math, 1980, т. 6, с. 19–26.
4. Daniel T. Gillespie *Stochastic Simulation of Chemical Kinetics* // Annu. Rev. Phys. Chem., 2007, т. 58, с. 35–55.
5. Ван Кампен Н. Г. *Стохастические процессы в физике и химии* М. : Высшая школа, 1990.
6. Charlebois D. *MATLAB Stochastic Simulation Algorithm (SSA) - Gillespie's Direct Method* 2011
7. Марчук Г. И. *Математические модели в иммунологии. Вычислительные методы и эксперименты* М. : Наука, 3-е изд., 1991.

## Решение уравнения Смолуховского с применением специальных представлений массивов

*Работа удостоена диплома II степени*

*Матвеев Сергей Александрович*

*Кафедра Вычислительных Технологий и Моделирования*

*e-mail: matveevserega@rambler.ru*

*Научный руководитель: чл.-корр. РАН д.ф.-м.н., проф. Тыртышников Евгений Евгеньевич*

Одна из важных задач механики — описание неупругого столкновения взаимодействующих частиц сложной системы. Именно неупругие соударения частиц порождают такие физические явления, как свёртывание крови, рост кристаллов, выпадение атмосферных осадков и даже образование звёзд. Механизм коагуляции(слияния) частиц — один из основных механизмов эволюции дисперсных систем [1].

Рассмотрим систему, состоящую из большого числа хаотически движущихся частиц, образованных из определённого набора веществ (компонент). Предположим, что частицы равномерно распределены в пространстве, и будем учитывать лишь их парные взаимодействия. Каждой из частиц можно приписать векторную характеристику, скалярными компонентами которой будут массы или объёмы веществ, входящих в её состав. Изменение во времени объёмов компонент, входящих в состав частиц, можно описать с помощью уравнения коагуляции Смолуховского.

$$\frac{\partial n(v_1 \dots v_d, t)}{\partial t} =$$

$$\frac{1}{2} \int_0^{v_1} du_1 \dots \int_0^{v_d} K(v_1 - u_1 \dots v_d - u_d; u_1 \dots u_d) n(u_1 \dots u_d, t) n(v_1 - u_1 \dots v_d - u_d, t) du_d$$

$$- n(v_1 \dots v_d, t) \int_0^\infty du_1 \dots \int_0^\infty K(v_1 \dots v_d; u_1 \dots u_d) n(u_1 \dots u_d, t) du_d,$$

где  $v_1, \dots, v_d$  – объёмы соответствующих компонент среды в частице,  $K(v_1, \dots, v_d; u_1, \dots, u_d)$  – ядро уравнения. Первый член уравнения описывает рост числа частиц с объёмами компонент  $v_1, \dots, v_d$  за счёт слияния с частицами объёмов  $v_1 - u_1, \dots, v_d - u_d$ , второй член – убыль частиц объёмов  $v_1, \dots, v_d$  за счёт слияния с частицами с объёмов  $u_1, \dots, u_d$ .

Распределение концентрации  $n(\bar{v}, t)$  будем искать как решение задачи Коши для уравнения Смолуховского при заданном начальном условии

$$n(\bar{v}, 0) = n_0(\bar{v}).$$

В данной работе на основе применения ТТ-крестовых аппроксимаций [3] и разностной схемы предиктор-корректор построен и реализован программно на языке C++ метод приближённого решения задачи Коши для d-мерного уравнения Смолуховского с арифметической сложностью шага разностной схемы  $O(d^3 N^2 r^6)$ , где d – размерность задачи, N – число узлов в сетке по объёмным компонентам, r – максимальный из рангов ТТ-разложения, используемых массивов. Проведены тесты сходимости численного решения, построенного данным методом, к аналитическому решению задачи Коши для двумерного уравнения Смолуховского [2]. Продемонстрирована эффективность построенного метода на мелких двумерных сетках. Показана эффективность метода при решении задач Коши для уравнения Смолуховского для многокомпонентных сред (до шести компонент).

Широкие возможности использования параллелизма при построении ТТ-крестовых аппроксимаций позволяют в дальнейшем ускорить поиск численного решения задач Коши для уравнения Смолуховского и провести тесты эффективности построенного метода в сравнении с методами Монте-Карло для численного решения задачи Коши для уравнения Смолуховского, напрямую моделирующими поведение коагулирующих частиц [4]-[5].

### Литература

1. Галкин В.А. *Уравнение Смолуховского* -М.: ФИЗМАТЛИТ, 2001
2. Lushnikov A. A. *Evolution of coagulating systems: III. Coagulating mixtures* J. Colloid Interface Sci., 54 (1976) 94.
3. I Oseledets, E Tyrtshnikov *TT-cross approximation for multidimensional arrays* Linear Algebra and its Applications 432 (1), 70-88 , 2008
4. Kruis, F. E., Maisels, A. and Fissan, H. *Direct simulation Monte Carlo method for particle coagulation and aggregation*. AIChE J. 2000, doi: 10.1002/aic.690460905
5. Palaniswamy Geethpriya , Loyalka Sudarshan K. *Direct simulation monte carlo aerosol dynamics : Coagulation and collisional sampling* Nuclear Technology / Volume 156 / Number 1 / October 2006 / Pages 29-38
6. А.Е. Алоян *Динамика и кинетика газовых примесей и аэрозолей в атмосфере / курс лекций* - М. : ИВМ РАН, 2002



## Моделирование вирусной инфекции с помощью детерминированного и стохастического подходов

*Работа удостоена диплома III степени*

*Азиатцева Валерия Валерьевна*

*Кафедра вычислительных технологий и моделирования*

*e-mail: valeryaaziattseva@yandex.ru*

*Научный руководитель: д.ф.-м.н., проф. Бочаров Геннадий Алексеевич*

При описании биологических процессов можно использовать как детерминированные [3], так и стохастические модели, причем полученные результаты могут сильно отличаться. Например, процессы, происходящие в популяциях малой численности имеют выраженный дискретный характер, и стохастические эффекты оказывают значительное влияние на их динамику. В данной работе мы рассматриваем математические описания вирусной инфекции с возможностью образования квазивидов и иммунным ответом со стороны Т-лимфоцитов. Детерминированная модель представлена системой обыкновенных дифференциальных уравнений, выписанных ниже [1], а стохастическая версия основана на применении алгоритма Гиллеспи [2].

$$\left\{ \begin{array}{l} \frac{dI_k}{dt} = \sum_{k'=1}^4 s_{k'} m_{k'k} I_{k'} - u I_k - p I_k \left( \sum_{i=1}^{N_D} \chi_k E_i \right); \\ \frac{dN_i}{dt} = -a N_i \left( \sum_{k=1}^4 \chi_k I_k \right) + b N_i - c N_i + w; \\ \frac{dE_1}{dt} = (a N_i + a' M_i) \left( \sum_{k=1}^4 \chi_k I_k \right) - r E_1 - d E_1 - g E_1; \\ \frac{dE_i}{dt} = -r E_i + 2r E_{i-1} - d E_i - g E_i; \\ \frac{dE_{N_D}}{dt} = -d' E_{N_D} - g E_{N_D} + 2r E_{N_D-1} \\ \frac{dM}{dt} = -a' M_i \left( \sum_{k=1}^4 \chi_k I_k \right) - h M_i + g \sum_{i=1}^{N_D} E_i \end{array} \right.$$

Для сравнения динамики численности зараженных клеток мы провели расчеты с использованием обеих моделей при одних и тех же начальных условиях и параметрах. При использовании детерминированной версии на семидесятый день после заражения общее количество зараженных клеток составило порядка  $10^6$  клеток. Распределение общей численности зараженных клеток по 41 реализации алгоритма Гиллеспи выглядит следующим образом:

- порядка  $10^4$  — 24 реализации;
- порядка  $10^5$  — 3 реализации;
- порядка  $10^6$  — 14 реализаций.

Отметим, что длительность одной реализации составила от 15 минут до нескольких часов, в зависимости от общего количества клеток в системе. Анализируя полученные результаты, можно сделать следующие выводы:

- Использование стохастического подхода показывает возможность существования различных вариантов течения инфекции, некоторые из которых значительно различаются. В нашей модели это связано со случайным характером возникновения мутаций в вирусе, поэтому наиболее приспособленный двухточечный мутант в некоторых реализациях так и не появился. Тем самым, продемонстрирована роль стохастических эффектов в динамике инфекции.
- В моделях, описывающих реальные инфекции необходимо учитывать большое количество клеток [4], и стохастическое описание таких процессов с использованием алгоритма

Гиллеспи характеризуется высокой вычислительной сложностью и требует значительных временных затрат. Поэтому дальнейшее развитие данного подхода может быть связано с эффективными реализациями алгоритма или использованием гибридных моделей.

