**Московский институт электроники и математики
национального исследовательского университета
«Высшая школа экономики»**

Институт прикладной математики им. М.В. Келдыша РАН

Московский государственный технический университет им. Н.Э. Баумана

Новые информационные технологии в автоматизированных системах

Материалы шестнадцатого научно-практического семинара

Москва 2013

УДК 621.38
ББК 32.81
Н - 76

Н - 76 Новые информационные технологии в автоматизированных системах: материалы шестнадцатого научно-практического семинара. – Моск. Ин-т электроники и математики национального исследовательского университета «Высшая школа экономики». М., 2013, 332 с.
ISBN 978-5-94506-335-8
ISSN 2227-0973

Содержит материалы, представленные к рассмотрению на научно-практический семинар “Новые информационные технологии в автоматизированных системах”.

Представляет интерес для научных сотрудников, преподавателей, аспирантов и студентов, работающих по указанным научным направлениям.

ISBN 978-5-94506-335-8

УДК 621.38
ББК 32.81

© Институт прикладной математики им. М.В. Келдыша РАН, 2013

«Компьютерная графика»

Программная система для анализа и оптимизации вычислительных свойств гибридных разностных схем Burgers2

А.Е. Бондарев*, А.В. Бондаренко**, В.А. Галактионов*, Т.Н.Михайлова*, И.Г. Рыжова*

**Институт прикладной математики им. М.В. Келдыша РАН, Россия, Москва*

***ФГУП ГосНИИАС, Россия, Москва*

e-mail: bond@keldysh.ru; vlgal@gin.keldysh.ru

1. Введение

Данная работа посвящена описанию разработанного инструментального сервисного программного средства Burgers2 (ИСПС Burgers2), предназначенного для проведения оптимизационной настройки и отладки гибридных разностных схем. Данное программное средство реализовано на основе сочетания решения оптимизационной задачи и визуального представления результатов. Приведены примеры тестовых расчетов с иллюстрациями.

Инструментальное сервисное программное средство Burgers2 основано на численном решении уравнения Бюргерса и использовании его характерных свойств. Уравнение Бюргерса является модельным уравнением, описывающим распространение возмущений в диссипативной среде. Его принято рассматривать как одномерный аналог уравнений Навье-Стокса. Уравнение Бюргерса применяется в различных областях от вычислительной математики и вычислительной физики до экономических и социальных задач, то есть всюду, где необходимо моделировать распространение возмущений в среде с диссипацией.

Характерной особенностью уравнения Бюргерса является наличие точного решения, выражаемого аналитически. Это делает его идеальным объектом для настройки и отладки вычислительных инструментов – численных методов, разностных схем. Подход настройки с помощью уравнения Бюргерса вычислительных свойств разностных схем был предложен в работах [1-3], где рассматривалось влияние счетной вязкости разностных схем на расчет области высоких градиентов в нестационарных задачах. Идеология подхода, использованного в [1-3], реализована вычислительно и программно в данной работе.

2. Уравнение Бюргерса и гибридные разностные схемы

Рассмотрим краевую задачу, основанную на применении линейного уравнения Бюргерса, представленного в следующем виде:

$$\frac{\partial f}{\partial t}+\frac{\partial f}{\partial x}=\nu\frac{\partial^2 f}{\partial x^2}+f_1(x,t) \qquad (1)$$

Начальные и граничные условия представляются в виде:

$$f(x,t_0)=y(x,t_0) \qquad (2)$$

$$f(0,t)=y(0,t),\ f(1,t)=y(1,t), \qquad (3)$$

где $y(x,t)$ - точное решение, описывающее функцию типа бегущей волны.

Точное решение $y(x,t)$ имеет вид

$$y(x,t) = y(\xi) = e^{-d(1-2\xi)^2}, \text{ где } \xi = \frac{t-x}{t^*} \qquad (4)$$

Величины t^* и d характеризуют ширину и крутизну волны и являются параметрами задачи. Внешний вид точного решения (4) (бегущей волны) представлен на рисунке 1. Решение представлено на момент времени $t = 0.6$, что соответствует положению волны на середине отрезка $x \in [0,1]$.

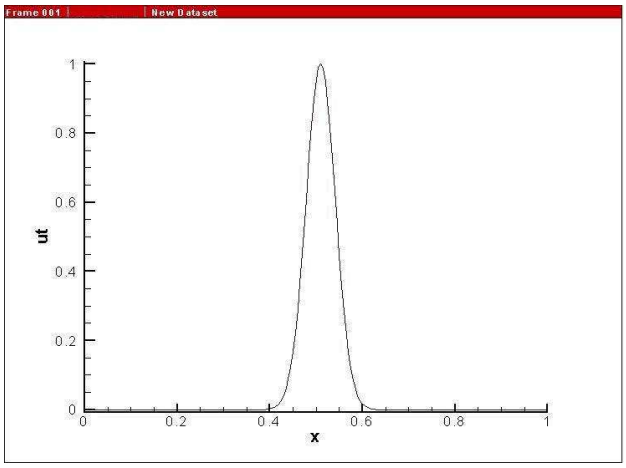


Рис. 1. Точное решение уравнения Бюргерса на момент времени $t = 0.6$

Функция $f_1(x,t)$ в правой части уравнения (1) является выражением следующего вида:

$$f_1(x,t) = -\frac{\nu}{t^*}[16d^2(1-2\xi)^2 - 8d]e^{-d(1-2\xi)^2} \qquad (5)$$

Наличие точного решения позволяет использовать уравнение Бюргерса для построения на его основе программного средства для настройки и отладки вычислительных свойств разностных схем, подобно [1-3].

Для этих целей вычисляется функция $K(x,t)$, представляющая собой модуль процента абсолютной ошибки:

$$K(x,t) = \left|[f(x,t) - y(x,t)] * 100\%\right|$$

где $f(x,t)$ - результат расчета, $y(x,t)$ - точное решение.

Далее ищется $K_{\max} = \max K(x,t)$ - максимальное значение модуля процента абсолютной ошибки во всем диапазоне изменения аргументов функции. Анализ поведения функции $K(x,t)$ и значения $K_{\max}$ позволяет судить о вычислительных свойствах рассматриваемой конкретной конечно-разностной схемы, подобно [1-3].

Рассматриваемые в применяемом подходе разностные схемы относятся к классу гибридных разностных схем. Гибридные разностные схемы – это схемы, соединяющие вычислительные свойства различных разностных схем. Подробное описание и классификация различных типов гибридных разностных схем приведены в обзорной работе [4].

Согласно [4], в простейшем случае гибридную схему можно записать как комбинацию:

$$S * G_1 + (1-S) * G_2 \qquad (6)$$

где S – коэффициент гибридности, G_1 и G_2 – разностные схемы, обладающие различными интересующими исследователя свойствами. Например, G_1 – схема первого порядка точности, а G_2 – второго порядка и т.п. Большинство применяемых для решения практических задач разностных схем являются гибридными. Согласно [4], к гибридным схемам относятся такие широко известные алгоритмы, как FCT (flux corrected transport), различные типы TVD (total variation diminishing) разностных схем, схемы типа ENO (essentially non-oscillatory) и WENO (weighted essentially non-oscillatory) и многие другие. Использование гибридных схем позволяет исследователю объединять различные разностные схемы в комбинацию, используя наилучшие свойства каждой из них. В то же время, осуществляя подобное объединение, необходимо иметь достаточно четкое представление о свойствах и ограничениях коэффициентов гибридности (весовых коэффициентов) для того, чтобы используемые свойства разностных схем соответствовали физической и математической моделям рассматриваемой задачи.

3. Постановка и решение оптимизационной задачи

Рассмотрим применение подхода [1-3] к конкретному варианту построения гибридной разностной схемы. В качестве гибридной разностной схемы использовалась гибридная разностная схема [3], где в качестве G_1 в выражении (6) используется схема Крэнка-Никольсона, а в качестве G_2 - схема Лакса.

Рассмотрим подробно построение разностной схемы [3] в применении к уравнению Бюргерса, записанному в общей форме:

$$\frac{\partial f}{\partial t} + c\frac{\partial f}{\partial x} = \nu\frac{\partial^2 f}{\partial x^2} \qquad (7)$$

Введем обозначения конечно-разностных операторов для аппроксимации пространственных переменных как

$$\delta_1 f_i^n = f_{i+1}^n - f_{i-1}^n \ , \quad \delta_2 f_i^n = f_{i+1}^n - 2f_i^n + f_{i-1}^n \qquad (8)$$

С помощью операторов (8) конечно-разностная схема (WW-схема), предложенная в [3], может быть записана для уравнения (7) с помощью весовых параметров S_1 и S_2 следующим образом:

$$f_i^{n+1} = S_1 f_i^n + \frac{1}{2}(1-S_1)(f_{i+1}^n + f_{i-1}^n) - \frac{c\Delta t}{2\Delta x}\left[S_2\delta_1 f_i^{n+1} + (1-S_2)\delta_1 f_i^n\right] +$$
$$+\frac{\nu\Delta t}{(\Delta x)^2}\left[S_2\delta_2 f_i^{n+1} + (1-S_2)\delta_2 f_i^n\right], \qquad (9)$$

где $\Delta t, \Delta x$ - шаги по времени и пространственной переменной.

Нетрудно заметить, что при выборе $S_1 \in]0;1[$, $S_2 = 0.5$ разностная схема (9) является линейной комбинацией схемы Крэнка-Никольсона, обладающей вторым порядком точности по времени и пространству, и схемы Лакса, обладающей существенной искусственной вязкостью. Представив весовой параметр S_1 в виде $S_1 = 1 - S_k\Delta t$, можно записать добавочный член с искусственной вязкостью как

$$S_k\frac{(\Delta x)^2}{2}\frac{\partial^2 f}{\partial x^2} \ . \qquad (10)$$

Таким образом, WW –схема (9) представляет собой неявную безусловно устойчивую гибридную схему второго порядка по времени и пространству, обладающую искусственной вязкостью, позволяющей устранять нефизические осцилляции вблизи разрывов. Управление искусственной вязкостью осуществляется путем выбора весов. Несомненным достоинством данной разностной схемы является то обстоятельство, что искусственной вязкостью можно управлять, непосредственно задавая соответствующее значение весового коэффициента S_k.

Данная разностная схема успешно использовалась для решения широкого круга модельных и практических задач вычислительной аэрогазодинамики. Основное достоинство схемы (9) – возможность объединения свойств схемы Крэнка-Никольсона и схемы Лакса.

Схема Крэнка-Никольсона обладает вторым порядком аппроксимации по времени и пространству, обеспечивая тем самым повышенную точность, но, в то же время, порождает нежелательные осцилляции в области высоких градиентов, уничтожая решение. Схема Лакса имеет первый порядок аппроксимации по времени, «размазывая» решение в области высоких градиентов, но обладает свойством подавлять нежелательные осцилляции.

Целью данного исследования является нахождение таких значений весовых параметров, которые обеспечивали бы достижение повышенной точности (минимальная ошибка), но не допускали бы возникновения осцилляций. Таким образом, уже в постановке задачи заложен конфликт целевых функций.

Для реализации этой цели нужно решать оптимизационную задачу и, следовательно, провести формальную постановку многокритериальной многопараметрической задачи оптимизации, согласно [5].

Оптимизационная задача в данном случае формулируется как поиск следующих величин:

$$\min K_{\max}(\nu, S_k), \text{для всех } \nu \in [\nu_1, \nu_2], S_k \in [0,1] \qquad (11)$$

$$\min S_k, \text{ так чтобы выполнялось } N \le N_k.$$

Здесь N - количество знакоперемен второй производной $\frac{\partial^2 f}{\partial x^2}$. Этот параметр определяет возникновение нежелательных осцилляций. При их возникновении он начинает неконтролируемо расти. Используя это свойство, в процессе решения данный параметр постоянно сравнивается с некоторым эталоном гладкости решения, определяемым пользователем программы с помощью задания значения параметра N_k.

Данная оптимизационная задача решается с помощью применения простого в реализации сеточного метода [5]. Суть этого простого и эффективного метода заключается в сеточном разбиении определяющих параметров оптимизационной задачи (аргументов целевой функции) и вычислении значений целевой функции в каждой точке полученной сетки. Наименьшее из этих значений приближенно принимается за минимум функции во всей области. Ранее традиционно считалось, что для задач большой размерности данный метод непригоден из-за больших вычислительных затрат. Однако современные вычислительные средства позволяют применять его успешно и эффективно, а применение параллельных вычислений позволяет ускорять процессы расчетов на порядки даже при самой безыскусной параллелизации. Это обстоятельство делает самые неприхотливые и простые методы

реально применимыми к практическим задачам. Более того, их простота и надежность создают им в этом случае значительные преимущества.

Для рассматриваемой задачи сеточное разбиение проводится в диапазонах $\nu \in [\nu_1, \nu_2], S_k \in [0,1]$. Для каждой точки сетки вычисляется максимальная ошибка $K_{\max}$. Каждая точка проверяется на возможное появление осцилляций с помощью блока-анализатора. Если осцилляции возникают, то эти точки отбрасываются, путем задания в них заведомо большой отрицательной величины. Как результат, мы получаем поверхность $K_{\max}(\nu, S_k)$. С помощью полученной поверхности определяется нужный нам минимум. Сама поверхность представляется визуально в целях контроля решения и наглядности.

4. Программная реализация и примеры расчетов

Построение программной реализации описанных в предыдущих разделах алгоритмов проводится по модульному принципу. Общая структура построения приведена в схеме, представленной на рисунке 2.

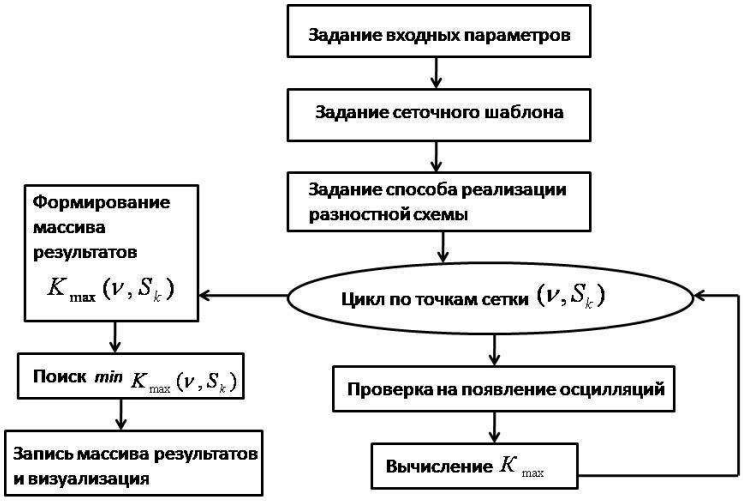


Рис.2. Схема взаимодействия программных модулей

Сначала в соответствующем модуле задаются входные параметры оптимизационной задачи и, таким образом, проводится сеточное разбиение определяющих параметров. В качестве входных параметров служат: nt – число точек по x, anuin – значение ν_1, anufin– значение ν_2, m – число точек разбиения отрезка $[\nu_1, \nu_2]$, ski – значение Sk_1, skf – значение Sk_2, n - число точек разбиения отрезка $[Sk_1, Sk_2]$, а также величины tz и d – соответствующие параметрам t^* и d, описывающим бегущую волну в точном решении.

Задание сеточного шаблона проводится в отдельном программном модуле. Это сделано для того, чтобы повысить гибкость программного средства, т.к. построение гибридной схемы может проводиться не только по простейшему принципу (6), но и с использованием более сложной линейной комбинации.

Реализация конечно-разностной схемы предусмотрена в соответствующем модуле, как для явных, так и для неявных схем. Неявные схемы реализуются с помощью скалярной прогонки.

Далее реализуется цикл по точкам сетки (ν, S_k). Для каждого набора значений (ν, S_k) проводится решение уравнения Бюргерса с помощью рассматриваемой разностной схемы. Решение проверяется на наличие нежелательных и неконтролируемых осцилляций. Проводится сравнение с точным решением и вычисляется значение $K_{\max}$. Таким образом формируется искомый массив результатов $K_{\max}(\nu, S_k)$ и проводится поиск положительного минимума данного массива.

Вывод данных осуществляется на экран и в текстовый файл (по умолчанию C:\res1.txt). Для каждого значения ν в строке приведено значение минимальной ошибки и соответствующее значение Sk. Отрицательное значение минимальной ошибки свидетельствует о наличии нежелательных осцилляций. Пример такой записи отражен ниже:

```
nu=  1.000000E-03  minerr=     2.761018      sk=  8.000001E-01
nu=  2.000000E-03  minerr=     1.280981      sk=  3.000000E-01
nu=  3.000000E-03  minerr=  4.440367E-01     sk=  1.000000E-01
nu=  4.000000E-03  minerr=    -1.000000      sk=  9.000001E-01
nu=  5.000000E-03  minerr=  5.271882E-01     sk=  1.000000E-01
nu=  6.000000E-03  minerr=  7.957876E-01     sk=  1.000000E-01
nu=  7.000000E-03  minerr=     1.002550      sk=  1.000000E-01
nu=  8.000000E-03  minerr=     1.165241      sk=  1.000000E-01
nu=  9.000001E-03  minerr=     1.295036      sk=  1.000000E-01
nu=  1.000000E-02  minerr=     1.401329      sk=  1.000000E-01
```

В программе предусмотрена запись данных в виде массива, предназначенного для визуального представления с помощью программы TecPlot. По умолчанию запись производится в C:\result.txt.

С помощью разработанного инструментального программного средства был проведен ряд расчетов для различных разностных схем и шаблонов. Ниже приведены два характерных примера результатов подобных расчетов.

В первом примере использовался сеточный шаблон (9) с выбором весовых коэффициентов $S_1 \in]0;1[$, $S_2 = 0.5$, обеспечивающий гибрид схем Лакса и Крэнка-Никольсона. На рисунке 3 приведен вид поверхности абсолютной ошибки. Область отрицательных данных отмечает места появления осцилляций.

С помощью полученных данных для любого значения ν легко подобрать значение весового коэффициента S_k, обеспечивающее минимальную ошибку, а, следовательно, максимальную точность рассматриваемой гибридной разностной схемы.

Второй пример рассматривает также шаблон (9) при выборе весовых коэффициентов $S_1 \in]0;1[$, $S_2 = 1$. Такой выбор весовых коэффициентов подразумевает отказ от использования второго порядка аппроксимации для пространственной переменной и переход к схеме первого порядка. Вид поверхности абсолютной ошибки приведен на рисунке 4.

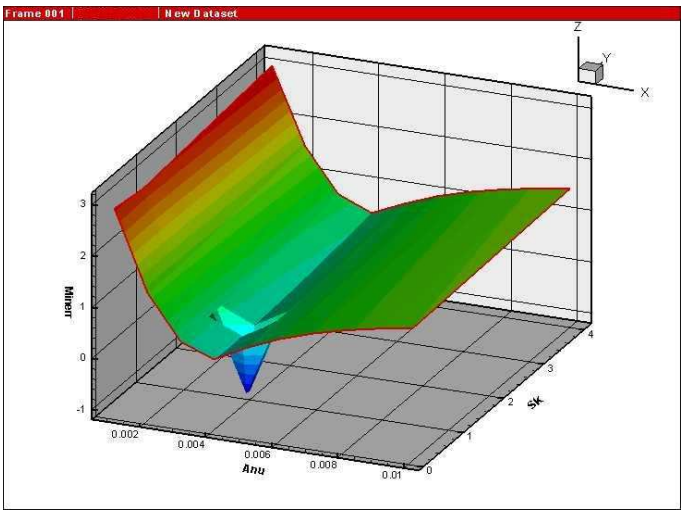


Рис. 3. Поверхность абсолютной ошибки при выборе весовых коэффициентов

$$S_1 \in]0;1[\,,\ S_2 = 0.5$$

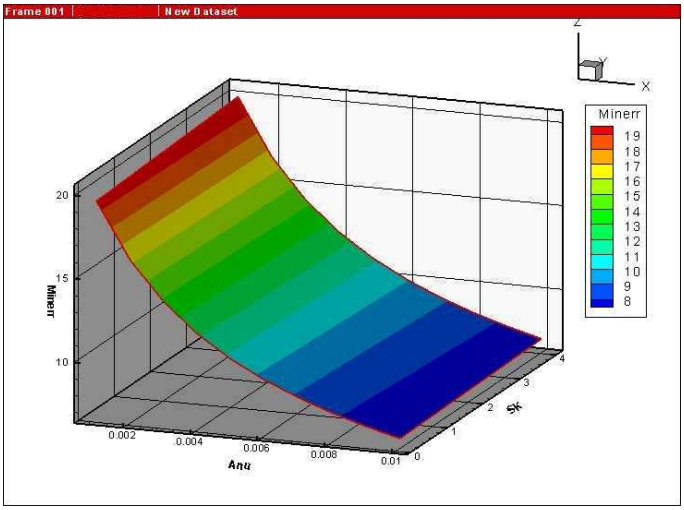


Рис. 4. Поверхность абсолютной ошибки при выборе весовых коэффициентов

$$S_1 \in]0;1[\,,\ S_2 = 1\,.$$

Данный рисунок дает точное представление об изменении поведения абсолютной ошибки при переходе к разностной схеме первого порядка аппроксимации по пространственной переменной. Меняется структура поверхности, значение ошибки резко увеличивается во всем поле определяющих параметров.

Программа Burgers2 реализована на языке Fortran90 на основе системы Microsoft Fortran Power Station 4.0. Данная версия является прототипной и разработана для реализации алгоритма. В дальнейшем данная версия нуждается в создании интерфейса, удобного для конечного пользователя и обеспечивающего:

- удобную организацию ввода-вывода данных;
- сопряжение с внешними устройствами;
- сопряжение с системами визуализации.

Также следует заметить, что решение рассматриваемой оптимизационной задачи реализовано в данной программе с помощью последовательного алгоритма. Однако здесь возможна организация надстройки для проведения параллельных вычислений для наборов определяющих параметров вместо цикла по точкам сетки. Подобный подход к решению оптимизационной задачи был предложен и опробован в работе [6].

Заключение

Разработано инструментальное сервисное программное средство Burgers2 (ИСПС Burgers2), предназначенное для проведения оптимизационной настройки и отладки гибридных разностных схем. Программное средство позволяет отлаживать вычислительные свойства гибридных разностных схем на модельном уравнении Бюргерса, описывающем распространение возмущений в диссипативной среде. Основой алгоритма является сочетание решения оптимизационной задачи и визуального представления результатов. Данное программное средство может служить инструментом отладки и настройки вычислительных свойств разностных схем, а также использоваться для образовательных целей. Разработанная версия программы является прототипом, в дальнейшем предполагается создание интерфейса для удобства конечного пользователя и разработка данного программного средства в варианте, предназначенном для проведения параллельных расчетов.

Литература

[1] Русаков С.В. Влияние счетной вязкости некоторых разностных схем на расчет области высоких градиентов в нестационарных задачах / Численные методы в аэродинамике:сб.трудов. М:.Изд.МГУ. 1980. вып.4. С.58-67.

[2] Русаков С.В. О некотором классе схем повышенного порядка точности по пространственным переменным / Численные методы в аэродинамике:сб.трудов. М:.Изд.МГУ. 1980. вып.5. С.3-13.

[3] Бондарев А.Е. Численное решение уравнения Бюргерса в области высоких градиентов // Препринты ИПМ им.М.В.Келдыша. 1990. № 12. 13 с.

[4] Куликовский А.Г., Погорелов Н.В., Семенов А.Ю. Математические вопросы численного решения гиперболических систем уравнений. М.: Физматлит, 2001, 608 с.

[5] Штойер Р. Многокритериальная оптимизация. Теория, вычисления и приложения. М.: Радио и связь. 1992. 504 с.

[6] Бондарев А.Е. Решение задачи оптимизационного анализа с помощью параллельных вычислений // Новые информационные технологии в автоматизированных системах: материалы пятнадцатого научно-практического семинара. М.:МГИЭМ. 2012. С. 89-93.

Интеграция модуля компьютерной графики в систему CATIA

Барладян Б.Х., Дерябин Н.Б., Шапиро Л.З.
Институт прикладной математики им. М.В.Келдыша РАН, Москва

Аннотация: Рассмотрены два подхода к интеграции системы синтеза реалистичных изображений и оптического моделирования в системы автоматизированного проектирования. Предложен подход, обеспечивающий частичную автоматизацию этой интеграции. Рассмотрены детали интеграции системы **Inspirer2** в систему автоматизированного проектирования CATIA.

Ключевые слова: *автоматизация проектирования, реалистичные изображения, оптическое моделирование.*

1. Введение

Системы оптического моделирования, основанные на физически аккуратном (корректном) моделировании распространения света, являются мощным инструментом для разработчиков и дизайнеров в самых различных областях человеческой деятельности и, прежде всего, в архитектурном проектировании, ландшафтном дизайне, автомобильной и авиационной промышленности. Они позволяет быстро и необходимой точностью получать как реалистичные изображения проектируемых объектов, так и численные значения различных параметров уже на этапе проектирования этих объектов. Примерами таких объектов могут служить различные здания и сооружения, жилые и офисные помещения, интерьеры автомобилей и самолетов. При этом контролируется эстетичность интерьеров и внешнего вида, рациональность, экономичность, соответствие стандартам систем освещения искусственным и естественным светом. Типичными приборами, где применяется оптическое моделирование, являются также системы подсветки экранов мониторов и различных мобильных устройств, разного рода управляющих кнопок. Примером системы оптического моделирования является **Inspirer2** [1, 2].

Методы синтеза реалистичных изображений, используемые в компьютерной графике, являются эффективным средством повышения производительности труда в таких областях человеческой деятельности как архитектура, ландшафтный дизайн, автомобильная и авиационная промышленность, конструирование различного рода оптических приборов.

Можно сформулировать следующие требования, предъявляемые к современным системам синтеза реалистичных изображений и оптического моделирования:

1) Физически аккуратные методы оптического моделирования.

Синтез реалистичных изображений, близких по восприятию к тому, что появится после реализации проекта, необходимых в архитектурном проектировании, градостроительстве, ландшафтном дизайне, автомобильной и авиационной промышленности возможен только в системе, основанной на физически аккуратном (корректном) моделировании распространения света. Эти методы также абсолютно необходимы при проектировании оптических устройств.

2) Учет сложных физических эффектов.

Адекватное моделирование объектов и материалов, изготавливаемых с использованием современных технологий, невозможно без учета сложных физических эффектов. К таким эффектам следует отнести рассеивание света на стохастических микронеровностях (шероховатостях) поверхностей и на регулярных

микрорельефах поверхностей (десятки и сотни миллионов микро-призм заданной формы, размера и распределения), объемное рассеивание света на микрочастицах внутри материала, поляризационное рассеивание света в тонких пленках и ряд других подобных явлений. Адекватное моделирование таких эффектов требует в свою очередь использования мульти спектральных методов. Моделирование в RGB пространстве не позволяет получить правильные цвета для восприятия человеком.

3) Задание правильных исходных данных.

Синтез реалистичных изображений требует задания корректных исходных данных – геометрии сцены, оптических характеристик поверхностей и материалов.

4) Интегрирование в системы автоматизированного проектирования.

Все современные компании в настоящее время на различных этапах производства и сопровождении в эксплуатации сложных технических изделий используют так называемые технологии управления жизненным циклом изделий (Product Lifecycle Management (**PLM**)). Эти системы обеспечивают управление всей информацией об изделии и связанных с ним процессах на протяжении всего его жизненного цикла, начиная с проектирования и производства до снятия с эксплуатации. Информация об объекте, содержащаяся в **PLM**-системе является цифровым макетом этого объекта. Геометрическая модель объекта в таких системах создается, как правило, с помощью специальных систем автоматизированного проектирования (**САПР**). Для трехмерных геометрических объектов стандартом де факто такой системы является **CATIA** [3]. По крайней мере, ее используют большинство авиационных, кораблестроительных и автомобильных компаний. **Autodesk 3DS MAX** [4] широко используется для презентаций различных архитектурных проектов. Синтез реалистичных изображений для объектов, разрабатываемых в этих системах, а также получение их различных оптических характеристик таких, как распределение освещенности и яркости на различных реальных или виртуальных поверхностях, является важной частью общего проектирования. Для обеспечения эффективного использования подсистемы оптического моделирования в общем процессе проектирования необходимо обеспечить максимально гладкое ее взаимодействие с системой автоматизированного проектирования, где создается геометрия объектов, и хранятся их атрибуты. Разработчик изделия в системе автоматизированного проектирования должен использовать оптическую подсистему просто как одну из подсистем, которая автоматически воспринимает геометрию, созданную в системе, и те атрибуты, которые должны быть учтены в моделировании. Соответственно оптическая подсистема должна автоматически учитывать все те изменения в геометрии или атрибутах, которые могут вноситься на различных этапах проектирования изделий. Без такой эффективной интеграции разработчики просто не будут пользоваться этой подсистемой за исключением критических случаев, когда без оптического моделирования нельзя обойтись в принципе.

Задача интегрирования систем физически корректного моделирования распространения света, используемых для построения реалистичных изображений, с современными **САПР** является трудной. Известны попытки ее решения, разработанные компаниями **Discreet** (**Lightscape**) [5], **Mental Images** [6] и **Chaos Group** (**VRay**) [7] для **3DS Max**. Для системы автоматизированного проектирования CATIA компанией OPTIS был выпущен встроенный модуль моделирования освещенности **SPEOS CAA V5 Based** [8]. Компания **Mental Images** интегрировала свою программу моделирования освещенности в **Autodesk 3DS MAX**, **Maya** и

CATIA(**PhotoStudio**). В настоящей работе рассматриваются два подхода к решению этой задачи, и предлагается способ автоматизации этого интегрирования. Предлагаемый подход позволяет обеспечить максимальное повторное использование не только уже существующих вычислительных модулей, но и даже частичное повторное использование графических пользовательских интерфейсов.