### Литература

1. Elisabeth L. Read, Allison A. Towo-Dwyer, Arup K. Chacraborty *Stochastic effects are important in intrahost HIV evolution even when viral loads are high* P Natl Acad Sci USA 2012, 109(48):19727-32.
2. Daniel T. Gillespie *Stochastic simulations of chemical kinetics* Annu Rev Phys Chem 2007, 58:35-37.
3. Марчук Г. И. *Математические модели в иммунологии. Вычислительные методы и эксперименты* М.: Наука, 1991.
4. Цинкернагель Р. *Основы иммунологии* М.: Мир, 2008.

## Исследование и развитие технологий автогенерации вычислительного кода на GPU в применении к задачам тензорной аппроксимации. Реализация параллельного алгоритма ALS

Кузнецов Максим Алексеевич

Кафедра ВТМ

e-mail: earendillock@ya.ru

Научный руководитель: к.ф.-м.н., доц. Оселедец Иван Валерьевич

Для решения современных вычислительных задач необходимо использовать большие вычислительные мощности. Одним из наиболее эффективных вычислительных инструментов являются графические процессоры, однако, несмотря на развитие инструментов разработки, написание GPU-кода занимает достаточно много времени. Само по себе программирование на GPU — сложная задача, поэтому хотелось бы иметь технологии автоматического распараллеливания, однако они часто проигрывают «ручному» программированию. При этом можно выделить класс задач, которые допускают возможность автоматической генерации эффективного GPU-кода, а получающийся код можно использовать в динамических языках (Python). До недавнего времени удобных инструментов такого рода не было, но они стали интенсивно развиваться. Был взят за основу проект `loopy` (<http://git.tiker.net/loopy.git>), разработанный Андреасом Клекнером, который позволяет генерировать OpenCL Python-модули для небольших, но трудоемких циклов.

В качестве примера алгоритма аппроксимации тензора рассматривается алгоритм построения канонического разложения.

Каноническим разложением многомерного массива (*тензора*) называется представление вида

$$A(i_1, i_2, \dots, i_d) = \sum_{\alpha=1}^r U_1(i_1, \alpha) U_2(i_2, \alpha) \dots U_d(i_d, \alpha), \quad (1)$$

где  $U_k$  называются *факторами* канонического разложения, а  $r$  — каноническим рангом.

Уравнение (1) является основным. Подробнее о тензорах и их разложениях можно узнать в обзоре [1].

Опишем идею алгоритма ALS. Пусть задан тензор  $A$  с элементами  $A_{i_1 \dots i_d}$ . Задача состоит в том, чтобы найти его каноническое приближение, а именно найти такие матрицы  $U_1, \dots, U_d$

$$A_{i_1, \dots, i_d} \approx \sum_{\alpha=1}^r U_1(i_1, \alpha) U_2(i_2, \alpha) \dots U_d(i_d, \alpha). \quad (2)$$

Математическая постановка задачи состоит в том, чтобы решить задачу (2) в смысле наименьших квадратов

$$\sum_{i_1, \dots, i_d} \left( A(i_1, \dots, i_d) - \sum_{\alpha=1}^r U_1(i_1, \alpha) U_2(i_2, \alpha) \dots U_d(i_d, \alpha) \right)^2 \rightarrow \min. \quad (3)$$

Будем решать вариационную задачу поиска аппроксимации тензора с помощью алгоритма ALS (Alternating Least Squares), подробное изложение которого можно найти в статье [2]. Основная идея алгоритма, состоит в том, чтобы фиксировать все факторы канонического разложения, кроме одного, и искать минимум функционала только по нему. Путем циклических перестановок, используя уже полученные факторы, строятся последующие, до тех пор, пока не будет достигнута требуемая точность аппроксимации или, пока не сработают другие критерии остановки алгоритма (превышение максимального количества итераций, превышение времени выполнения программы).

Средствами автоматической генерации кода удобно воспользоваться для распараллеливания тензорных алгоритмов с использованием вычислительного потенциала графических процессоров, как в работе [3]. Ускорение работы программы ожидается существенным, как в силу высокой производительности GPU, так и благодаря структуре самого алгоритма. Тензорные алгоритмы широко востребованы, а создание их эффективной и быстрой реализации является одной из приоритетных задач, в то время как возможности автоматической генерации кода на GPU позволяют создать такую реализацию быстро.

На примере алгоритма ALS продемонстрированы возможности автоматического распараллеливания тензорных алгоритмов с использованием пакета `loopu`. Получено серьезное ускорение ALS, изучен способ автоматического распараллеливания тензорных алгоритмов.

#### Литература

1. Kolda T. G., Bader B. W. *Tensor decompositions and applications* SIAM Review, 2009
2. Carroll J. D., Chang J. J. *Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition* Psychometrika, 1970
3. Ballard G., Kolda T., Plantenga T. *Efficiently computing tensor eigenvalues on a GPU* //Parallel and Distributed Processing Workshops and Phd Forum (IPDPSW), 2011 IEEE International Symposium on. – IEEE, 2011. – С. 1340-1348.

## Параллельная MPI реализация Wavelet Tensor Train (WTT) разложения

Харюк Павел Васильевич

Кафедра вычислительных технологий и моделирования

e-mail: hariyuki@yandex.ru

Научный руководитель: к.ф.-м.н. Оселедец Иван Валерьевич

Классические вейвлеты играют важную роль в анализе сигналов различной природы. Вейвлет-преобразования основаны на наборах фильтров, которые гарантированно обеспечивают нулевые моменты соответствующего порядка. Однако, данные фильтры фиксированы, что накладывает существенное ограничение на число нулевых моментов для каждого отдельного вейвлета.

WTT-преобразование, в отличие от классических вейвлетов, вычисляет фильтры адаптивно, и представляет собой синтез вейвлет-подхода и стандартного ТТ-разложения. Данное преобразование используется для представления многомерных массивов в виде псевдо-разреженных в базисе некоторого набора фильтров. В силу того, что разложение массива, полученное в результате WTT-преобразования, содержит достаточное количество элементов, близких к нулю, данное преобразование потенциально применимо в задачах сжатия данных. WTT-преобразование можно применять для набора однородных данных, при этом подход к построению фильтров может быть разным - можно представить набор данных как один большой многомерный массив, и на его основе вычислить общие фильтры, можно выбрать часть

данных из этого набора и поступить аналогично, либо ограничиться отдельным представителем в качестве источника фильтров для всех данных.

В настоящее время вместе с ростом объёмов обрабатываемой информации растут также ресурсы и время, которые необходимо затратить на обработку данных. В связи с этим ростом вычислительные задачи часто решают с привлечением суперкомпьютерных вычислений. Идея данной работы состоит в том, чтобы реализовать параллельный алгоритм WTT-разложения и исследовать его масштабируемость. Алгоритм реализован на языке Python с использованием технологии MPI, исследование проводилось на основании тестовых запусков на суперкомпьютере «Ломоносов» для искусственных данных.

В силу того, что алгоритм поиска фильтров WTT-разложения нуждается в обменах данными на каждом шаге, не удалось достичь высоких показателей эффективности параллельной реализации данного процесса. С ростом числа процессов убывает эффективность и возрастают накладные расходы, особенно быстро для малых рангов разложения. Однако, для больших рангов скорости убывания эффективности и роста накладных расходов уменьшаются, и это позволяет сделать предположение о том, что можно значительно снизить указанные эффекты при согласовании ранга и числа процессов.

Реализованные алгоритмы прямого и обратного преобразования нуждаются только в одном обмене данными каждый, поэтому для них характерен рост эффективности и снижение накладных расходов, особенно для больших рангов. В случае прямого преобразования происходит это очень медленно и связано с тем, что в конце работы алгоритма происходят последовательные, а не параллельные вычисления. Обратное преобразование, наоборот, начинается с последовательной работы, а основная часть выполняется параллельно. В силу этой особенности для данного алгоритма наблюдаются лучшие показатели сильной масштабируемости и эффективности.

#### Литература

1. Оселедец И.В. *Вычислительные тензорные методы и их применения* Диссертация на соискание учёной степени УДК 519.6, 2011.
2. I. Oseledets, E. Tyrtyshnikov *TT-cross approximation for multidimensional arrays* Linear Algebra Appl., 432 (2010).

## Исследование свойств относительных порядков для векторных систем

*Роговский Александр Игоревич*

*Кафедра нелинейных динамических систем и процессов управления*

*e-mail: alexander.rogovskiy@gmail.com*

*Научный руководитель: м.н.с. Краев Андрей Владимирович*

Рассматривается линейная дискретная динамическая система общего положения:

$$\begin{cases} x^{t+1} = Ax^t + B\xi^t \\ y^t = Cx^t, \quad t=0,1,2,\dots \end{cases}$$

где  $x \in \mathbb{R}^n$ ,  $y \in \mathbb{R}^l$ ,  $\xi \in \mathbb{R}^n$ ,  $A \in \mathbb{R}^{n \times n}$ ,  $B \in \mathbb{R}^{n \times l}$ ,  $C \in \mathbb{R}^{l \times n}$

Вектор  $r = (r_1, r_2, \dots, r_l)$  называется вектором относительного порядка (далее ОП) для этой системы, если

$$\begin{aligned} 1) C_i B = 0, C_i A B = 0, \dots, C_i A^{r_i-2} B = 0, C_i A^{r_i-1} B \neq 0, \\ 2) \det H(r_1, \dots, r_l) = \det \begin{pmatrix} C_1 A^{r_1-1} B \\ \dots \\ C_l A^{r_l-1} B \end{pmatrix} \neq 0; \end{aligned}$$

где  $C_i$  – строки матрицы  $C$ ,  $i = 1, 2, \dots, l$ .

Известно (см. [1]), что робастное обращение системы возможно при выполнении определения относительного порядка Исидори (либо если система может быть приведена к виду

с ОП путем невырожденной замены выходов). Однако для корректности задачи обращения выполнение этого условия не требуется (см. [2]). Возникает вопрос, влечет ли выполнение условий корректности (системы, для которых эти условия выполнены, назовем для краткости «обращаемыми») выполнение также условий ОП Исидори.

В данной работе показано, что, в общем случае, ответ на этот вопрос отрицательный: существуют «обращаемые» системы, не имеющие ОП. Однако в некоторых частных случаях из корректности следует существование ОП. Доказано, что если у системы общего положения количество выходов (и входов) на единицу меньше количества состояний, то выполнение условий корректности у такой системы влечет существование у нее ОП. Также доказано, что приведение системы к виду с ОП Исидори возможно лишь невырожденной заменой выходов, замена входов в этом случае бесполезна (замена переменных пространства состояний также не меняет вектора ОП, что показано в [1]).

Предложен альтернативный способ введения относительного порядка, в котором роль строк матрицы  $C$  выполняют столбцы матрицы  $B$ . Показано существование «обращаемых» систем, не имеющих ни одного из указанных выше ОП.

### Литература

1. Ильин А. В., Коровин С. К., Фомичев В. В. *Методы робастного обращения динамических систем*. М.: ФИЗМАТЛИТ, 2009.
2. Краев А. В. *Необходимые условия обратимости линейных дискретных динамических систем*. // Дифференциальные уравнения, 2011, т. 47, № 4, с. 592–594.
3. Краев А. В. *Некоторые свойства относительного порядка линейных стационарных динамических систем*. // Нелинейная динамика и управление: сборник статей, 2013, вып. 8 под ред. С.В. Емельянова, с. 105–112.

## Критерий принадлежности классу $W_l^p$ обобщенного из класса $L_p$ решения волнового уравнения

*Мокроусов Илья Сергеевич*  
Кафедра общей математики  
e-mail: [mokill44@gmail.com](mailto:mokill44@gmail.com)

*Научный руководитель: д.ф.-м.н., профессор, академик РАН Ильин Владимир Александрович*

Главным предметом изучения данной курсовой работы являются задача для волнового уравнения с одной пространственной переменной

$$u_{tt}(x, t) - u_{xx}(x, t) = 0$$

Данное уравнение является линейным гиперболическим дифференциальным уравнением в частных производных, задающим малые поперечные колебания тонкой мембраны или струны, а также другие колебательные процессы в сплошных средах, таких как акустика, преимущественно линейная: звук в газах, жидкостях и твёрдых телах и электромагнетизм. Находит применение и в других областях теоретической физики, например при описании гравитационных волн.

Как и в работах [1], [2] будем рассматривать в прямоугольнике  $Q_T = [0 \leq x \leq L] \times [0 \leq t \leq T]$  обобщенное из класса  $L_p(Q_T)$  решение  $u(x, t)$  смешанной задачи для волнового уравнения

$$u_{tt}(x, t) - u_{xx}(x, t) = 0$$

с нулевыми начальными условиями

$$u(x, 0) = 0, \quad u_t(x, 0) = 0$$

и граничными условиями первого рода

$$u(0, t) = \mu(t), \quad u(L, t) = 0$$

второе из которых мы, не ограничивая общности, будем считать однородным.

Мы установим необходимые и достаточные условия на граничную функцию  $\mu(t)$ , обеспечивающие принадлежность рассматриваемого обобщенного из класса  $L_p(Q_T)$  решения  $u(x, t)$  классу  $W_p^1(Q_T)$ .

Ранее в [1], [2] был установлен критерий принадлежности классу  $W_p^1$  обобщенного из класса  $L_p$  решения волнового уравнения.

Основным результатом данной курсовой работы является:

**Теорема 1** Если при фиксированных  $T > 0$  и  $p \geq 1$  функция  $\mu(t)$  принадлежит классу  $L_1[0, T - \varepsilon]$  при любом  $\varepsilon$  из интервала  $0 < \varepsilon < T$ , то для принадлежности обобщенного из класса  $L_p(Q_T)$  решения  $u(x, t)$  смешанной задачи (см. выше) классу  $W_p^1(Q_T)$  необходимо и достаточно, чтобы граничная  $\mu(t)$  имела на полусегменте  $0 \leq t < T$  обобщенную производную  $\mu^{(l)}(t)$  и чтобы существовал интеграл

$$\int_0^T (T-t) |\mu^{(l)}(t)|^p dt$$

#### Литература

1. Ильин В. А., Кулешов А. А. Критерий принадлежности классу  $W_p^1$  обобщенного из класса  $L_p$  решения волнового уравнения. Доклады Российской Академии наук. 2012. Т 447, № 1, с. 15-17
2. Ильин В. А., Кулешов А. А. Критерий принадлежности классу  $L_p$  при  $p \geq 1$  обобщенного решения смешанной задачи для волнового уравнения. Дифференциальные уравнения. 2012. Т 48, № 12, с. 1607-1611
3. Ильин В. А., Кулешов А. А. О некоторых свойствах обобщенных решений волнового уравнения из классов  $L_p$  и  $W_p^1$  при  $p \geq 1$ . Дифференциальные уравнения. 2012. Т 48, № 11, с. 1493-1500
4. Ильин В. А., Кулешов А. А. Об определении обобщенного из класса  $L_p$  решения смешанной задачи для волнового уравнения через интегральное тождество. Доклады Российской Академии наук. 2012. Т 447, № 3, с. 247-251

## Задачи оптимального управления с фазовыми ограничениями

Работа удостоена диплома III степени

Захаров Евгений Владимович

Кафедра системного анализа

e-mail: justenterr@gmail.com

Научный руководитель: д.ф.-м.н., проф. Арутюнов Арам Владимирович

Курсовая работа посвящена задачам оптимального управления с фазовыми ограничениями:

$$\dot{x}(t) = f(x, u, t), \quad t \in [t_1, t_2], \quad x(t_1) = x_1, x(t_2) = x_2,$$

$$K^0(p) + \int_{t_1}^{t_2} f^0(x, u, t) dt \rightarrow \min,$$

$$u(t) \in U, \quad \text{для п.в. } t \in [t_1, t_2],$$

$$G(x(t), t) \leq 0, \quad \forall t \in [t_1, t_2],$$

$$K_1(p) = 0, \quad K_2(p) \leq 0, \quad p = (x_1, x_2, t_1, t_2)$$

Для данных задач в работе доказываются необходимые условия в виде принципа максимума Понтрягина в форме Дубовицкого и Милютина. Для линейных задач с выпуклыми фазовыми и концевыми ограничениями и интегральным функционалом показано, что принцип максимума, при выполнении некоторых дополнительных условий на множители Лагранжа, будет достаточным условием оптимальности.

Пусть  $G(x, t)$  — скалярная функция. Будем говорить, что допустимый процесс  $(p^*, x^*, u^*)$  является двусторонне управляемым в смысле [1], если  $\forall t : G(x^*(t), t) = 0$  существуют  $u'$  и  $u'' \in U$ , т.ч.:

$$\begin{aligned} \left\langle \frac{\partial G}{\partial x}(x^*(t), t), f(x^*(t), u', t) \right\rangle &> 0, \\ \left\langle \frac{\partial G}{\partial x}(x^*(t), t), f(x^*(t), u'', t) \right\rangle &< 0 \end{aligned}$$

В принципе максимума, доказанном в [3], в качестве множителя Лагранжа присутствует невозрастающая функция  $\mu$ . Были исследован вопрос непрерывности данной функции в условиях двусторонне управляемости — построен пример, когда в задаче существуют оптимальный двусторонне управляемый процесс, для которого любой соответствующий ему множитель Лагранжа  $\mu$  обязан иметь разрыв в момент выхода процесса на границу.

#### Литература

1. А. В. Арутюнов. *Условия экстремума. Аномальные и вырожденные задачи*. М.: Факториал, 1997.
2. А. В. Арутюнов, Г. Г. Магарил-Ильяев, В. М. Тихомиров. *Принцип максимума Понтрягина. Доказательство и приложения*. М.: Факториал Пресс, 2006.
3. A. V. Arutyunov, D. Yu. Karamzin, F. L. Pereira. *The Maximum Principle for Optimal Control Problems with State Constraints by R.V. Gamkrelidze: Revisited*. J. Optim. Theory Appl. (2011) 149.
4. В. И. Благодатских, А. Ф. Филиппов. *Дифференциальные включения и оптимальное управление*. Топология, обыкновенные дифференциальные уравнения, динамические системы, Сборник обзорных статей. 2. К 50-летию института, Тр. МИАН СССР, 169, 1985.
5. И. П. Натансон. *Теория функций вещественной переменной*. СПб.: Лань, 2008.
6. И. В. Грисанов. *Лекции по математической теории экстремальных задач*. Издательство Московского университета, 1970.
7. А. Н. Колмогоров, С. В. Фомин. *Элементы теории функций и функционального анализа*. М.: ФИЗМАТЛИТ, 2012.
8. Ф. П. Васильев. *Методы оптимизации*. М.: Факториал Пресс, 2002.