2. Подходы к интеграции с системами автоматизированного проектирования

В состав систем автоматизированного проектирования входит, как правило, набор инструментальных средств разработки программного обеспечения (**SDK** – software development kit). Этот инструментарий предназначен для расширения и дополнения функциональных возможностей данной системы. Так средства **3ds Max SDK** предназначены, в первую очередь, для создания новых возможностей (Plugins) по созданию и модификации геометрических объектов (процедурные объекты и модификаторы), материалов и текстур. Соответствующие средства **CATIA V5** также содержат мощные средства для точного моделирования трехмерных объектов на базе использования различных представлений криволинейных поверхностей и топологических операций. Эти средства и позволяют встраивать в системы автоматизированного проектирования синтез реалистичных изображений и оптическое моделирование как одну из специальных подсистем.

Можно рассматривать два подхода к интеграции с системами автоматизированного проектирования:

1. Максимально полное использование инструментальных средств разработки данной системы автоматизированного проектирования и перепрограммирование существующей системы оптического моделирования с использованием новых средств.

2. Максимально полное использование модулей уже существующей и отлаженной системы оптического моделирования и синтеза реалистичных изображений.

Оба подхода имеют очевидные преимущества и недостатки, и в практической реализации приходится выбирать некоторое компромиссное решение. Так, при интеграции нашей системы моделирования освещенности с **3ds Max**, было принято решение о максимальном использовании возможностей предоставляемых **3ds Max SDK**. В этом случае пользователь, уже привыкший к используемым моделям материалов, источников света, наложения текстур, пост процессинга, мог бы использовать их в новом плагине в том виде, как они описаны для встроенного визуализатора. Дополнительно пользователь может использовать расширенные модели материалов и источников света, обеспечиваемые нашим плагином. Эти модели необходимы для обеспечения физически аккуратного (корректного) моделирования распространения света, поскольку исходные модели не были рассчитаны на такое моделирование. Основным преимуществом нашего плагина над стандартным визуализатором является получение не прямого освещения. Реалистичность полученных изображений в этом случае получается существенно выше даже при использовании относительно искусственных моделей материалов (свойств поверхностей). Это объясняется тем, что моделирование освещенности базируется на физических законах распространения света. Неточность моделей материалов приводит только к некоторым ошибкам в величинах освещенности, но во многих случаях это не критично с точки зрения реалистичности полученных

изображений, в то время как пользователь избавлен от процедуры точного задания атрибутов материалов и источников света, их измерения и т.д.

При интегрировании с **CATIA** (Dassault Systems) была поставлена задача получения максимально точных результатов моделирования освещенности. Такое моделирование требует введения источников света, отсутствующих в **CATIA**, задания оптических свойств используемых материалов, расчета освещенностей различных объектов сцены и использования для анализа полученных результатов различных видов виртуальных измерительных объектов. В ряде случаев необходимо задавать оптические свойства не только поверхностей объектов, но и среды, в которой распространяется свет. К таким параметрам относятся, например, дисперсия (зависимость индекса рефракции от длины волны), объемное рассеяние и флюоресценция. При описании параметров объемного рассеяния необходимо учитывать что материал, из которого изготовлен объект, может быть многокомпонентным, например, состоять из прозрачного материала заполненного частичками, изготовленными из других материалов. Полное описание такого многокомпонентного материала требует достаточно сложного пользовательского интерфейса. В частности он требует описания каждого из составляющих его материалов, концентрации каждого сорта частиц и их параметров. Для работы с подсистемой оптического моделирования необходим ряд других, достаточно сложных пользовательских интерфейсов, например, анализа полученных результатов моделирования в виде спектральных изображений в плоскости камеры или других виртуальных измерительных объектов. Следует также учитывать, что внутренние требования к построению объектов в **Inspirer2** и **CATIA** существенно отличаются. Следовательно, для создания объектов, описывающих соответствующие модели в **CATIA,** потребуется их полное перепрограммирование и конверсия при передаче этих данных в среду моделирования **Inspirer2**.

Сложных интерфейсов требует также визуальное представление распространения света в оптической системе в виде трассируемых лучей. Двойная разработка и отладка таких сложных пользовательских интерфейсов и внутренних объектов сначала для базовой системы моделирования освещенности, а затем для интегрированной в **CATIA** требовала бы слишком много времени. Существенным фактором являлось также необходимость длительного использования достаточно дорогой среды программирования и специфической квалификации программистов для работы в этой среде. В силу всех этих причин было принято решение максимально использовать в интегрированной подсистеме уже разработанные и отлаженные модули базовой подсистемы. Нам удалось разработать такую технологию интеграции модулей базовой системы оптического моделирования в систему автоматизированного проектирования, которая позволяет использовать модули базовой системы практически без изменений. Реализация интеграции нашей системы оптического моделирования **Inspirer2** в систему **CATIA** описана в двух следующих разделах.

3. Основные блоки интегрированной системы

Интегрированная система **Inspirer CATIA** (**IC**) была образована путем интеграции в систему **CATIA** пост процессора обработки результатов **HDRIPostprocessor** входящего в систему **Inspirer2** и специального сервера **I2 Server** реализованного на ядре системы **Inspirer2** (**Inspirer2 SDK**).

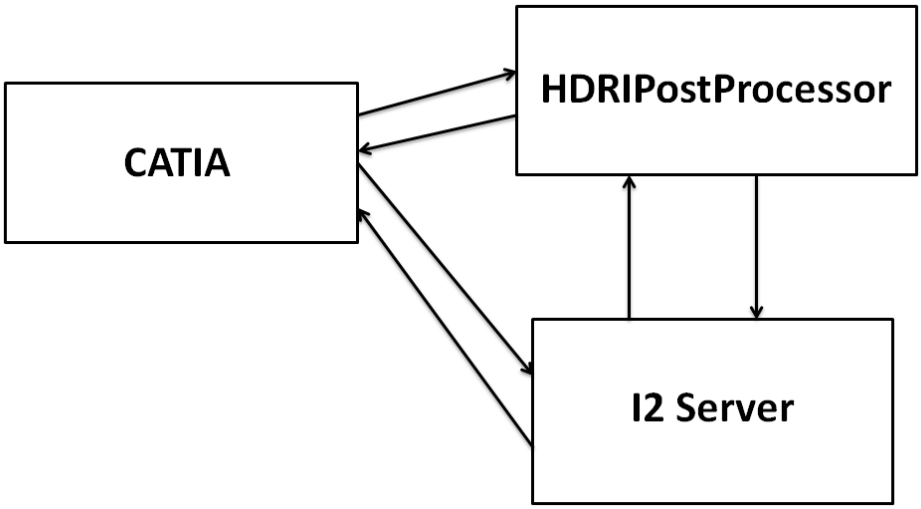


Рис.1 Основные компоненты системы Inspirer CATIA

CATIA – это базовая система автоматизированного проектирования. С точки зрения системы моделирования освещенности она обеспечивает создание и модификацию геометрии объектов сцены и хранение данных. Для целей моделирования освещенности она была расширена возможностями описания и редактирования оптических атрибутов поверхностей, источников света и виртуальных измерительных объектов. На базе Inspirer2 SDK была также обеспечена возможность расчета не прямой освещенности поверхностей, измерения оптических характеристик сцены с помощью виртуальных измерительных объектов и получения реалистических изображений с помощью трассировки лучей.

HDRIPostprocessor позволяет производить обработку полученных реалистичных изображений и анализ результатов измерений полученных на виртуальных измерительных объектах. В частности он позволяет:

- преобразовывать полученные изображения в физических величинах (изображения с высоким динамическим диапазоном) в изображения в RGB пространстве с ограниченным диапазоном, используя различные способы тонального отображения (операторы сжатия динамического диапазона яркостей) [9].
- отображать результаты моделирования с использованием тоновой заливки (искусственные цвета).
- Обрабатывать результаты моделирования, полученные в **RGB** и спектральных цветовых пространствах.
- Обрабатывать результаты моделирования, полученные на различных видах виртуальных сенсоров – угловое распределение яркости, распределение яркости/освещенности на прямоугольнике виртуального сенсора.

I2 Server позволяет интегрировать большинство функциональностей **Inspirer2 SDK** в **CATIA** в полуавтоматическом режиме. В настоящее время он обеспечивает реализацию пользовательского интерфейса для описания параметров среды распространения света, доступ к **Inspirer2** библиотекам описания сред, Монте Карло моделирование для визуального представления распространения света в оптической системе в виде трассируемых лучей, ряд вспомогательных функциональностей, необходимых для функционирования системы. Взаимодействие и синхронизация между блоками системы осуществляется с помощью сообщений и событий

операционной системы **Windows**. **I2 Server** может быть относительно легко расширен для поддержки новых функциональностей, реализованных в **Inspirer2** для поддержки их в Inspirer CATIA. Добавление, например, новой модели среды распространения света вообще происходит автоматически, поскольку и необходимый пользовательский интерфейс и само моделирование (Монте Карло трассировка) реализовано непосредственно в **I2 Server**. Если новая функциональность требует передачи дополнительных параметров между блоками, то это требует реализации лишь относительно небольших программных надстроек в соответствующих блоках. Детали реализации описаны в следующем разделе.

4. Синхронизация компонент в системе Inspirer CATIA

I2 Server реализован как обычное Windows приложение. Это приложение стартует одновременно с **CATIA. Его** главное окно скрыто. Его главная оконная процедура заменена для обработки сообщений, посылаемых **CATIA.** Специальные **Windows** события используются для синхронизации **CATIA** и **I2 Server** в процессе вычислений и перерисовки окна **CATIA.**

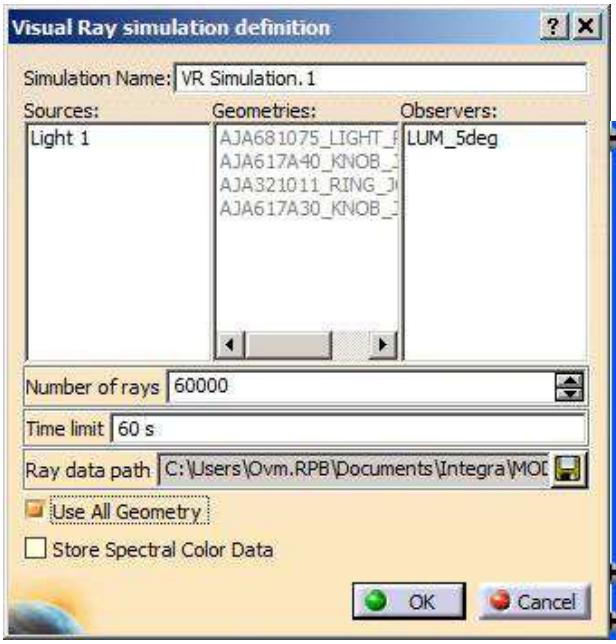
Рассмотрим синхронизацию компонент более детально на примере визуализации распространения света в оптической системе в виде трассируемых лучей. На первом шаге пользователь должен выбрать источники света, геометрические объекты и виртуальные сенсоры, которые будут участвовать в данном моделировании. Это, естественно, делается непосредственно в **CATIA** с помощью интерфейса, изображенного на рис. 2а. Пользователь выбирает соответствующие объекты непосредственно в окне визуализации или в дереве сцены. Имена выбранных объектов отображаются в соответствующих окнах. В этом же диалоге можно установить ограничение времени моделирования, количество трассируемых лучей и другие необходимые параметры. После завершения выбора параметров моделирования соответствующий элемент появляется в поддереве моделирований. Теперь моделирование может быть запущено с помощью выпадающего, по правой кнопке мыши, меню – рис 2б. Непосредственно моделирование выполняется в **I2 Server**. **CATIA** для этого сначала формирует описание сцены в формате IOF (бинарный формат системы **Inspirer2**), сохраняет его на диске и посылает **I2 Server** сначала сообщение `IDM_VR_SCENE_LOAD` (загрузи сцену) а за тем `IDM_VR_CALCULATE` (моделирование визуализации лучей). Дополнительные данные (путь на сохраненное описание сцены, ограничение по времени, количество лучей и т.д.) передается при этом с использованием общей памяти разделяемой процессами (shared memory).

Выполнение каждой команды в **I2 Server** синхронизируется с **CATIA** по общей схеме.

1. **CATIA** сбрасывает первое синхронизирующее событие (*event1*)
2. **CATIA** посылает сообщение и ждет взведения *event1*
3. **I2 Server** извлекает данные из общей памяти, помещает дескриптор окна в общую память.
4. **I2 Server** сбрасывает второе событие (*event2)* и взводит *event1*. Ждет взведения *event2*.
5. **CATIA** извлекает из общей памяти дескриптор окна, сбрасывает *event1* и взводит *event2*.
6. **I2 Server** дождавшись взведения *event2*, выполняет команду моделирования и взводит *event1*. Во время моделирования индикатор показывает степень выполнения

команды: отношение количества отмоделированных лучей к тому, которое заказано в процентах. Результаты моделирования **I2 Server** записывает на диск в специальном формате.

7. **CATIA,** ожидая взведения *event1*, каждые 100 миллисекунд выполняет перерисовку своего окна и делает окно **I2 Server** активным.