## Аппроксимация множеств достижимости гибридных систем с линейной структурой и дискретным временем

*Работа удостоена диплома I степени*

*Маянцев Кирилл Сергеевич*

*Кафедра системного анализа*

*e-mail: kirill.mayantsev@yandex.ru*

*Научный руководитель: к.ф.-м.н., асс. Точилин Павел Александрович*

В работе рассматривается математическая модель сложной системы, относящаяся к классу гибридных. Для нее характерно сочетание непрерывной динамики, описываемой при помощи дифференциальных или разностных уравнений, а также дискретных правил для моментального, скачкообразного изменения законов движения. Такие смены режимов функционирования, называемые переключениями, могут реализовываться при определенных условиях, в

зависимости от текущего состояния. Совокупная система обладает сложной, нелинейной динамикой даже при очень простых уравнениях, описывающих отдельные ее компоненты.

Математические модели гибридных систем активно изучаются в связи с различными приложениями ([2], [3], [4], [5]). Для таких моделей актуальным является решение разнообразных классических задач теории управления: проблем достижимости, разрешимости, верификации, синтеза управлений.

В данной курсовой работе рассматривается частный случай математической модели гибридной системы, в которой непрерывная компонента задана при помощи набора разностных уравнений с дискретным временем:

$$\begin{cases} x(k+1) = A_j(k)x(k) + B_j(k)u(k), \\ x(k) \in \mathbb{R}^{n_x}, u(k) \in \mathbb{R}^{n_u}, \\ A_j(k) \in \mathbb{R}^{n_x \times n_x}, B_j(k) \in \mathbb{R}^{n_x \times n_u}. \end{cases}$$

Здесь  $u(k)$  — вектор управляющих параметров, на который накладываются некоторые ограничения,  $j$  — номер «активной» подсистемы.

Замена одной системы таких уравнений на другую (переключение) происходит при пересечении траекторией текущей подсистемы определенной гиперплоскости в фазовом пространстве. Предполагается, что переключения являются обязательными и не сопровождаются дополнительными скачкообразными изменениями вектора фазовых переменных. Каждая из систем разностных уравнений включает управляющие параметры, на возможные значения которых наложены «жесткие» ограничения. Для такой системы решаются задачи достижимости и разрешимости на конечном интервале времени, при любом допустимом количестве переключений.

Целью работы является теоретическое исследование структуры множеств и трубок достижимости и разрешимости гибридной системы, получение формул, позволяющих рассчитывать отдельные «ветви» таких множеств. Стоит отметить, что каждая «ветвь» представлена в виде суперпозиции конечного набора многозначных операторов, результат действия каждого из которых может быть аппроксимирован, например, при помощи многогранника.

Основная часть работы посвящена исследованию конкретного примера гибридной системы на плоскости, получающейся при моделировании движения математического маятника с возможными препятствиями, играющими роль дополнительных точек подвеса маятника. Для этого примера была изучена структура множеств достижимости. В работе предлагается использовать подход, основанный на внешней или внутренней аппроксимации частей искомых множеств при помощи многоугольников. Одно из ключевых требований к таким многоугольникам, позволяющее контролировать сложность вычислений — это ограничение на максимальное количество их вершин. Для данного типа аппроксимаций был разработан эффективный алгоритм для расчета множеств достижимости. Итоговым результатом работы является написанная программа, позволяющая проводить численные расчеты множеств достижимости для рассматриваемого примера гибридной системы, а также строить иллюстрации методами компьютерной графики.

### Литература

1. Кормен Т., Лейзерсон Ч. *Алгоритмы: построение и анализ*. М.: МЦНМО, 1999.
2. Куржанский А. Б., Варайя П. *Задачи динамики и управления в гибридных системах*. // Труды международного семинара «Теория управления и теория обобщенных решений уравнения Гамильтона–Якоби». Екатеринбург: изд-во Уральского университета, стр. 21–37, 2005.
3. Куржанский А. Б., Точилин П. А. *Слабо инвариантные множества гибридных систем*. // Дифференциальные уравнения, т. 44, №11, с. 1523–1533, 2008.
4. Branicky M. S., Borkar V. S., Mitter S. M. *A unified framework for hybrid control: model and optimal control theory*. // IEEE transactions on automatic control. Vol. 43, no. 1, pp. 31–45, Jan 1998.



5. Schaft A. J. van der, J. M. Schumacher. *An Introduction to Hybrid Dynamical Systems. Lecture Notes in Control and Information Sciences*. Springer, №251, 2000.

## Многомерная фильтрация в прогрессивном синтезе видео

Груздев Алексей Михайлович

Кафедра автоматизации систем вычислительных комплексов

e-mail: [aleks-gruzdev92@yandex.ru](mailto:aleks-gruzdev92@yandex.ru)

Научные руководители: к. ф.-м. н. Игнатенко Алексей Викторович, аспирант ИПМ  
им. М.В. Келдыша Фролов Владимир Александрович

Метод Монте-Карло является одним из самых распространённых методов в компьютерном моделировании задач физики, химии, математики, экономики, оптимизации и др. Накладывая минимальные ограничения на исходные данные, метод Монте-Карло позволяет моделировать как статические, так и динамические процессы с любой точностью. Но серьёзным недостатком метода является его вычислительная сложность.

В курсовой работе рассмотрен прогрессивный (постепенно улучшающий результат) алгоритм синтеза видео последовательности на основе многомерной фильтрации. По сравнению с существующими подходами, метод предлагает принципиально новый и значительно более удобный способ контроля над процессом рендеринга при создании анимации и способен значительно (до 10 раз) ускорить синтез фотореалистичных видео последовательностей.

В применении к синтезу видеопоследовательности метод Монте-Карло [1] вызывает шум при смене кадров за счёт случайности выборки. Любой фильтр, обрабатывающий кадры независимо, не способен устранить данный тип шума, в то время как предложенный многомерный фильтр обрабатывает группы соседних кадров. Тем самым, учитывая временную близость, метод устраняет шум между кадрами.

Трассировка путей Монте-Карло обладает высокой вычислительной сложностью, и синтез видео с высокой точностью требует длительного времени. В современной индустрии синтезируются и обрабатываются изображения в несколько мегапикселей, таким образом, расчет одного кадра будет занимать несколько часов в лучшем случае. При этом любой результат расчета будет доступен только после завершения вычислений. В курсовой работе предлагается использовать прогрессивный подход к синтезу [3], позволяющий получить некоторый «грубый» результат сравнительно быстро и, затем, постепенно улучшать качество видео. Причём результат вычисления может быть отображён интерактивно и остановлен в любой момент времени, когда желаемое качество достигнуто.

Используя больше информации о каждом пикселе помимо его цвета, предложенный метод достигает высокого качества фильтрации, сравнимого с передовыми алгоритмами фильтрации [2], что в свою очередь значительно ускоряет скорость сходимости прогрессивного процесса синтеза (до 10 раз).

Принимая во внимание высокую скорость предложенного фильтра и высокую скорость трассировки малого числа путей (1-4 пути), предполагается, что рассмотренный алгоритм будет работать в реальном времени при эффективной реализации на GPU.

### Литература

1. Edwards D. *Carlo Integration* <http://www.cs.utah.edu/~edwards/research/mcIntegration.pdf>
2. Gastal E. and Oliveira M. *Adaptive manifolds for real-time high-dimensional filtering* In ACM SIGGRAPH 2012 vol 31, issue. 4, July 2012
3. Schwenk K., Kuijper A., Behr J., Fellner D.W. *Practical noise reduction for progressive stochastic ray tracing with perceptual control* In IEEE Computer Graphics and Applications, vol 32, issue 6, pp 46-55. Nov.-Dec. 2012.

## Применение сенсора Kinect в задачах чтения по губам

*Работа удостоена диплома I степени*

*Старцев Михаил Леонидович*

*Кафедра Автоматизации Систем Вычислительных Комплексов*

*e-mail: mikhail.startsev@gmail.com*

*Научный руководитель: к.ф.-м.н. Баяковский Юрий Матвеевич*

Чтение по губам - способ понимания речи, интерпретируя поступающую визуальную информацию (в частности, движение лица и губ). Звуковая информация недоступна, ненадёжна (или считается таковой в рамках поставленной задачи).

Особенность задачи чтения по губам состоит в требовательности к точности распознавания точек губ, что является некоторой проблемой в случае сенсора Kinect, так как пространственная зона, для которой надёжно работает разметка лица, достаточно удалена от сенсора (порядка 1.5 – 3 метров).