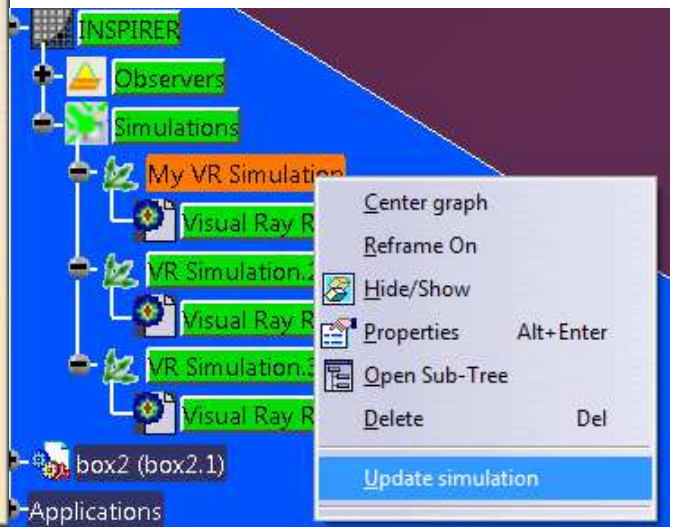


Рис. 2а. Выбор объектов *Рис. 2б. Запуск моделирования*

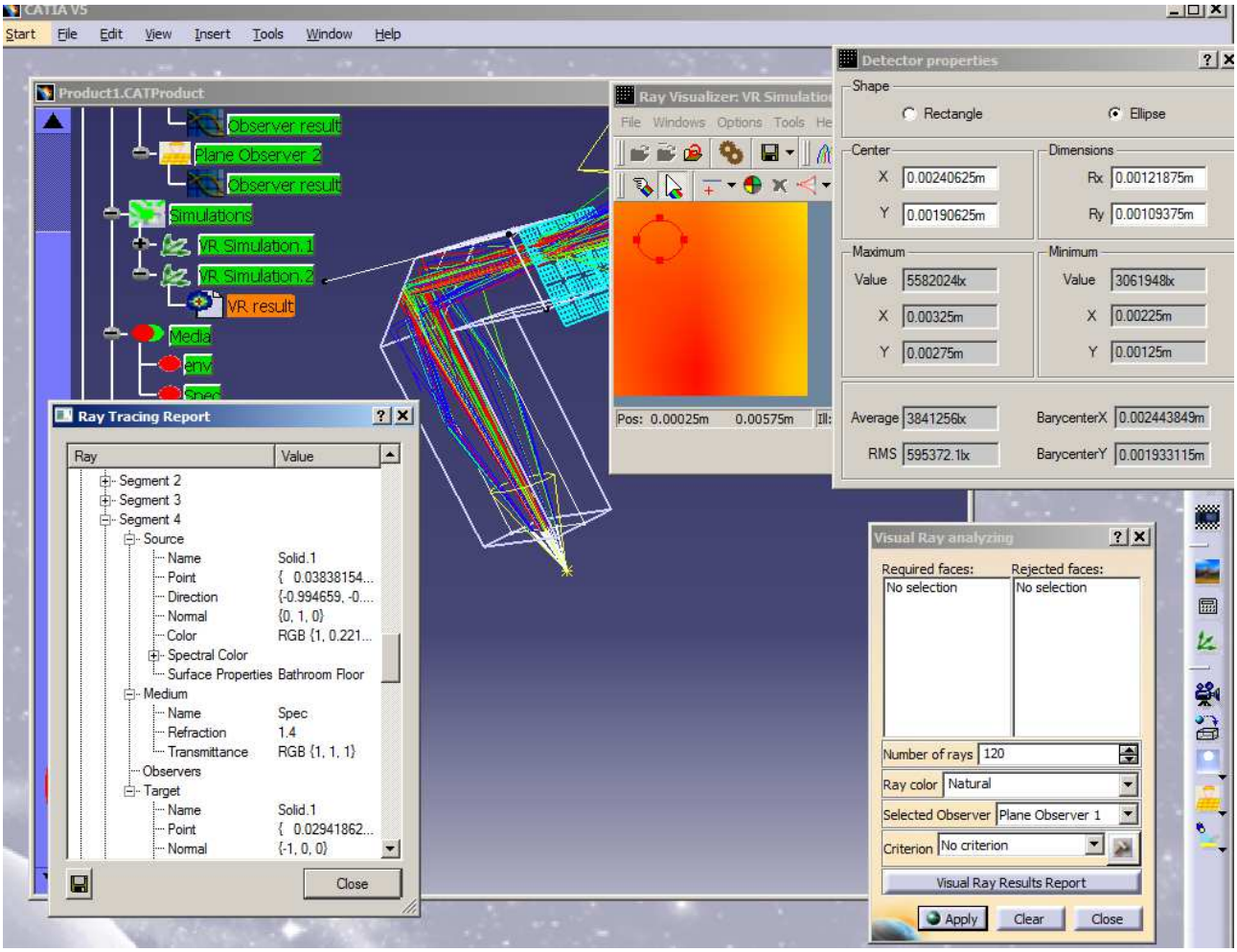


Рис. 3. ***CATIA*** *и дополнительные диалоги*

Диалог “Visual Ray analyzing” реализован непосредственно в **CATIA.** Пользователь устанавливает здесь различные критерии для визуализируемых лучей. В частности можно визуализировать только те лучи, которые пересекают (или не пересекают) заданный набор граней объектов. Дополнительный набор более сложных ограничений может быть установлен с помощью специального критерия. Создание и редактирование такого критерия реализовано в **I2 Server**. Вызов редактора (рис.4) обеспечивает кнопка . Если при моделировании использовался виртуальный сенсор, и он выбран в списке (Selected Observer), то одновременно с диалогом в **CATIA** на экране отображается окно **HDRIPostprocessor**. Последний кроме своих стандартных возможностей обеспечивает отбор визуализируемых лучей с помощью эллиптического или прямоугольного детектора (окно расположено в правом верхнем угле рис.3). Взаимодействие **CATIA** с **HDRIPostprocessor** и **I2 Server** происходит по общей схеме, описанной выше. После нажатия кнопки **Apply** сначала посылается соответствующая команды в **HDRIPostprocessor** (если он активен) и оттуда через общую память передается информация о параметрах детектора, затем посылается команда в **I2 Server**, которому передается вся необходимая информация о критериях, для отбора лучей полученных в результате моделирования. **I2 Server** производит отбор лучей согласно полученному критерию и возвращает отобранные лучи через общую память для визуализации. **CATIA** отображает эти лучи в своем окне поверх геометрии сцены.

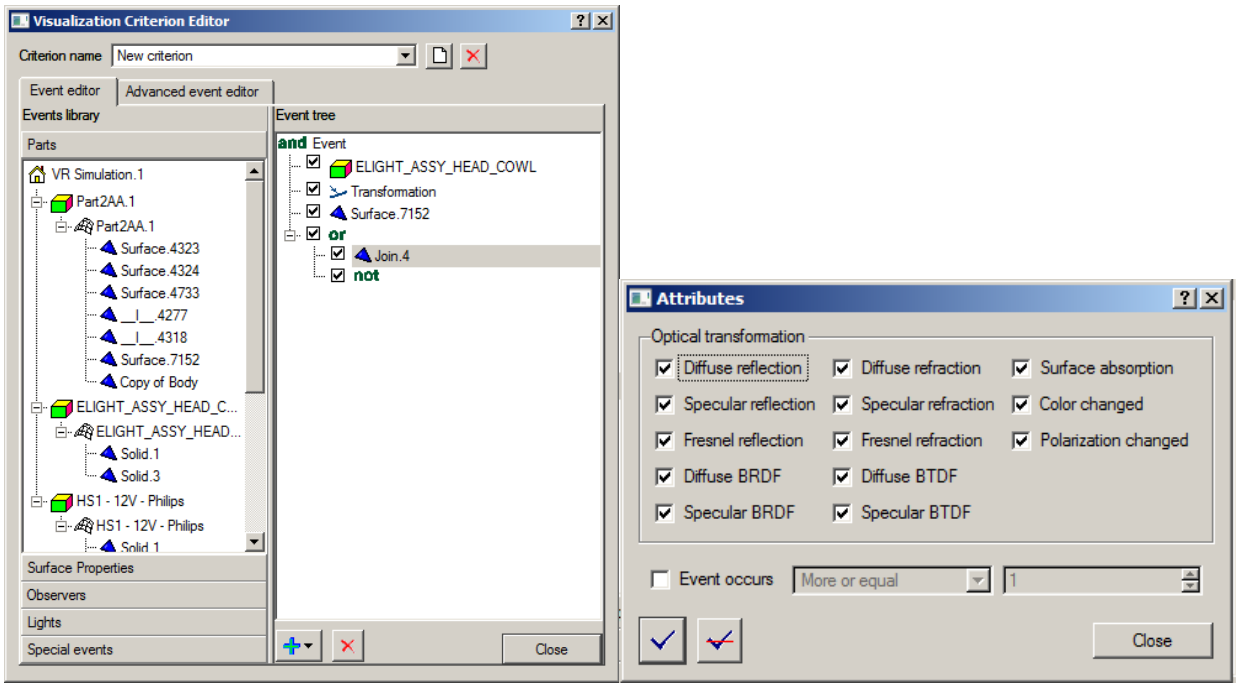


Рис. 4. Редактор критерия

Дополнительная функциональность позволяет детально проанализировать историю трассировки лучей, отобранных с помощью текущего критерия. Соответствующий диалог (нижний левый угол Рис.3) может быть вызван с помощью клавиши **Visual Ray Results Report**. Это окно позволяет проанализировать события, произошедшие вдоль траектории лучей, изменения цвета (в том числе и спектральных компонент), поверхности на которых происходили события, среды и т.д. Визуализация этого окна обеспечивается **I2 Server** по соответствующей команде из

CATIA, которая, в свою очередь, обеспечивает удержание этого окна поверх собственного. Это обеспечивается п.7. общей схемы синхронизации.

5. Заключение

Предложенный в работе подход к интеграции систем синтеза реалистичных изображений и оптического моделирования в систему автоматизированного проектирования был реализован при интеграции ряда функциональностей нашей системы оптического моделирования **Inspirer2** в систему **CATIA**. Подход позволяет частично автоматизировать процесс интеграции, поскольку модули обработки событий и синхронизации практически требуют минимальных расширений при добавлении новых функциональностей. В результате время интеграции новой функциональности, уже реализованной в **Inspirer2**, в **CATIA** сокращается многократно. Это особенно существенно, поскольку стоимость среды программирования в **CATIA** (стоимость лицензии) довольно высока и само программирование требует квалифицированных специалистов со значительным опытом работы в этой среде.

Список литературы

1. A. Ignatenko, B. Barladian, K. Dmitriev, S. Ershov, V. Galaktionov, I. Valiev, A. Voloboy. A Real-Time 3D Rendering System with BRDF Materials and Natural Lighting. The 14-th International Conference on Computer Graphics and Vision GraphiCon-2004, Conference Proceedings, Moscow, 2004, pp. 159-162.
2. Волобой А.Г., Галактионов В.А. Машинная графика в задачах автоматизированного проектирования // "Информационные технологии в проектировании и производстве", № 1, 2006, с. 64-73.
3. Dassault Systems, company website, http://www.3ds.com/home.
4. Autodesk 3DS Max, company website, http://usa.autodesk.com/3ds-max.
5. Lightscape software, lighting simulation software, http://www.digitalbroadcasting.com/product.mvc/Lightscape-software-0001.
6. Mental Images, company website, http://www.mentalimages.com.
7. Chaos Group, company website, http://www.chaosgroup.com.
8. П. Гольдовский, А. Кокова. Моделирование оптических явлений и свойств задаваемых изделий. САПР и графика, №8, 2004,с. 46-47.
9. Б.Х. Барладян, А.Г. Волобой, В.А. Галактионов, Э.А. Копылов. Эффективный оператор сжатия динамического диапазона яркостей, "Программирование", № 5, 2004, с. 35-42.

Задача позиционирования камеры заднего вида

Е.Д. Бирюков
Институт прикладной математики им. М.В. Келдыша РАН, Россия, Москва
e-mail: peb@gin.keldysh.ru

Аннотация. Камеры заднего вида в настоящее время широко используются на автомобилях. Основное назначение камеры заднего вида - обеспечение безопасности и комфорта при движении задним ходом. Изображение с камеры отображается на специальном экране в салоне автомобиля. При этом зачастую используются камеры с широким углом обзора.

Камеру необходимо расположить на корпусе кузова автомобиля таким образом, чтобы обеспечивался хороший обзор, при этом сама камера была бы по возможности защищена от внешних воздействий, в том числе от попадания грязи при движении. Целью данной работы является расчет такого взаимного расположения кузова автомобиля и камеры, при котором никакие элементы кузова не будут мешать обзору из камеры.

Постановка задачи

Камера заднего вида располагается на задней оконечности кузова автомобиля, как правило, в специальном углублении. При этом отдельные элементы кузова, дополнительные декоративные накладки и другие предметы могут загораживать обзор из камеры. Задачей является вычисление такой формы элементов задней части кузова автомобиля, при которой они не будут мешать обзору из камеры.

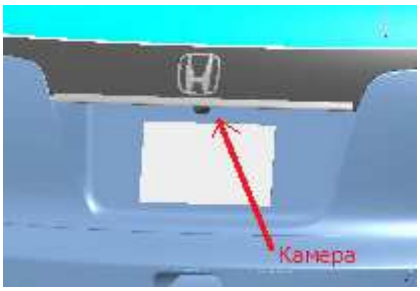


Рис. 1. Расположение камеры на задней части кузова автомобиля

В большинстве случаев изображение, проецирующееся камерой, подвергается кадрированию. В камерах заднего вида зачастую используются сферические линзы, проекция области обзора которых на плоскость представляет собой окружность. При построении кадра прямоугольной формы значительная часть исходного изображения при этом остается за кадром. Поэтому для нормальной работы камеры достаточно, чтобы никакие детали кузова не перекрывали бы область, находящуюся внутри прямоугольного кадра.

Определение области кадрирования на изображении с камеры

При указанных условиях становится возможным следующий способ решения поставленной задачи. На изображении, спроецированном с помощью камеры, строится двухмерный контур. Затем по данному контуру восстанавливается трехмерная фигура, проекцией которой на плоскость экрана является данный контур. После этого становится возможным выяснить, пересекается ли полученная фигура с какими либо элементами кузова автомобиля. Конечной целью является получение

такой формы всех элементов кузова, при которой они не будут пересекаться с построенной фигурой и будут целиком находиться за ее пределами.

Техническая реализация задачи выглядит следующим образом. Проводится рендеринг сцены из положения камеры с учетом ее угла обзора и функции нелинейных искажений. Использование такой функции позволяет задать камеру с углом обзора, большим чем 180 градусов. Такие камеры (так называемый «рыбый глаз») широко используются в качестве автомобильных камер заднего вида [1]. Далее на полученном двухмерном изображении пользователь задает контур требуемой формы и для двухмерных координат его точек (с некоторым шагом, определяемым требуемой точностью) определяется положение в пространстве с учетом той же функции нелинейных искажений. Затем по полученному трехмерному контуру строится искомая фигура.

Контур на плоскости проекции можно определить разными способами. Целесообразно использовать для этих целей какую-либо параметрическую кривую, так как для такой кривой отсутствуют ограничения на направление. Было решено использовать кубическую кривую Безье, которая довольно часто используется в компьютерной графике, в том числе, в системе автоматизированного проектирования CATIA [2]. Сплайн Безье полностью определяется пользователем в плоскости проекции камеры, а далее для всех его точек с некоторым шагом определяются координаты в пространстве. Получившаяся кривая уже не является сплайном Безье (строго говоря, она является ломаной, но при определенном расстоянии между точками для вычисления обеспечивается достаточная гладкость этой кривой).

Особеннности использования камеры со сверхшироким углом обзора

Для обычной камеры, угол обзора которой меньше, чем 180 градусов, достаточно построить контур на плоскости. Трехмерная фигура, проекцией которой на плоскость является данный контур, представляет собой прямой конус с основанием в форме данного контура. Однако в некоторых случаях угол обзора камеры заднего вида может быть больше, чем 180 градусов. Такая камера проецирует изображение на плоскость, но построить конус с основанием, однозначно определенным с помощью контура на плоскости проекции для данной камеры не представляется возможным, поскольку при этом возможна ситуация, когда контур в пространстве будет пересекать сам себя, что недопустимо. Поэтому в данном случае целесообразно использовать контур на сфере, центр которой совпадает с положением камеры. При этом принимается, что радиус сферы, на которой должны лежать все точки контура, равен 1. Фигура, по которой будет определяться попадание частей кузова в область обзора камеры будет строиться следующим образом: один конец образующей будет находиться неподвижно в центре сферы, другой конец будет двигаться вдоль контура на сфере. Такую фигуру можно назвать конусом, опирающимся на сферическое основание.

Для вычисления трехмерных координат какой-либо точки в пространстве по ее двухмерным координатам на плоскости проекции камеры используются следующие формулы:

$$dX = pt.X - center.X$$

$$dY = pt.Y - center.Y$$

где pt – координаты точки на плоскости проекции, center – координаты точки на плоскости, соответствующей осевой линии камеры.

$$r = \sqrt{dX^2 + dY^2}$$
$$\theta = Pix2Angle(r)$$

где Pix2Angle – табличная функция нелинейных искажений, заданная для данной камеры. Эта функция вводит соответствие между расстоянием от проекции точки до проекции осевой линии камеры.

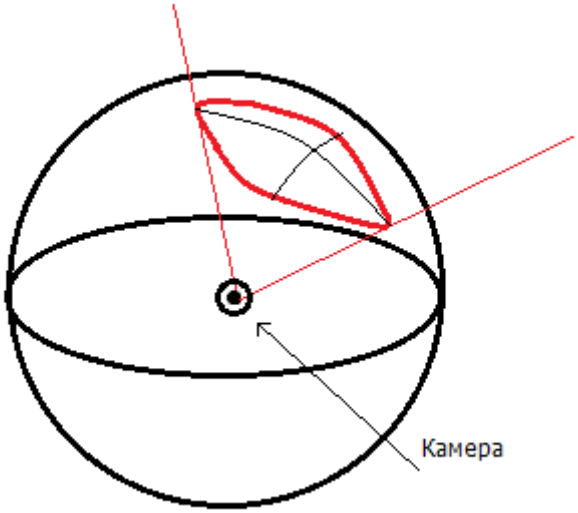


Рис. 2. Фигура, построенная с помощью контура на сфере

Затем задается вектор v, соответствующий осевой линии камеры (0, 0, -1)
И трансформируется следующим образом:

$$v.Z = -Cos(\theta)$$
$$v.X = \sqrt{1 - v.Z^2} * dX * r$$
$$v.Y = \sqrt{1 - v.Z^2} * (-dY) * r$$

Трансформированный вектор v задает направление от точки положения камеры до искомой точки в пространстве. Умножив его на расстояние от камеры до точки (в данном случае оно равно радиусу сферы, т.е. 1), мы получим координаты точки.

Дополнительные условия для кривой на сфере

В случае, когда вся кривая целиком лежит на поверхности сферы, то определить ее кривой Безье в декартовых координатах на плоскости проекции невозможно. Поэтому условия задачи были несколько изменены:

1. Сплайн Безье определен декартовыми координатами в пространстве. При этом на поверхности сферы должны находиться только узловые точки сплайна Безье; кроме того, касательные к узловым точкам должны быть также касательными к сфере. Остальные точки определяются уравнениями кубической кривой Безье по четырем коэффициентам, определенным положением узловых точек и направлениям касательных, и при этом могут находиться не на поверхностии сферы. В то же время они должны находиться достаточно близко к поверхности сферы. Из этого следует еще одно уловие:

2. Предыдущее допущение позволяет построить сплайн, все точки которого находятся достаточно близко к поверхности сферы только в том случае, если расстояние между соседними узлами сплайна достаточно невелико по сравнению с

радиусом сферы. Если же узлы сплайна находятся далеко друг от друга, то промежуточные точки между узлами могут заметно приблизиться к центру сферы (который совпадает с центром камеры) и даже попасть за пределы области видимости камеры. Поэтому в случае, если любая точка на сплайне приближается к центру на расстоянее менее 0.1 радиуса сферы, то такая ситуация считается недопустимой и в сплайн в этом месте добавляется дополнительный узел. Его координаты вычисляются таким образом, чтобы на проекции камеры (с учетом ее функции нелинейных искажений) координаты этого узла совпадали бы с координатами исходной промежуточной точки, приблизившейся к центру сферы на расстоянее менее 0.1 ее радиуса.

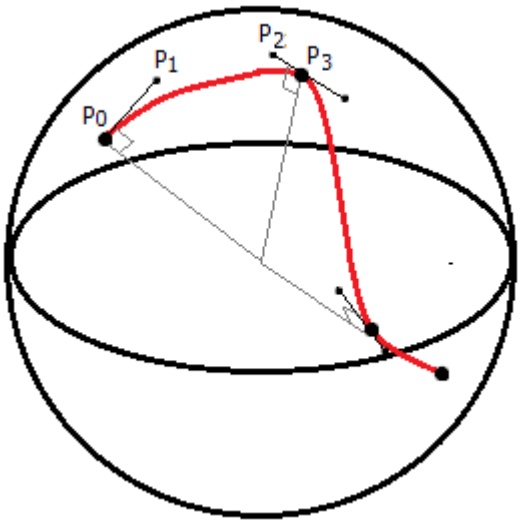


Рис. 3. Кривая Безье с узловыми точками на сфере

3. На плоскости проекции камеры по краям могут находиться области, для которых функция нелинейных искажений камеры не определена. Располагать узловые точки сплайна Безье в этих областях пользователю запрещено.

Таким образом, фигура, отображающаяся на плоскости проекции камеры, не является сплайном Безье, а итоговая кривая в пространстве, напротив, таковым сплайном является. Пользователь может задать его на плоскости проекции камеры лишь косвенно. Однако для поставленной задачи такой точности вполне достаточно.

Результаты

Используя предложенный алгоритм, была разработана компьютерная программа (являющаяся дополнением к системе автоматизированного проектирования CATIA), которая позволяет нарисовать контур поверх изображения с камеры, построить сплайн Безье, соответствующий данному контуру, и далее построить конус, опирающийся на данный контур. Итоговый объект в системе CATIA отображен на рис. 4. Используя данный объект можно определить его пересечение с какими-либо деталями кузова как визуально, так и используя в дальнейшем какие-либо алгоритмы для определения пересечения поверхностей в пространстве (которые выходят за рамки данной работы).

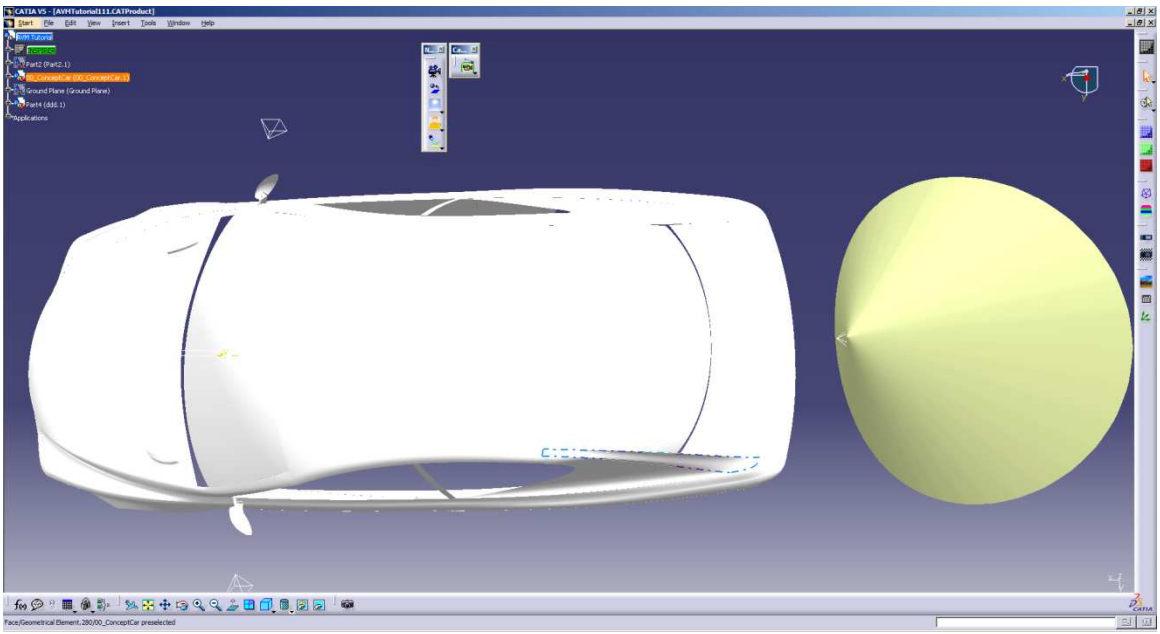


Рис. 4. Объект, построенный в системе CATIA

Литература

1. C. Hughes, M. Glavin, E. Jones, P. Denny. Wide-angle camera technology for automotive applications: a review. IET Intell. Transp. Syst., 2009
2. Alyn Rockwood and Peter Chambers. Introduction to Curves and Surfaces. SIGGRAPH 1996

Способы реализации механизма скользящего стерео-курсора в реальном времени

А.Е. Афанасьева, А.В. Игнатенко
Лаборатория Компьютерной Графики и Мультимедиа,
Факультет Вычислительной Математики и Кибернетики,
Московский Государственный Университет им. М.В. Ломоносова
afedorova@graphics.cs.msu.ru, ignatenko@graphics.cs.msu.ru

Аннотация. В современной компьютерной графике актуальны задачи восстановления формы объекта по его фотографиям, построения геометрической модели объекта по набору изображений. Данные задачи усложняются, если мы имеем дело с объектами сложной формы, с прозрачными или плохо различимыми деталями изображения. Для работы с подобного рода объектами целесообразно использовать стереомониторы.

В последнее время стерео-технологии, например, NVidia 3D Vision, достигли приемлемого уровня для профессионального использования.

Новые технологии требуют соответствующего интерфейса. Для разных задач могут понадобиться различные инструменты. В работе рассматривается задача создания трехмерного скользящего курсора для стереомонитора. Данный инструмент можно использовать как для выделения некоторых важных частей объекта или выделения границ объекта, так и для удобства пользователя при ручном построении приближенной геометрической модели объекта (пользователю не нужно дополнительно устанавливать курсор на поверхность объекта). Схожие технологии могут использоваться в различных областях виртуальной реальности ([1]).

Ключевые слова. 3D-курсор, скользящий курсор, стереопара, стерео-соответствие, диспаритет, корреляция, корреляционное окно, ведущий курсор

Введение

Стерео-изображение формируется из двух изображений стереопары, соответственно, изображение курсора складывается из двух его изображений на каждом, соответственно, на каждом изображении курсор имеет свои координаты.

Трехмерный скользящий курсор – это трехмерный курсор, который устанавливается визуально на поверхность текущего объекта (объекта с данными экранным координатами), если пользователь перемещает указатель мыши в плоскости экрана. Поскольку глубина курсора должна определяться по движению мыши, алгоритм определения глубины должен обладать высокой скоростью.

Для этого потребуется вычислять стерео-соответствие в некоторой области вокруг курсора в реальном времени.

Один из курсоров при таком подходе будет считаться ведущим. Ведущий курсор – курсор, по положению которого ищется соответствие на втором изображении стереопары.

Методы вычисления стерео-соответствий

Условно методы вычисления стерео-соответствий можно разделить на несколько основных групп:

1) Методы, основанные на корреляционном окне
2) Методы, связанные с поиском локальных особенностей на изображении
3) Построение карты глубины, карты диспаритетов

Первая группа методов использовалась в данной работе благодаря своей простоте и скорости.

Вторая группа методов – методы, связанные с поиском так называемых ключевых точек, – основаны на поиске в некотором фрагменте изображения его локальных особенностей, а затем учет их взаимного расположения для поиска соответствия конкретному заданному пикселу. В чистом виде данный подход не надежен, т.к. на изображении могут встречаться фрагменты и без локальных особенностей, а также фрагменты с недостаточным для работы количеством локальных особенностей.

Третий подход наиболее надежен, т.к. позволяет исключить выбросы и скачки диспаритета, а так же анализ всего изображения в целом, но не подходит в силу своей низкой скорости. Построение карты диспаритета использовалось в данной работе для оценки общего результата.

Метод простой корреляции

Реализован алгоритм поиска соответствий на изображениях при помощи метода корреляционного окна. Корреляционное окно – прямоугольная область на изображении с заданной высотой и шириной, по которой ищется соответствие позиций курсора. Параметры корреляционного окна могут выбираться, например, как фиксированная часть высоты или ширины изображения.

Пусть положение курсора на первом изображении стереопары – (x, y). Искомое положение курсора на втором изображении (x + dx, y). Задача состоит в нахождении диспаритета dx. Диспаритет может быть как положительный, так и отрицательный.

Дополнительные входные параметры: размер корреляционного окна (w, h), минимально и максимально возможный диспаритет (dispMax).

На втором изображении выделяется прямоугольная область с координатами (x – w / 2 – dispMax, y – h / 2, w + 2 * dispMax, h).

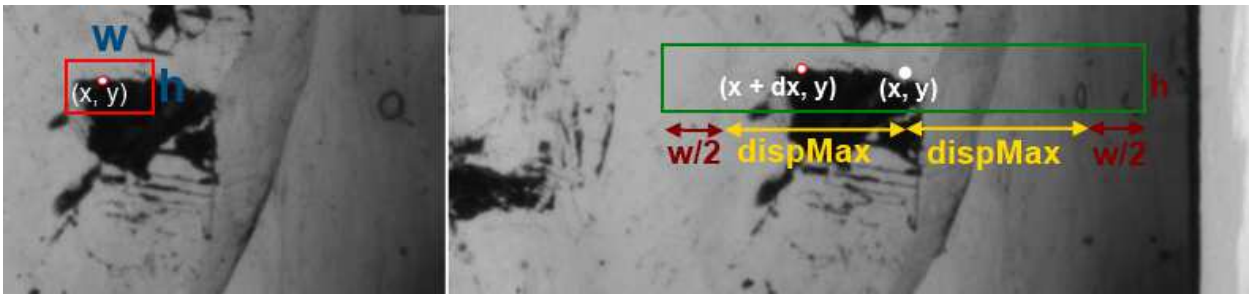


Рис. 1.Корреляционное окно. Максимальный разброс диспаритета

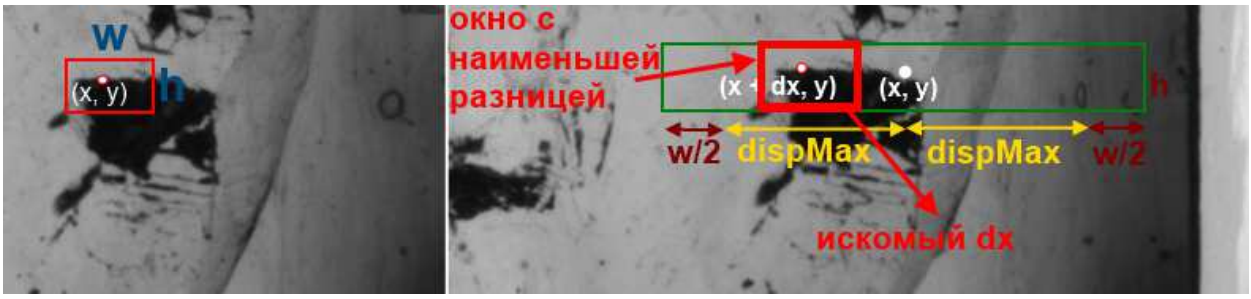


Рис. 2. Наилучшее корреляционное окно. Диспаритет

Корреляционное окно первого изображения последовательно сравнивается с каждым подокном выделенной области на втором изображении. Сравнение по метрике L2 (квадрат абсолютной разности изображений). Выбирается dx, которому

соответствует окно с минимальной разницей при сравнении с окном на первом изображении.