Более формально, поставлена задача распознавания отдельных слов конечного (небольшого) словаря для одного спикера. В качестве словаря взяты цифры (такой подход используется во многих статьях [3][4], поэтому по результатам можно будет понять, имеет ли смысл применение Kinect вместо других сенсоров с более высоким разрешением цветного изображения, но, например, без карты глубины на выходе).

Предложенный в курсовой работе метод решения поставленной задачи можно разбить на следующие этапы:

1. Разметка лица

2. Сбор базы для использования на дальнейших этапах.

*Собранная база состоит из не менее десяти последовательностей размеченных кадров для каждой цифры (всего 135 записей). Суммарно база содержит более пяти тысяч размеченных контуров губ.*

3. Нормализация данных

4. Метод главных компонент (МГК)

*Для повышения устойчивости метода и сокращения размерности пространства (для достаточно точного восстановления контура требуется 3-4 первых главных компоненты).*

5. Разбиение пространства на кластеры

*Вместо самого контура на этап обучения скрытой марковской модели подаётся номер ближайшего кластера в качестве наблюдаемого состояния («символа») порождаемой моделью последовательности.*

6. Обучение скрытой марковской модели для каждой цифры

*В качестве модели выбрана скрытая марковская модель (СММ), как широко используемая в области распознавания речи (см. [1]), а так же в чтении по губам, например в подходах, описанных в статьях [2] и [3].*

*В предложенном методе для каждой цифры обучается алгоритмом Витерби (описан, например, в [1]) эргодическая (возможны переходы из любого состояния в любое) модель. При классификации последовательности размеченных контуров выбирается цифра с наибольшей вероятностью порождения её моделью данной последовательности.*

7. Скользящий контроль для оценки качества классификации и выбора лучших параметров метода.

Программа для записи и разметки базы, обучение скрытых марковских моделей, классификация и скользящий контроль реализованы на языке C#. Обработка (нормализация, метод главных компонент и т.п.) и визуализация данных в основном производилась на языке

*R.* Проведённые тесты производительности классификации показывают, что среднее время обработки одной последовательности из базы составляет приблизительно 0.2 мс.

Преимущество используемого подхода состоит, в частности, в простоте реализации и возможности быстро дополнить уже существующую модель. Также стоит отметить, что понижение мощности словаря может дать ощутимый прирост точности классификации.

### Литература

1. L. R. Rabiner, *A tutorial on hidden Markov models and selected applications in speech recognition*, 1989.
2. Yuxuan Lan, Barry-John Theobald and Richard Harvey, *View Independent Computer Lip-Reading*, School of Computing Sciences University of East Anglia Norwich, UK // 2012 IEEE International Conference on Multimedia and Expo. P. 432–437.
3. S. L. Wang, A. W. C. Liew, W. H. Lau, and S. H. Leung, *An Automatic Lipreading System for Spoken Digits With Limited Training Data*, IEEE Transactions on Circuits and Systems for Video Technology, Vol. 18. No. 12. December. P. 1760–1765.
4. Meng Li and Yiu-ming Cheung, *A Novel Motion Based Lip Feature Extraction for Lip-reading*, 2008 International Conference on Computational Intelligence and Security. P. 361–365.

## Распознавание жестов рук с помощью сенсора Kinect

*Работа удостоена диплома II степени*

*Сигалов Даниил Алексеевич*

*Кафедра АСВК*

*e-mail: dsigalov@mail.ru*

*Научный руководитель: к.ф.-м.н., доцент Баяковский Юрий Матвеевич*

Распознавание жестов рук с помощью компьютера давно является объектом интереса многих исследователей, являясь частью так называемого «естественного» интерфейса взаимодействия машины и человека. Качественное распознавание позволило бы поднять на новый уровень как развлечения, так и управление приносящими реальную пользу приложениями или упрощенную передачу им данных. В качестве конкретного примера применения можно привести распознавание языка жестов немых.

В последнее время внимание к распознаванию жестов пережило новый всплеск в связи с появлением в широкой доступности сенсоров глубины, позволяющих получить информацию о трехмерном строении наблюдаемых объектов. Уже сейчас разработаны и применяются методы, позволяющие с помощью таких устройств получать информацию о движениях в масштабе всего тела, таких, как движениях головы, ног, корпуса, движений всей рукой в целом.

Проблемой остается распознавание жестов, выражаемых более мелкими частями тела, такими, как пальцы рук. Эти проблемы вызваны прежде всего низкой разрешающей способностью сенсоров на таком масштабе. В данной работе приводится метод, решающий описанную проблему для задачи распознавания статических жестов, т. е. распознавания жестов по одному кадру. В качестве сенсора используется сенсор Kinect от корпорации Microsoft.

Распознавание производится в несколько этапов:

1. На изображении находится точка  $P$ , расположенная на ладони руки. Координаты этой точки получаются с сенсора Kinect вместе с цветным изображением и картой глубины.
2. Выделяется часть изображения, соответствующая ладони. Это делается с помощью отсечения на карте глубины точек, отдаленных от точки  $P$  больше, чем на заданный порог.
3. Границей ладони будем считать границу найденной выше области, а центром ладони - точку области, наиболее удаленную ото всех точек границы.
4. Удалим из рассмотрения все точки области с расстоянием до центра ладони меньшим, чем определенное значение  $R$ . Для оставшихся после удаления связанных компонент,

которые должны соответствовать пальцам, возьмём три характеристики - нормализованную площадь компоненты  $w_i$ , ее начало  $r_{ia}$  и конец  $r_{ib}$ . Начало и конец определяются следующим образом: выделяется часть границы ладони, ограничивающая данную компоненту и представляется в полярной системе координат с центром в центре ладони; берутся углы точек, на которых выделенная часть контура обрывается, минимальный из взятых углов назовем началом компоненты  $r_{ia}$ , максимальный - концом  $r_{ib}$ . Углы приводятся к диапазону  $[0..1]$ .

5. Таким образом, признаком жеста будет набор из триплетов  $(w_i, r_{ia}, r_{ib})$ . Метрикой между такими наборами будет Finger-Earth Mover's Distance (FEMD) — модифицированная Earth Mover's Distance [4]. В [1] дается определение FEMD, а также показывается, что такая метрика будет устойчива к искажениям, вызванным низким разрешением сенсора.
6. Для распознавания выделяется несколько классов жестов, для каждого класса берется набор характерных представителей. Класс, до представителей которого расстояние по данной метрике оказалось минимальным, будет считаться классом распознаваемого жеста. В случае, если на изображении присутствуют две руки, распознавание производится независимо.

Предложенный алгоритм был полностью реализован в виде набора функций для MATLAB. Проведены тесты эффективности указанного метода.

Данный метод может быть использован как самостоятельно, так и в составе более сложной системы распознавания жестов.

#### Литература

1. Ren Z., Yuan J., Zhang Z. *Robust Hand Gesture Recognition based on Finger-Earth Mover's Distance with a Commodity Depth Camera* Proc. of ACM Multimedia 2011. P. 1093-1096.
2. Raheja J. L., Chaudhary A., Singal K. *Tracking of Fingertips and Centres of Palm using KINECT* Proc. of the 2011 Third International Conference on Computational Intelligence, Modelling & Simulation. P. 248-252.
3. Grauman K., Darrell T. *Fast contour matching using approximate earth mover's distance* CVPR. 2004. P. I:220-227.
4. Rubner Y., Tomasi C., Guibas L. J. *The earth mover's distance as a metric for image retrieval*. IJCV, 40. 2000. P. 99-121.
5. Freeman W. T., Roth M. *Orientation Histograms for HandGesture Recognition* IEEE Intl. Wkshp. on Automatic Face and Gesture Recognition, Zurich, June. 1995.P. 296-301.

## Генерация синтетических обучающих выборок для классификатора автодорожных знаков

*Работа удостоена диплома I степени*

*Моисеев Борис Дмитриевич*

*Кафедра автоматизации систем вычислительных комплексов*

*e-mail: bdmoiseev@gmail.com*

*Научный руководитель: к.ф.-м.н. Конушин Антон Сергеевич*

Большинство современных методов компьютерного зрения основаны на алгоритмах машинного обучения. Такие алгоритмы нуждаются в обучающей выборке, во многих случаях — значительного размера. Часто получение такой выборки может быть затруднительно или даже невозможно. Эта проблема может быть решена путем генерации синтетической обучающей выборки.

Синтетические данные применяются в самых разных областях компьютерного зрения: реконструкции стерео, распознавании рукописного и печатного текста, лица или позы человека, детекции людей на фотографиях, распознавании автодорожных знаков. Все использующиеся

методы можно разделить на два класса: методы, основанные на создании 3D-модели (например, распознавание лица и позы человека), и методы, основанные на использовании двумерных пиктограмм (распознавание текстов, автодорожных знаков и т.д.).