Поиск осуществляется исключительно по горизонтали, за счет этого сильно сокращается число сравнений, и возможен полный перебор.

Корреляция с предварительным поиском по уменьшенному изображению

При данном подходе сначала ищется диспаритет для уменьшенных изображений. Уменьшенное изображение, как правило, получается более отчетливым, и корреляция работает лучше, с меньшим количеством скачков. Корреляция по уменьшенному изображению также позволяет заметно сократить время расчета диспаритета.

Чем сильнее уменьшено изображение, по которому осуществляется предварительный поиск, тем ниже точность результата.

Данный подход позволяет значительно сократить скачки диспаритета на изображении, т.к. рассматриваемое окно на уменьшенном изображении содержит больше информации, чем исходное корреляционное окно.

График корреляции

По графику корреляции можно судить о достоверности полученного диспаритета.

Рис. 3. Неявный минимум на графике корреляции

Рис. 4. Размытый минимум на графике корреляции. Объект нечеткий или полупрозрачный. Возможно, потребуется уточнение результата

Рис. 5. Явный минимум на графике корреляции. Результат достоверен

Рис. 5. Явный минимум отсутствует.
Корреляционное окно слишком мало, а объект незаметен

Корреляция с предварительным поиском по уменьшенному изображению с учетом диспаритета в соседних пикселах

В данном алгоритме диспаритет ищется не только в текущем пикселе, но и в соседних. Соседние пикселы берутся на расстоянии в 5 или 10 раз меньше размера корреляционного окна.

В тестовой программе возможен выбор 2, 4, 6, 8 и 24 соседних.

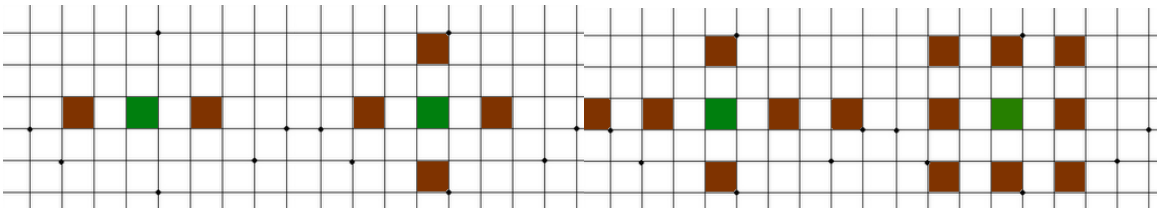


Рис. 6. Расположение пикселов-соседей для дополнительного расчета диспаритета

Для каждого пиксела-соседа строится график корреляции. У каждого графика находится некоторое фиксированное количество самых сильных локальных минимумов (на текущем этапе по результатам тестирования лучший результат показывает алгоритм, в котором берется один минимум). После того как построены графики корреляции для всех соседей и рассчитаны все минимумы, выбирается тот минимум, который чаще всего встречается на графике корреляции.

Домножение на весовую функцию

В текущей реализации алгоритма все графики корреляции домножаются на весовую функцию.

В настройках тестовой программы возможно домножение на линейную функцию или на экспоненту.

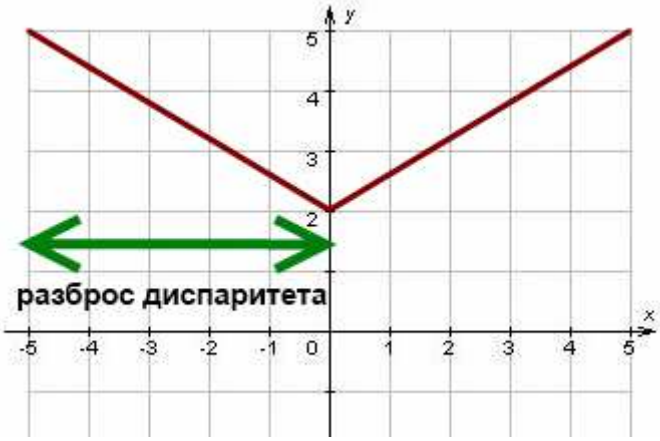


Рис. 7. Линейная весовая функция для графика корреляции

Для улучшения результата работы достаточно домножения на линейную функцию. Таким образом будет понижен “вклад” в результат локальных минимумов графика корреляции, образующих слишком большой диспаритет.

В алгоритме, который учитывает значения диспаритета в соседних пикселах, на весовую функцию домножается, соответственно, график корреляции для каждого пиксела-соседа.

Рис. 8. Исходная стереопара

Рис. 9. Справа – левое изображение стереопары,
слева – построенное по карте диспаритета

Результаты

Алгоритм хорошо работает на четких границах крупных объектов. В случае характерных регулярных особенностей на изображении (сеток, решеток, сложных узоров и т.п.), если ячейка единичной структуры много меньше размера окна, возможны ошибки.

Далее приведен результат работы алгоритма. Для стереопары построена карта диспаритетов. Затем по правому изображению с учетом карты диспаритетов реконструируется левое изображение стереопары. По полученной реконструкции и ее сходству/различию с оригинальным изображением можно судить о результате работы алгоритма.

Заключение

Поиск по уменьшенному изображению и домножение на весовую функцию сглаживают результат, позволяя убрать лишние скачки диспаритета.

Размер корреляционного окна выбирается автоматически в зависимости от размера изображения, и не всегда является оптимальным. Ошибки возникают, например, в случае существенного различия во взаимном расположении объектов на изображении стереопары.

В случае работы с простыми изображениями (например, четкие фотографии) предложенный подход дает минимальное число ошибок.

Для борьбы с ошибками в сложных случаях потребуется алгоритм, учитывающий зависимости во взаимном расположении объектов.

Список литературы

[1] Robert J. Teather, Wolfgang Stuerzlinger. Investigating One-Eyed and Stereo Cursors for 3D Pointing Tasks. Dept. of Computer Science & Engineering, York University

[2] Robert J. Teather, Wolfgang Stuerzlinger. Cursors for 3D Pointing. CHI’12, May 5–10, 2012, Austin, Texas, USA.

Определение индекса преломления прозрачного материала с помощью визуализации

В.В. Афанасьев, А.В. Игнатенко, И.О. Тисевич
Институт прикладной математики РАН им. М.В. Келдыша,
Лаборатория Компьютерной Графики и Мультимедиа
Факультета ВМК МГУ им. М.В. Ломоносова
vafanasjev@graphics.cs.msu.ru, ignatenko@graphics.cs.msu.ru,
itisevich@graphics.cs.msu.ru

Аннотация. В данной работе описываются способы определения индекса преломления материала путём его подбора и сравнения визуализированных изображений с этим индексом преломления и оригинальных фотографий материала. Метод не требует специальных источников света, в отличие от прямых методов, и накладывает меньшие ограничения на форму исследуемого объекта. Дополнительно требуется восстановление формы исследуемого объекта с достаточной точностью.

Ключевые слова: индекс преломления, показатель преломления, коэффициент преломления, синтез изображений.

Введение

Задача определения оптических свойств материалов возникает достаточно часто, например, при создани новых типов стекла. Существует множество прямых методов измерения индекса преломления, основанных на пропускании узкого луча света через призму, вырезанную из исследуемого материала. К таким призмам предъявляются различные требования, которые усложняют их изготовление.[1] Методы прямого измерения не всегда применимы из-за необходимости придавать материалу особую форму, иметь узко направленный источник света.

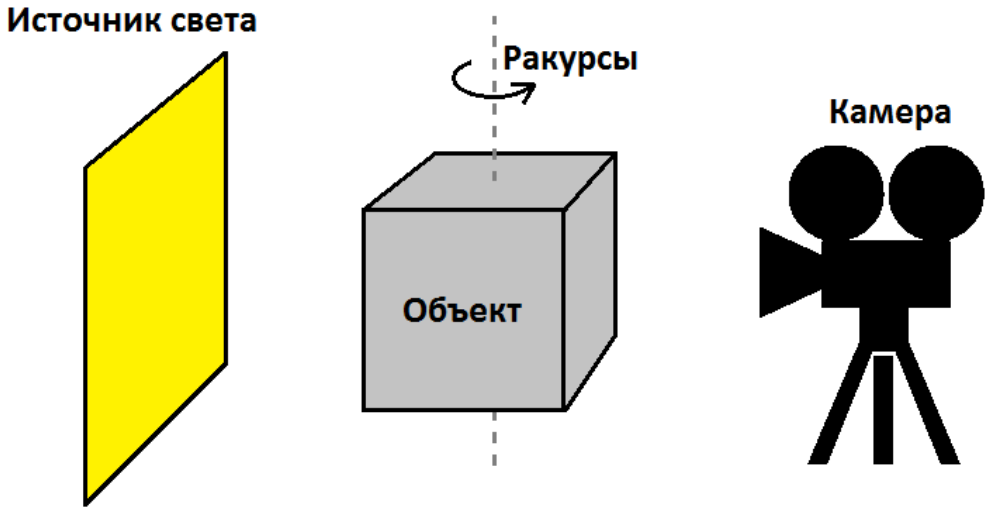
Также существуют иммерсионные методы, с помощью которых измеряется в том числе показатель преломления. Они сводятся к тому, чтобы выбрать из набора иммерсионных жидкостей наиболее похожие на материал по индексу преломления, погрузить материал в жидкость и визуально наблюдать наличие или отсутствие эффектов, связанных с преломлением на границе вещества с иммерсией.[2] Недостатком иммерсионных методов является то, что нужно в итоге получить иммерсионную жидкость с показателем преломления, очень близким к измеряемому, и при этом известным. Иммерсионные жидкости являются расходным материалом, т.к. их смешивают для получения подходящего значения индекса преломления. Кроме того, исследуемый материал должен быть твёрдым.

Предлагается метод, основанный на фотографировании объекта, сделанного из исследуемого материала, моделировании всех известных условий съёмки (положения камеры, формы объекта, параметров источника света) и визуализации этой сцены с корректным расчётом распространения света в сцене. Путём изменения неизвестного индекса преломления материала нужно максимизировать сходство фотографии и результата визуализации по некоторому выбранному признаку и таким образом подобрать коэффициент преломления.

Параметры съёмки

Исследуемый материал снимается с нескольких ракурсов, положение этих ракурсов запоминается для последующей визуализации. Материал должен быть освещён источником света с той длиной волны, для которой определяется индекс

преломления. В идеальном случае это монохроматический источник, например, лазер. Чем шире спектр источника света, тем больше будет дисперсия в исследуемом материале, тем менее точно возможно будет измерить индекс преломления (тогда имеет смысл говорить об измерении индекса преломления для некоторой “средней” длины волны).



Без калибровочного объекта

Данный способ применим в случае, если в наличии имеется только исследуемый материал. В этом случае материалу требуется придать заранее известную форму с небольшим числом граней (например, куб или призма). На фотографиях с разных ракурсов можно наблюдать в ближних гранях объекта образы его дальних граней и рёбер. Положение этих образов зависит от индекса преломления и может быть сопоставлено с результатами визуализации.

Проблемы:

- Не все материалы шлифуются так, что рёбра остаются чёткими, могут появиться сколы, которые неточно отсканированы или слишком толстые, чтобы сопоставить их с нужной точностью.
- Не всякая форма объекта подходит для наблюдения на просвет его обратных граней при большом значении индекса преломления, т.к. на части ракурсов через эти грани может быть не виден источник света, и они будут полностью тёмными. Тем не менее, этот способ не накладывает таких сильных ограничений на форму объекта, как прямые методы измерения.

С калибровочным объектом

Данный способ основан на использовании твёрдого калибровочного объекта с заранее известным индексом преломления.

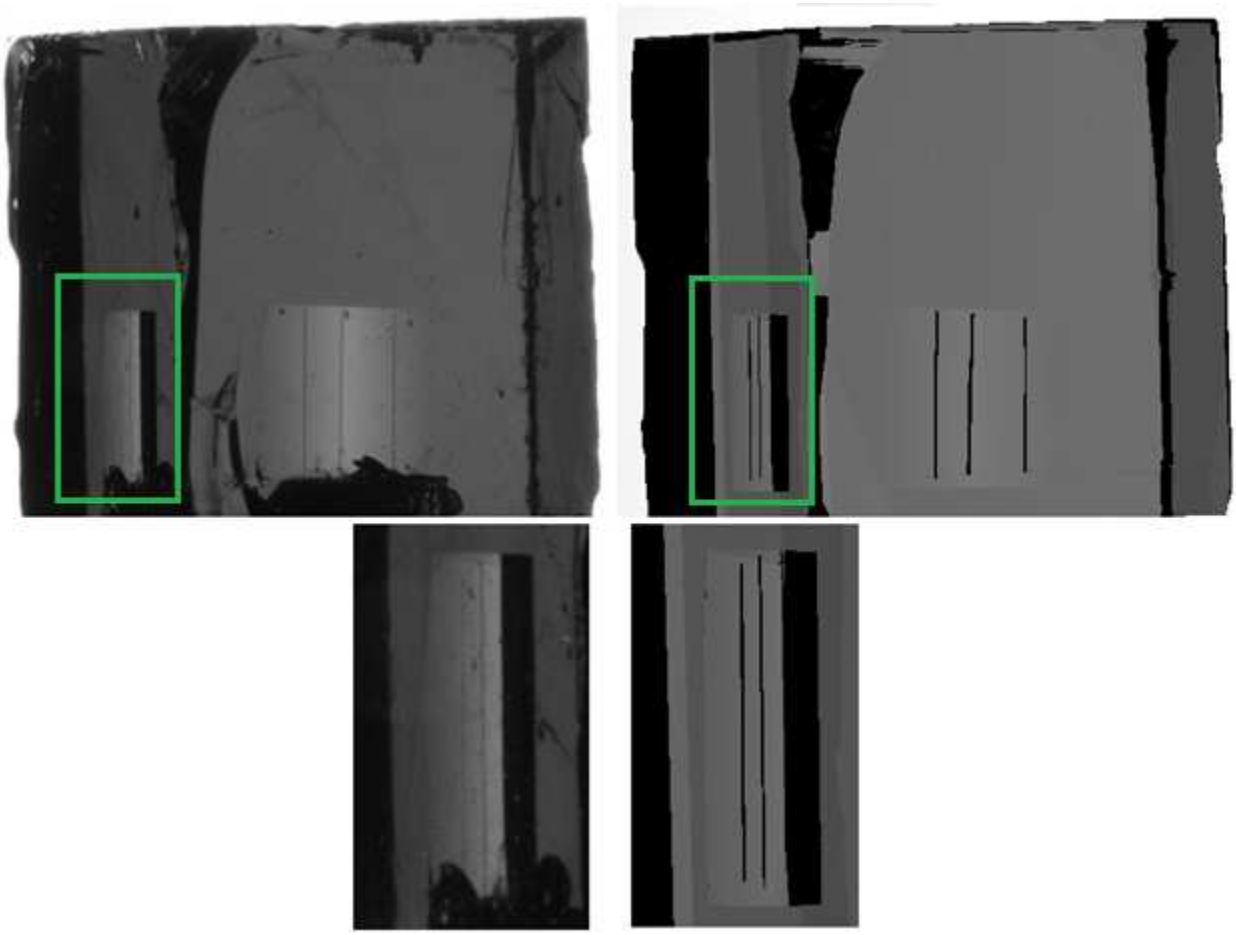
Калибровочный объект может иметь произвольную, или заданную форму. Пример – призма из стекла. Он заливается расплавом исследуемого материала в форму, содержащую небольшое число плоских граней (например: куб, шестигранная призма), чтобы через каждую грань калибровочный объект был виден целиком. Если материал может находиться в твёрдом состоянии при нормальных условиях, после его отвердевания грани формы шлифуются для уменьшения визуальных искажений калибровочного объекта. В случае, если материал жидкий, возможно заполнить им такую же твёрдую форму, например, из стекла, а внутри поместить калибровочный

объект. В этом случае индекс преломления материала формы и её толщина также должны быть известны.

Снимается последовательность фотографий объекта с разных ракурсов при вращении объекта вокруг вертикальной оси. Выбирается интервал, внутри которого должен лежать истинный показатель преломления материала. В последовательности выделяются некоторые характерные кадры, после чего строятся изображения сцены для данных ракурсов и разных индексов преломления из выбранного интервала. Шаг изменения индекса преломления выбирается меньшим максимальной точности метода. Далее производится сравнение получившихся изображений с фотографиями на предмет сходства эффектов, зависящих от искомого индекса преломления. Оно может быть как ручным, так и автоматическим.

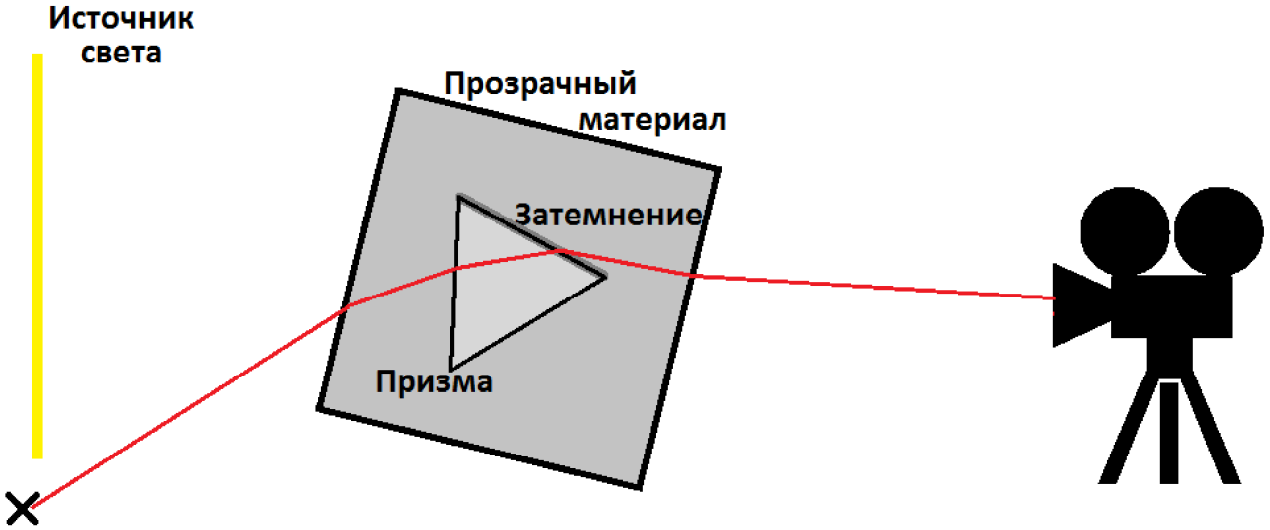
По положению затемнений

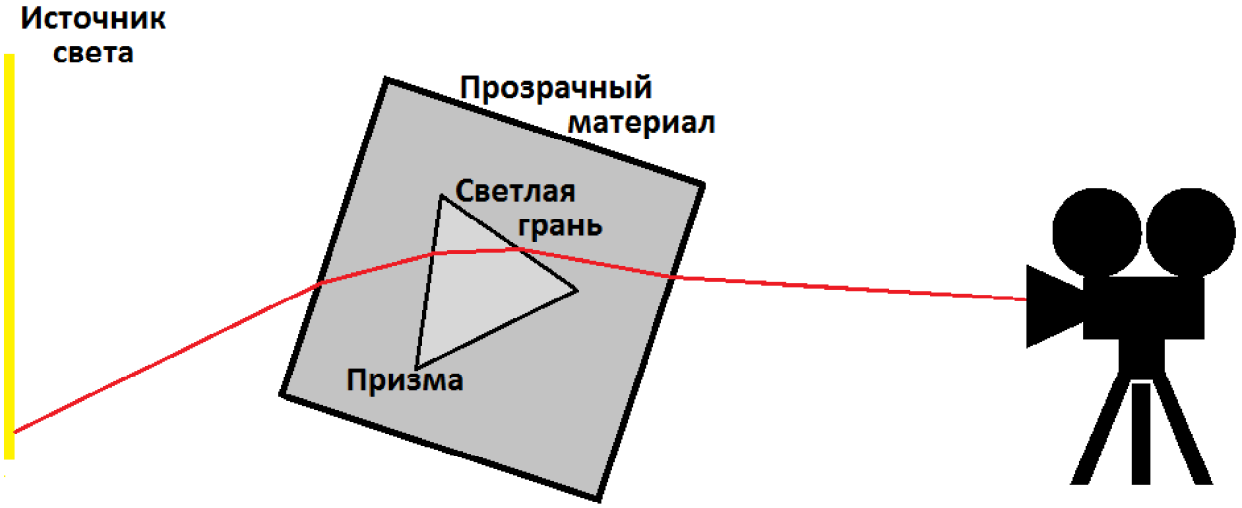
При разнице в показателях преломления у тестируемого материала и калибровочного объекта на границе между ними свет преломляется и отклоняется от прямой траектории. Поэтому если поставить за исследуемым объектом источник света известной формы и снимать объект на просвет, на калибровочном объекте будут видны светлые и тёмные области, соответствующие источнику света и тёмному фону. Если точно задать форму источника света, то при совпадении предполагаемого индекса преломления материала с настоящим кадры, содержащие одинаковые затемнения, будут иметь одни и те же номера в последовательности фотографий и визуализированных изображений.



На рисунке представлены фотография (слева) и синтезированное изображение (справа). На правой грани призмы, наблюдаемой через грань иммерсионного куба, видно затемнение. Оно есть на фотографии и на синтезированном изображении.

Для определения индекса преломления таким способом из последовательности ракурсов выбираются те, на которых затемнение на некоторой грани появляется и исчезает. Путём изменения индекса преломления при визуализации достигается синхронность появления и исчезновения затемнений на последовательности фотографий и на синтезированной последовательности. На рисунках показано появление и исчезновение затемнения на грани призмы внутри иммерсионного куба при незначительном повороте:





Преимуществом метода является возможность работы с калибровочным объектом произвольной формы. Однако формы, имеющие плоские грани, дают более точный результат, т.к. затемнение на такой грани появляется и исчезает целиком, а не постепенно, за меньшее число кадров последовательности.

Погрешность метода определяется погрешностью построения формы источника света, формы исследуемого объекта и формы калибровочного объекта, также угловым разрешением последовательности снимков. Неточности в форме источника света и объектов дают рассогласованность затемнений на различных ракурсах, что позволяет оценить точность конкретного измерения.

Проблемы:

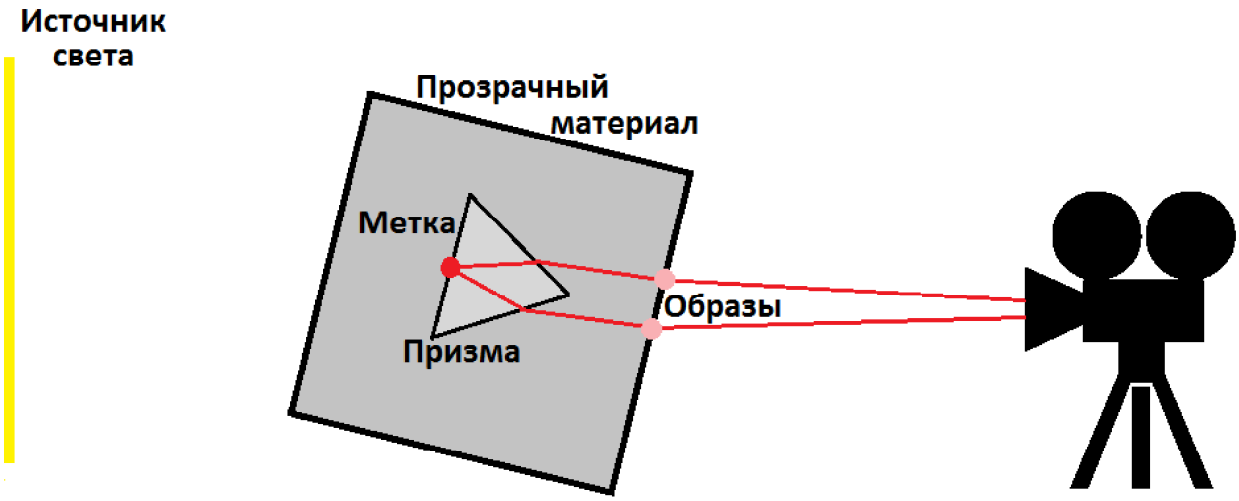
- Неточность формы объекта

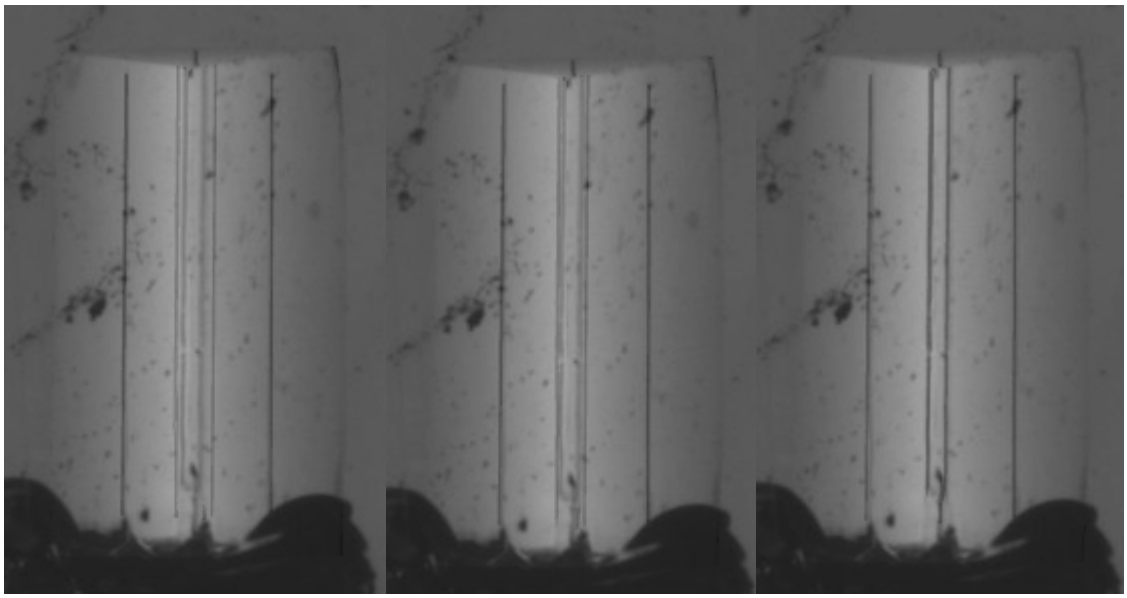
- Неточность формы источника света
- Затемнения иной природы

По меткам

Данный метод может быть применён при относительно небольшой разнице в индексах преломления исследуемого материала и калибровочного объекта.

Используется калибровочный объект в форме прямой призмы (например, трёхгранной). На грани призмы наносятся вертикально непрозрачные метки. При наблюдении объекта с ракурса, на котором одно из вертикальных рёбер призмы находится между меткой на противоположной грани и камерой, видно 2 образа этой метки в 2 соседних гранях призмы.





Расстояние между образами и их положение зависят от угла между гранями призмы и от разницы в индексах преломления.

На рисунках показана фотография призмы в иммерсионном стекле, на которую наложены синтезированные изображения с того же ракурса с разными индексами преломления стекла. На левом рисунке видно несовпадение 2-х образов меток в центре кадра на фотографии и на синтезированном изображении. Это говорит о том, что индекс преломления иммерсии подобран неправильно. Из последовательности синтезированных изображений для разных индексов преломления с этого ракурса

выбирается то, на котором образы меток совпадают с метками на фотографии, соответствующий индекс преломления считается истинным. Совпадение показано на правом рисунке.

Проблемы:

- Толщина меток
- Сколы ребра калибровочного объекта
- Малое изменение положения образов меток при большом угле между гранями

Ручной и автоматический подбор

В данной работе определение индексов преломления различных материалов осуществлялось только вручную. Однако на основе предложенных методов возможно реализовать автоматический подбор

Визуализация не позволяет даже при правильном подборе всех параметров получить изображение, полностью совпадающее с фотографией, так как учитываются не все эффекты, не идеальны исследуемый материал (содержит загрязнения) и калибровочный объект. Поэтому простая минимизация абсолютной разности между фотографией и синтезированным изображением может не привести к истинному значению искомого показателя преломления.

Для оценки правильности выставленного при синтезе изображения индекса преломления нужно выбрать некоторую характерную зону (затемнение или образ метки), чтобы неверная разница на других частях изображения не вносила негативный вклад.

При использовании затемнений абсолютная разность изображений в выбранной области будет однозначно указывать на совпадение или несовпадение затемнений, если затемнение образовалось на плоской грани (т.к. обычно либо вся грань затемнена, либо освещена). Для автоматического подбора индекса преломления нужно только правильно выделить эти области на кадрах в последовательности.

В случае, если затемнение видно на криволинейной поверхности, форма затемнения сильно зависит от кривизны поверхности. При небольшой погрешности построения модели форма затемнения на синтезированном изображении может сильно отличаться от формы на фотографии. Здесь для автоматического подбора потребуется более сложная метрика, учитывающая эти особенности.

Если для подбора индекса преломления используется объект с метками, то для автоматического метода нужно отделять образы меток от шумов на фотографии и загрязнений в исследуемом материале.

Заключение

С помощью предложенных методов был измерен индекс преломления различных типов иммерсионного стекла. Точность измерения составила ~0.002 для метода затемнений и 0.001 – 0.003 для метода меток. Последовательности изображений для подбора индекса преломления были синтезированы методом трассировки лучей с использованием графического процессора для ускорения.