В качестве области для исследования эффективности применения синтетических обучающих выборок была выбрана задача распознавания автодорожных знаков. Современные методы её решения нуждаются в больших обучающих выборках, сбор которых требует значительных ресурсов. В результате был разработан алгоритм генерации синтетических изображений, основанный на комплексном подходе: сначала на основе пиктограммы знака создается трехмерная модель, а затем двумерное изображение модели подвергается различным трансформациям, моделирующим осложняющие распознавание факторы (плохое и неравномерное освещение, некачественная камера и т.п.)

Разработанный алгоритм состоит из следующих основных шагов:

- Выбор пространственного положения знака — высота, расстояние до наблюдателя и угол относительно направления обзора наблюдателя;
- Выбор угла поворота знака — реальный знак может быть повернут вокруг произвольной оси как вследствие ошибок при установке, так и из-за изменения направления дороги;
- Проецирование знака на плоскость камеры;
- Смещение относительно границ изображения и изменение масштаба — подобное возникает при недостаточно точной работе детектора;
- Получение примеров фона из набора пейзажных фотографий и наложение на него полученного ранее изображения знака;
- Изменение яркости и контраста изображения;
- Размытие по Гауссу — моделирование плохой фокусировки камеры или не очень чистого объектива;
- Шум по Гауссу — моделирование шума от камеры;
- Добавление локальных дефектов знака.

Параметры для описанных преобразований выбираются случайным образом в рамках установленных опытным путем границ.

Для оценки качества предложенного алгоритма проводилось сравнение результатов работы классификаторов, обученных на синтетических и реальных данных. В качестве реальных данных использовались три базы фотографий из Германии, Бельгии и Швеции. Классификаторы с невысокой обобщающей способностью (LDA и SVM) показали более высокую точность классификации при обучении на реальных данных (93.28% и 95.7% против 43.6% и 79.01% соответственно), что можно объяснить более высокой изменчивостью синтетических данных. Классификаторы с хорошей обобщающей способностью (метод k ближайших соседей и сверточная искусственная нейронная сеть) показали лучшую точность при обучении на реальных данных (96.91% и 97.87% против 72.81% и 96.3% соответственно), из чего можно сделать вывод, что предложенный алгоритм с успехом применим для создания обучающих выборок для классификаторов автодорожных знаков.

Также была проведена исследовательская работа в малоизученной на данный момент области сравнения похожести синтетических и реальных изображений. Сравнение осуществлялось с помощью алгоритма оценки зашумленности фотографий из статьи [1]. В результате было установлено, что синтетические данные, имеют меньшую зашумленность и, следовательно, отличаются от реальных фотографий и нуждаются в доработке.

Таким образом, в рамках курсовой работы был составлен обзор методов генерации синтетических обучающих выборок, предложен и реализован алгоритм генерации синтетических изображений автодорожных знаков, а также была показана применимость предложенного алгоритма для задачи классификации автодорожных знаков. На основе работы была написана статья, принятая к публикации на международной конференции ACIVS. Возможными

направлениями развития работы являются разработка методов определения оптимальных значений параметров предложенного алгоритма, а также разработка и применение различных метрик для сравнения реальных и синтетических изображений.

### Литература

1. Min Goo Choi, Jung Hoon Jung, and Jae Wook Jeon, *No-Reference Image Quality Assessment using Blur and Noise*. International Journal of Electrical and Electronics Engineering, 2009.

## Способ оценки эффективности методов распознавания для задач с многими классами

*Работа удостоена диплома II степени*

*Чистякова Ксения Юрьевна*

*Кафедра автоматизации систем вычислительных комплексов*

*e-mail: ksjuchi@gmail.com*

*Научный руководитель: д.ф.-м.н., Сенько Олег Валентинович*

Задачи распознавания возникают в различных областях человеческой деятельности. С возникновением и развитием алгоритмов классификации возникает потребность в инструменте, позволяющем регулировать точности распознавания по каждому из классов для фиксированного распознающего оператора [1]. Это позволит:

1. достичь приемлемой точности распознавания даже для тех классов, на которых классификатор часто ошибается
2. распознавать объекты некоторого класса с заданной точностью
3. сравнивать между собой разные распознающие операторы

Данная задача тесно связана с ROC-анализом — техникой визуализации результатов работы алгоритмов распознавания и выбора классификаторов на основе их эффективности [3]. ROC-анализ широко изучен для задач с двумя классами и получил большое распространение в практических приложениях, позволяя достичь для них хороших результатов. Целью курсовой работы являлось обобщение техник ROC-анализа для задач с произвольным числом классов.

Центральным объектом ROC-анализа является ROC-кривая [3]. В обобщённом (при числе классов более 2) ROC-анализе вместо ROC-кривых рассматриваются ROC-поверхности. Обладая данными о ROC-поверхности классификатора, можно выбрать набор параметров, который обеспечивает наиболее приемлемую точность для конкретной задачи. В курсовой работе был предложен универсальный алгоритм построения ROC-поверхности для произвольного числа классов с экспоненциальной по количеству классов оценкой временной сложности. Также для частного случая 3 классов был разработан эффективный алгоритм, который является комбинацией универсального алгоритма построения ROC-поверхности и алгоритма для задач с двумя классами, который впервые был предложен в [3].

Общепринятым методом оценки эффективности классификатора является вычисление площади под графиком ROC-кривой (AUC, Area under curve) [3]. Для задач с тремя классами обобщением AUC является объем под ROC-поверхностью (VUS, Volume under surface). В курсовой работе был предложен алгоритм приближенного вычисления VUS, основной идеей которого является разбиение ROC-поверхности на слои по одной из координат.

Для некоторых задач может требоваться, чтобы результаты работы классификатора были близки к точке баланса, которая соответствует максимально близким друг к другу точностям распознавания по каждому из классов. Помимо того, что нахождение точки баланса важно для настройки работы классификатора, также на основе сравнения точек баланса для разных классификаторов могут делаться выводы об их эффективности относительно друг друга. В курсовой работе были рассмотрены 3 различных алгоритма построения точки баланса. Простейший из них — перебор всех точек ROC-поверхности. Однако этот способ, хоть

и находит точное значение точки баланса для заданного распознающего алгоритма и выборки, обладает медленной скоростью работы, поскольку требует предварительного построения ROC-поверхности. Для увеличения быстродействия были предложены алгоритмы, которые позволяют находить приближенное значение точки баланса без построения ROC-поверхности. В первом из рассмотренных алгоритмов, приближенное решение ищется методом многомерного бинарного поиска [2]. По каждой компоненте производится стандартный бинарный поиск, подзадачей которого является многомерный бинарный поиск меньшей размерности. Для нахождения точки баланса также использовался численный метод, основанный на многократной генерации случайных величин, называемый методом Монте-Карло. Для сравнения рассмотренных алгоритмов было проведено их тестирование на химических выборках, при этом значения точек баланса для разных алгоритмов поиска оказались близкими друг к другу.

Алгоритмы были реализованы в виде программы на языке C++ и вспомогательных скриптов для MATLAB для графического отображения полученных результатов.

Таким образом, в рамках данной курсовой работы алгоритмы построения ROC-кривой и вычисления AUC из области ROC-анализа были обобщены на задачи с тремя и более классами. Были предложены алгоритмы поиска точки баланса. Данные алгоритмы могут быть использованы в качестве вспомогательного средства в машинном обучении для регулирования точности распознавания, а также сравнения эффективности различных моделей распознавания.

### Литература

1. Журавлев Ю. И. *Об алгебраическом подходе к решению задач распознавания или классификации. Проблемы кибернетики*. М.: Наука, 1978.
2. Левитин А. В. *Алгоритмы: введение в разработку и анализ*. М.: Мир, 1999.
3. Fawcett T. *An introduction to ROC analysis*. Pattern Recognition Letters 27 (2006) 861-874.

## Построение метрики в сети Интернет

*Николаев Евгений Альбертович*  
*Кафедра алгоритмических языков*  
*e-mail: evgeny.albertovich@gmail.com*

*Научный руководитель: к.ф.-м.н., доц. Абрамов Владимир Геннадьевич*

В настоящее время набирают популярность сети CDN (Content Delivery Network). Сеть CDN представляет собой географически распределенную сетевую инфраструктуру. Цель ее использования состоит в том, чтобы как можно быстрее доставить контент до пользователя. Основу сети составляет система edge-серверов. При запросе пользователя, контент из источника записывается на «ближайший» к этому пользователю edge-сервер. При этом edge-сервер считается «ближайшим» в том случае, если пользователь может получить контент с этого сервера быстрее, чем с других edge-серверов. Затем контент кэшируется на этом сервере, а все остальные пользователи, по отношению к которым этот edge-сервер также является «ближайшим», будут получать уже не контент с источника, а закэшированный контент с edge-сервера. Таким образом, встает задача определения «ближайшего» edge-сервера к пользователю. Для определения «ближайшего» сервера необходимо определить метрику в сети Интернет, то есть ввести функцию «близости».

В данной работе для построения функции «близости» были использованы методы регрессионного анализа. Поиск функции осуществлялся в виде полинома Колмогорова — Габора

первой степени:  $a_0 + \sum_{i=1}^n a_i x_i$ , где

$a_i$  (при  $i = 0, 1, \dots, n$ ) — искомые коэффициенты,

$x_i$  (при  $i = 1, 2, \dots, n$ ) — некоторые параметры, которые мы считаем известными заранее.

В качестве параметров были выбраны число скачков от edge-сервера до подсети пользователя, а также время от момента отправки запроса до момента получения ответа. При этом множество параметров было расширено с помощью нескольких специальных функций преобразования:  $\exp(-x)$ ,  $\sin x$ ,  $\cos x$ ,  $\frac{1}{x}$ ,  $\sqrt{|x|}$ , с целью уменьшения неопределенности линейной модели.

Настройка коэффициентов производилась по обучающей выборке, затем на контрольной выборке оценивалась ошибка, выбирались лучшие значения коэффициентов. Обучающие значения для функции «близости» выбирались на основании скорости скачивания группой пользователей из разных регионов страны специального файла со всех edge-серверов.

Затем была проведена оценка полученного результата при помощи данных, полученных другим способом. На страницу популярного сайта был внедрен специальный скрипт, который осуществлял скачивание файла небольшого объема с некоторых edge-серверов, а затем записывал скорость скачивания в специальный файл. Таким образом, был получен гораздо больший набор данных, позволяющий оценить полученный ранее результат.

В результате проделанной работы были построены различные варианты функции «близости», приведены оценки ее ошибки. Также была показана непротиворечивость данных, использованных в качестве обучающих значений функции.

### Литература

1. Ходасевич Г. Б. *Обработка экспериментальных данных на ЭВМ: обработка одномерных массивов*. СПб.: СПбГУТ, 2008.
2. Воронцов К. В. *Машинное обучение (курс лекций)*, 2009.

## Модельный проблемно-ориентированный интерпретатор языка Лисп для анализа и синтеза гипертекстовых данных

*Работа удостоена диплома III степени*

*Кулёва Анна Сергеевна*

*Кафедра алгоритмических языков*

*e-mail: akuleva1993@gmail.com*

*Научный руководитель: к.ф.-м.н., доц. Столяров Андрей Викторович*

Работа посвящена разработке аппарата, позволяющего обработку XML-документов и «строящихся на лету» XML-деревьев.

Как известно, XML-тег состоит из имени, списка атрибутов (возможно пустого) и содержимого (возможно пустого). В качестве элемента содержимого может выступать либо строка, либо XML-тег. XML-тег имеет древовидную структуру, следовательно для его анализа наилучшим образом подходит функциональная парадигма.

Язык Лисп (LISP — LISt Processing Language) стимулирует функциональное мышление. Основой лисповских структур данных является s-выражение. В простейшем случае — это атом (множество атомов состоит из символов, множества чисел и специальных символов T и NIL), а в более сложном — это список. Подробнее о языке Лисп и его структурах данных можно узнать в пособии [1]. Благодаря такому представлению данных, из всех языков отвечающих функциональной парадигме, Лисп больше всего подходит для решения задачи анализа XML-тегов.

Существует множество проектов, тесно связанных с XML, основным языком которых является C++. Например, на этом языке написан популярный браузер Google Chrome.

Для того чтобы совместить удобства упомянутых выше языков, в данной работе используется библиотека IntelLib. С помощью IntelLib программист получает возможность в коде на языке C++ делать вставки синтаксически близких к языку Лисп выражений, не включая в проект сам язык Лисп.

С помощью библиотеки IntelLib были разработаны средства для представления XML-тегов и XML-атрибутов в виде s-выражений. Так же в этой библиотеке имеется интерпретатор языка Лисп, который пользователь может дополнить своими функциями. Из вышесказанного следует, что с помощью библиотеки IntelLib, проблемно-ориентированных функций, нацеленных на обработку XML-тегов, и языка Лисп получается средство для анализа XML-деревьев.

Существуют ситуации, когда нужно работать с «недостроенным» XML-деревом. Такая ситуация возникает, например, при обработке потока информации, получаемого по протоколу



XMPP (eXtensible Messaging and Presence Protocol). В такой ситуации XML-тег следует рассматривать с двух позиций: когда он только появился в потоке и, возможно, еще не построен, или когда он полностью построен. Первая ситуация требует мгновенной реакции. Например, тег stream означает начало потока информации, и все остальные теги потока будут являться его содержимым. Подробнее с тегами XML-потока можно ознакомиться в пособии [2]. Вторая ситуация имеет место, т. к. XML-тег может приходиться «порциями».

Для демонстрации работоспособности предложенной модели были созданы две программы на языке Лисп (600 и 550 строк соответственно):

1. Из набора валидных XML-документов создается веб-страница;
2. Jabber-бот, который проходит аутентификацию, авторизовывает пользователей и отвечает «эхом» на их сообщения.

Рассмотрим, в качестве примера, функцию на языке Лисп, которая будет вычислять глубину готового XML-дерева.

```
(defun find_depth (xml maxd)
  (cond
    ((null xml) (- 1 maxd))
    ((isstring xml) maxd)
    ((isxmltag xml)(find_depth (getcontents xml) (+ 1 maxd)))
    (t (maxfunc (find_depth (car xml) maxd)
                (find_depth (cdr xml) maxd))))
  )
)
```

В данном примере можно увидеть некоторые из созданных проблемно-ориентированных функций: isxmltag — определяет является ли объект XML-тегом; isstring — определяет является ли объект строкой; getcontents — возвращает список содержимого XML-тега.

Подводя итог, можно сказать, что библиотека Intelib, проблемно-ориентированные функции и язык Лисп в совокупности дают средство для обработки XML-деревьев.

### Литература

1. Хювёнен Э., Сеппянен Й. *Мир Лиспа. В 2-х т. Т. 1: Введение в язык Лисп и функциональное программирование*. М.: Мир, 1990.
2. P. Saint-Andre, Ed. *Request for Comments: 3920. Extensible Messaging and Presence Protocol (XMPP): Core*. October 2004

## Статистические методы определения эмоциональной направленности текста

*Астабацян Каранет Араевич*  
*Кафедра алгоритмических языков*  
*e-mail: karo@webkaro.ru*

*Научный руководитель: к.ф.-м.н., доц. Грацианова Татьяна Юрьевна*

Работа посвящена одному из аспектов анализа текстов — определению тональности. Тональностью текста принято называть некоторую совокупную оценку эмоциональной составляющей. Данный вид автоматизированного анализа текста становится все более и более актуальным в самых разных сферах деятельности человека, включая неявное психологическое тестирование источника текстов, сбор и оценку мнений о каком-либо продукте, составление эмоционально окрашенных текстов (возможно, с заранее заданными параметрами), оценку уже составленных текстов и многие другие.

Целью данной работы является реализация программной системы, анализирующей тональность текста, с последующей оценкой точности ее работы. Выдвигается гипотеза о том,

что по некоторой достаточно небольшой доле всех слов в тексте можно вполне обоснованно и точно сделать предположение об эмоциональной окрашенности текста.

Достаточно подходящим для решения подобного рода задач механизмом является один из алгоритмов машинного обучения - метод «Опорных векторов»[1]. В силу специфики выбранного метода работа в рамках данной курсовой может быть разделена на несколько этапов.

В первую очередь, это подбор обучающей и тестовой выборки, т.е. множеств текстов, для которых исследуемые характеристики уже известны. Из них извлекается некоторая статистическая информация, которая будет использована при анализе. Выбранный алгоритм никак не зависит от тематики текстов и может быть применен практически в любых областях. В качестве упомянутых выборок используются отзывы об отелях, опубликованные на одном из сайтов в интернете[2]. Такой выбор источника отзывов был сделан в связи с тем, что помимо самого отзыва посетители этого сайта ставят оценку от нуля до пяти баллов, которая для нас является основным показателем эмоциональной направленности текста.

Для получения отзывов реализуется клиент-серверная программа на языке PHP[3], выбирающая отзывы, подходящие по заданным параметрам. Таким образом, получены необходимые корпуса текстов с положительными и отрицательными отзывами.

Далее разрабатываются и реализуются алгоритмы представления каждого отзыва в виде числового вектора, который отражает наличие или отсутствие тех или иных характеристик текста, на основе которых система сможет судить о тональности. Основной характеристикой является отношение количества эмоционально окрашенных слов в тексте к общему количеству слов. Слова, считающиеся эмоционально окрашенными, выделяются программой на этапе предварительной статистической обработки обучающей выборки.

Следующим этапом работы является обучение системы и подбор оптимальных параметров этого процесса. Основным критерием оценки работы программы является величина, называемая точностью, которая рассчитывается как отношение количества правильно классифицированных отзывов к общему объему тестовой выборки. В результате тестирования программы удалось достичь точности 84%.

Таким образом, основные результаты работы таковы:

1. Реализован классификатор отзывов.
2. Написана программа, собирающая с сайтов отзывы.
3. Был собран корпус текстов отзывов об отелях(более 70 тысяч отзывов).
4. Разработаны и реализованы алгоритмы составления частотных словарей и словарей эмоционально насыщенной лексики.
5. Была проверена гипотеза о том, что достаточно небольшая часть текста несет в себе эмоциональный окрас.

### Литература

1. Ryszard S. Michalski, Jaime G. Carbonell, Tom M. Mitchell. *Machine Learning: An Artificial Intelligence Approach*, Tioga Publishing Company. 1983.
2. Сайт «TopHotels — информационный рейтинг отелей». [HTML] (<http://www.tophotels.ru>).
3. Эд Леки-Томпсон, Алек Коув, Стивен Новицки, Хью Айде-Гудман. *PHP 5 для профессионалов - Professional PHP 5*. М.: «Диалектика», 2006. С. 608.

## Применение метода опорных векторов для анализа речевых команд в приложении-помощнике для мобильных устройств

*Иванов Николай Игоревич  
Кафедра алгоритмических языков  
e-mail: nickolas.ivanov@gmail.com*

*Научный руководитель: к.ф.-м.н., доц. Бордаченкова Елена Анатольевна*

За последние несколько лет технологии распознавания речи продвинулись далеко вперед: уже сейчас доступны онлайн-сервисы, способные распознать произнесенную фразу за доли секунды, и при этом точность распознавания оказывается довольно высокой. Этот факт делает актуальной задачу создания особых приложений для мобильных устройств — интеллектуальных помощников [1]. Основной задачей таких приложений является помощь человеку в использовании различных функций мобильных устройств. К таким функциям можно отнести отправку sms, поиск информации в интернете, добавление напоминаний о предстоящих событиях. Отличительной особенностью интеллектуальных помощников является выполнение голосовых команд пользователя, которые во многих ситуациях являются самым удобным и быстрым способом управления мобильным устройством.

Чтобы найти новые подходы к анализу пользовательских команд на русском языке, было решено создать собственный русскоязычный интеллектуальный помощник. К настоящему моменту нами разработан прототип такого приложения. В нём поддерживается работа с семью различными типами голосовых команд: телефонный вызов, отправка смс и электронных писем, поиск в интернете, просмотр web-страниц в браузере, установка будильника и проверка баланса. Сценарий работы с приложением следующий: пользователь нажимает на кнопку для запуска процесса распознавания речи и произносит команду. Записанная фраза переводится в текстовое представление сервисом распознавания речи от Google, после чего приложение определяет тип команды и ее параметры и запускает соответствующую функцию на телефоне.

Настоящая курсовая работа посвящена реализации модуля приложения, отвечающего за определение типа и параметров пользовательской команды. Для решения этой задачи нами использовался метод опорных векторов (SVM) [2,3]. Его выбор был обусловлен следующими соображениями. Метод опорных векторов позволяет анализировать произвольные пользовательские команды, без наложения каких-либо синтаксических и лексических ограничений. Кроме того, при обучении классификатора на большом количестве размеченных данных начинают учитываться скрытые закономерности этих данных, чего сложно добиться без использования методов машинного обучения.

Анализ пользовательской команды в приложении осуществляется в два этапа. На первом этапе приложение-помощник применяет обученную модель SVM для определения типа команды (телефонный вызов, отправка смс). На втором этапе в тексте пользовательской команды выделяются ее параметры (адресат, текст сообщения), при этом для каждого типа пользовательской команды используется собственная модель SVM, что позволяет повысить точность определения параметров.

Для обучения моделей SVM и оценки качества классификации была собрана и размечена коллекция пользовательских команд на русском языке в количестве 400 экземпляров. Настройка и проверка работы моделей SVM осуществлялась при различных соотношениях объёмов обучающей и тестовой выборок. Эксперименты показали, что даже для относительно небольшой обучающей выборки (обучающая выборка — 100, тестовая выборка — 300 предположений) точность SVM в среднем составляет 94%. Это говорит о том, что для достижения приемлемого качества классификации с помощью предложенного метода достаточно небольшого числа размеченных примеров пользовательских команд: при добавлении в приложение поддержки нового типа команды, существующую коллекцию размеченных пользовательских команд необходимо будет пополнить примерно 15-ю размеченными командами нового типа, что не потребует больших усилий со стороны разработчика.

В заключительной главе курсовой работы производится анализ ошибок классификации и предлагаются способы улучшения настройки моделей SVM для увеличения точности класси-

фикации.

### Литература

1. *Intelligent personal assistant, Wikipedia* [Электронный ресурс] – URL: [http://en.wikipedia.org/wiki/Intelligent\\_personal\\_assistant](http://en.wikipedia.org/wiki/Intelligent_personal_assistant)
2. *Автоматическая обработка текстов на естественном языке и компьютерная лингвистика : учеб. пособие* / Большакова Е. И., Клышинский Э. С., Ландэ Д. В. и др. – М.: МИЭМ, 2011. – с.181–184.
3. Chih-Wei Hsu, Chih-Chung Chang, Chih-Jen Lin *A Practical Guide to Support Vector Classification*, 2010 <http://www.csie.ntu.edu.tw/~rcjlin/papers/guide/guide.pdf>

## Разработка и реализация облачного планировщика, учитывающего топологию коммуникационной среды при высокопроизводительных вычислениях

*Работа удостоена диплома II степени  
Дудина Ирина Александровна  
Кафедра системного программирования  
e-mail: eupharia@gmail.com*

*Научный руководитель: к.ф.-м.н., проф. Гайсарян Сергей Суменович*

Ключевым препятствием для использования облачных систем в области высокопроизводительных вычислений является падение производительности приложений в облачной среде [1]. Одним из факторов, ограничивающих производительность приложений не только в облачных системах, но и на суперкомпьютерах, является пропускная способность сети, обеспечивающей взаимодействие между процессами в рамках одной задачи. В то время как собственно вычисления дешевеют, узким местом при масштабировании параллельных приложений является взаимодействие между процессами. Учитывая особенности высокопроизводительных вычислений, даже незначительное снижение производительности может привести к сильному ухудшению при масштабировании числа узлов [2].

В настоящее время существует несколько облачных платформ, как закрытых, так и с открытым исходным кодом. К закрытым относятся: Amazon EC2, Microsoft Azure, Google Compute Engine. Среди открытых платформ наиболее известны системы OpenNebula, OpenStack, Nimbus, Eucalyptus. Одной из наиболее активно развивающихся является OpenStack – комплекс проектов свободного программного обеспечения для создания вычислительных облаков и облачных хранилищ (распространяется под лицензией Apache License 2.0).

Целью данной работы являлась разработка алгоритма планировщика OpenStack для сокращения падения производительности, вызванного задержками сети, для конкретного варианта коммуникационной среды — сети Infiniband с топологией «толстое дерево».

Задача планирования распределенных (в т.ч. параллельных) программ в рамках вычислительного кластера в целом аналогична задаче распределения виртуальных машин по узлам кластера, при условии, что в виртуальных машинах выполняется какая-либо распределенная программа. Для того чтобы при планировании учесть взаимодействие виртуальных машин, планировщик должен размещать виртуальные машины, решающие конкретную задачу, ориентируясь на положение других виртуальных машин этой задачи [3]. Чтобы логически объединить виртуальные машины одной задачи, была введена концепция групп виртуальных машин в OpenStack, аналогичная набору процессов одного приложения. Таким образом, цель планировщика — разместить виртуальные машины наиболее «компактно» — может быть поставлена в рамках одной группы виртуальных машин.

Эффективность конкретного варианта размещения процессов данного приложения на узлах сети зависит от двух параметров: топологии сети и характера взаимодействия процессов

задачи между собой. В данной работе рассматривается подход к планированию, основанный на оценке эффективности размещения с помощью Hop-Byte метрики [2, 4]. Для конкретного варианта коммуникационной среды (сети Infiniband с топологией «толстое дерево» производится подсчет Hop-Byte метрики в предположении, что взаимодействие между виртуальными машинами одинаково.

На основе полученных результатов разработан алгоритм планирования, выполнена его реализация в рамках общего подхода к планированию в OpenStack с использованием концепции групп виртуальных машин. Таким образом, в OpenStack была добавлена возможность планирования виртуальных машин с учетом требований высокопроизводительных вычислений. Результаты проведенных исследований говорят о том, что разработанный в рамках планировщика OpenStack алгоритм планирования позволяет повысить производительность некоторых приложений. При этом падений производительности, связанных с использованием алгоритма, ни на каких тестах выявлено не было.

#### Литература

1. Кудрявцев А. О., Кошелев В. К., Аветисян А. И. *Перспективы виртуализации высокопроизводительных систем архитектуры x86* Труды Института системного программирования РАН, т. 22, 2012.
2. Bhatele A. *Automatic Topology Aware Mapping For Supercomputers* University of Illinois at Urbana-Champaign, 2010.
3. Gupta A., Milojevic D., Kale L. V. *Optimizing VM Placement for HPC in the Cloud* Workshop on Cloud Services, Federation, and the 8th Open Cirrus Summit, San Jose, 2012.
4. Sudheer C. D., Srinivasan A. *Optimization of the Hop-Byte Metric for Effective Topology Aware Mapping* 19th International Conference on High Performance Computing, 2012