Список литературы

1. ГОСТ 28869-90. Материалы оптические. Методы измерений показателя преломления, 2005

2. Татарский В.Б. Кристаллооптика и иммерсионный метод. М., Недра, 1965

GMLePublish: web-система оценки алгоритмов тональной компрессии HDR-изображений

М.А. Матросов, А.В. Игнатенко
Лаборатория Компьютерной Графики и Мультимедиа,
Факультет Вычислительной Математики и Кибернетики,
Московский Государственный Университет им. М.В. Ломоносова
{matrosov, ignatenko}@graphics.cs.msu.ru

Аннотация. Многие современные методы обработки изображений не имеют объективной метрики для оценки их качества. К таким методам относится тональная компрессия изображений с широким динамическим диапазоном яркости (HDR-изображений). Такие изображения хранят в себе карту яркости сцены и не могут быть напрямую визуализированы на современных устройствах вывода. Методы тональной компрессии производят интеллектуальное сжатие динамического диапазона HDR-изображений и позволяют получать обычные изображения с низким динамическим диапазоном яркости (LDR-изображения).

При анализе качества работы методов тональной компрессии существенную роль играют субъективные оценки. Более того, для анализа промышленных изображений особенно важны оценки от экспертов в данной области. В связи с этим существует потребность в автоматизированной системе сбора и агрегации таких оценок. Эта система должна быть доступна для пользователей в любом месте и в любое время, т.к. эксперты могут иметь жёсткий график работы.

В данной работе предлагается подобная система в виде web-приложения на базе технологии Microsoft ASP.NET. Система предоставляет администраторам возможность загружать HDR-изображения и интегрировать реализации различных методов тональной компрессии. Она предоставляет функционал для удобного сбора пользовательских оценок. Оценки сохраняются в базе данных на Microsoft SQL Server и ассоциируются с конкретным пользователем. Собранные оценки впоследствии анализируются администраторами системы.

Ключевые слова: система оценки, система публикации, web, тональная компрессия, HDR-изображения.

Введение

Современные устройства ввода (такие как цифровые камеры) могут записать лишь небольшой динамический диапазон яркости (dynamic range of luminance). Для потребительских цифровых камер он составляет порядка 2000:1. В то же время, динамический диапазон воспринимаемой человеческим глазом яркости составляет порядка 10^9:1. Если не принимать во внимание возможность аккомодации глаза, то динамический диапазон воспринимаемой в рамках одной сцены яркости составляет порядка 10000:1 (данные взяты из [1, стр. 191]). Таким образом, невозможно в рамках одной фотографии запечатлеть весь динамический диапазон, доступный человеческому глазу.

Для преодоления этой проблемы используется технология изображений с широким динамическим диапазоном яркости (HDR-изображения). Они позволяют сохранять полную карту яркости сцены. Существует несколько способов их получения (см. [1, гл. 4]). Однако динамический диапазон современных дисплеев составляет порядка 1000:1, и они не могут напрямую отобразить HDR-изображения. Поэтому для визуализации HDR-изображения необходимо выполнить интеллектуальное сжатие его динамического диапазона с сохранением деталей и содержания. Этот процесс называется тональной компрессией (tone mapping, см. [2],

[3]). Далее по тексту мы будем сокращать термин «метод тональной компрессии» до ТМО (tone mapping operator).

На данный момент существует очень мало объективных метрик для оценки качества ТМО. Например, в [4] вычисляются карты пикселей, в которых произошли определённые изменения контраста в результате тональной компрессии. Однако итоговая оценка приемлемости полученного LDR-изображения сильно зависит от содержания исходной сцены, и надёжно её может дать только человек. В связи с этим существует потребность в субъективных метриках. Несколько подходов для субъективного тестирования предложены в [5], [6] и [7].

В данной работе мы описываем систему GMLePublish, которая используется в Лаборатории Компьютерной Графики и Мультимедиа для сбора субъективных оценок качества работы алгоритмов тональной компрессии.

Терминология

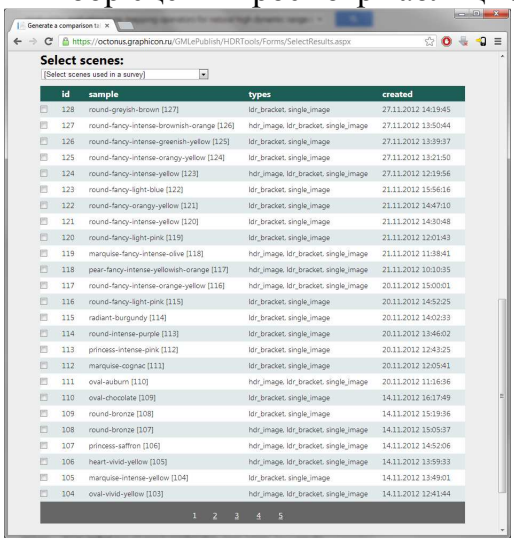
- Система – web-приложение GMLePublish
- Пример – HDR-изображение сцены или набор обычных LDR-фотографий сцены с разными экспозициями (LDR-bracket)
- ТМО – метод тональной компрессии (tone mapping operator)

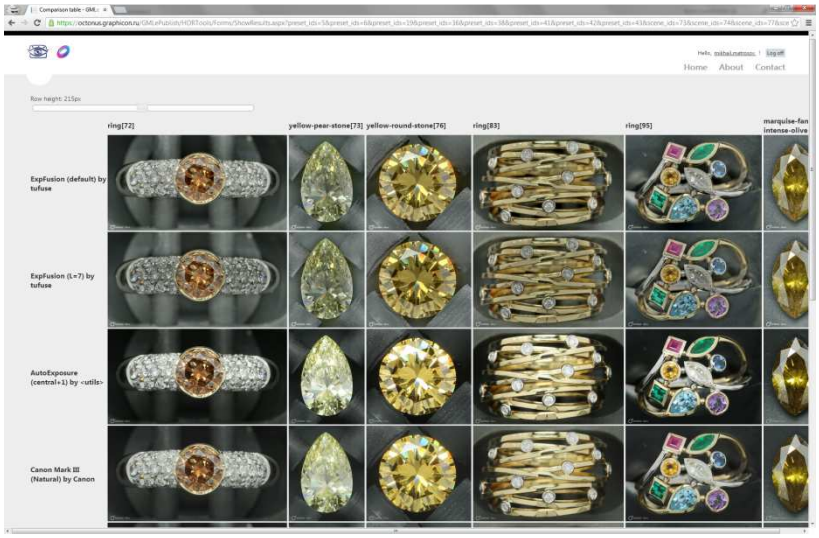
Общее описание

Система реализована в виде web-приложения и доступна в сети Интернет. Система предоставляет пользователям следующий функционал:

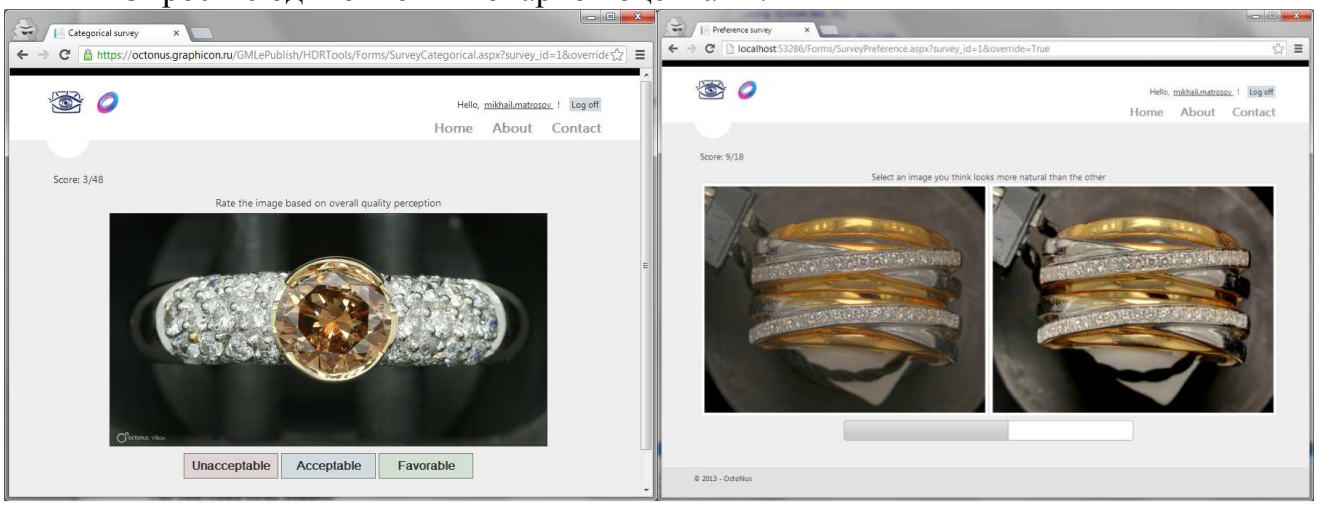
- Просмотреть список доступных примеров
- Вывести таблицу обработки различных примеров различными ТМО
- Пройти опрос с попарным сравнением по методу [6]
- Пройти опрос с одиночными оценками по методу [7]

Элементы управления для описанных действий приведены на рисунках ниже. Выбор сцен и просмотр таблицы:





Опросы с одиночной и попарной оценками:



Администраторы имеют доступ к следующим настройкам системы:

- Добавить/удалить примеры
- Добавить/удалить реализации ТМО
- Выбрать множество примеров и ТМО для создания опросов
- Посмотреть оценки пользователей

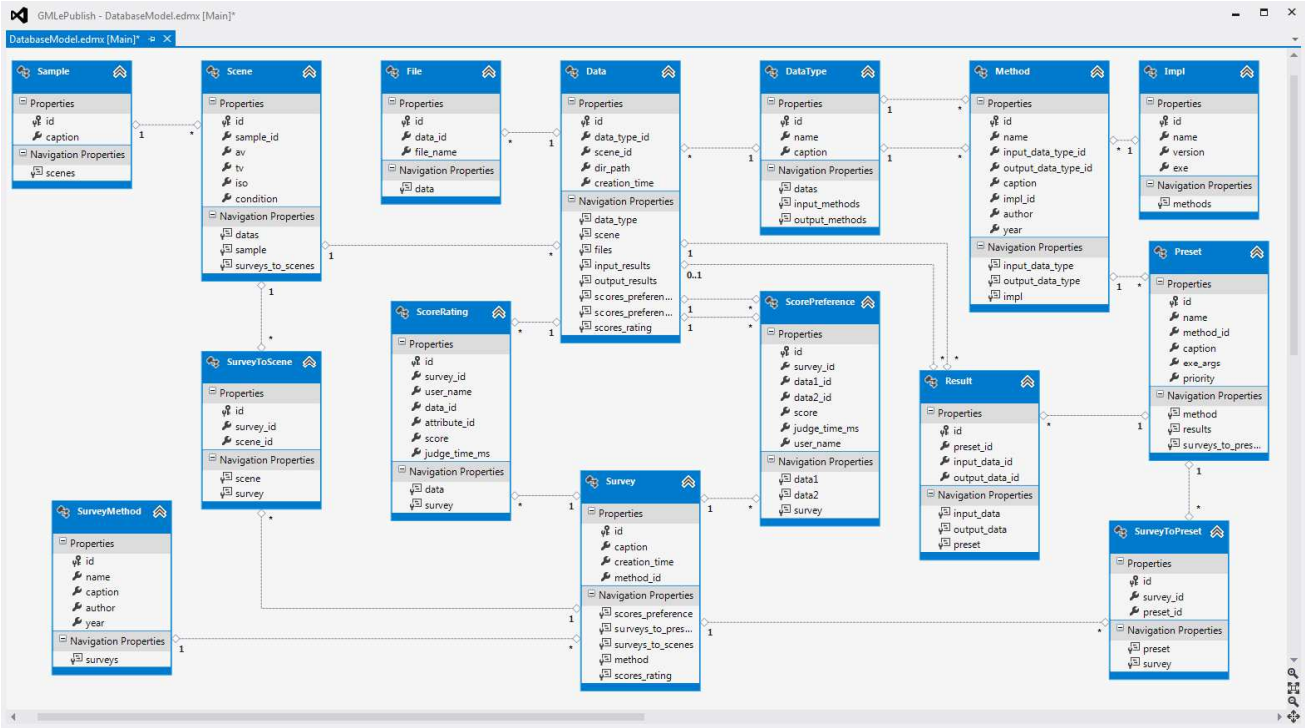
Технические характеристики системы

Система разработана на платформе Microsoft ASP.NET с использованием IDE Microsoft Visual Studio 2012. База данных для хранения оценок и настроек системы находится под управлением Microsoft SQL Server 2012. Веб-сервер работает под управлением Microsoft IIS 7.5. Серверная логика реализована на языке C# под Microsoft .NET 4.5. Клиентская логика написана на JavaScript с использованием библиотеки jQuery.

Запуск реализации ТМО происходит автоматически в момент генерации таблицы с результатами. Ключи для запуска и путь к соответствующему исполнимому файлу хранятся в базе данных. Полученные результаты сохраняются на диск, и в базе данных делается пометка о том, что результат применения данного ТМО к данному примеру уже получен. При повторном обращении к результату он не вычисляется заново.

Организация базы данных

База данных системы используется для хранения примеров, ссылок на реализации ТМО, оценок экспертов и т.п. Каждой таблице в базе данных ставится в соответствие класс в серверном коде на языке C# с помощью технологии Entity Framework. Связь между классами приведена на рисунке ниже.



- **Sample**. *Пример* или *объект*, описывает некоторый конкретный материальный объект, например конкретный образец породы, определённое ювелирное изделие и т.п.
- **Scene**. *Сцена*, совокупность объекта и условий съёмки. Включает настройки камеры и освещения. Каждый объект может быть снят в нескольких сценах.
- **Data**. *Данное*, совокупность файлов, соответствующих определённой сцене. Каждой сцене может соответствовать множество данных. К ним относятся как исходные данные – набор фотографий с разными экспозициями, так и синтезированные данные – HDR-изображение, полученное из набора фотографий, изображения, полученные в результате применения различных ТМО.
- **DataType**. *Тип данного*, задаёт формат файлов, содержащихся в определённом данном:
    - ldr_bracket – набор фотографий с разными экспозициями
    - hdr_image – полученное по набору HDR-изображение
    - ldr_image – изображение, результат применения ТМО к HDR-изображению
- **Method**. *Метод* или *алгоритм*, описывает определённый алгоритм со ссылкой на автора. Принимает на вход данное одного типа и генерирует данное другого типа. Например, методы ТМО будут принимать на вход данные типа hdr_image и генерировать данные типа ldr_image, а алгоритмы синтеза HDR-изображений будут принимать данные типа ldr_bracket и генерировать данные типа hdr_image.
- **Impl**. *Реализация*, ссылка на исполнимый файл, в котором реализован данный метод, а также информация о разработчике.
- **Preset**. *Настройка* или *сценарий*, задаёт параметры для метода, указывает аргументы командной строки для исполнимого файла.

- **Result**. *Результат*, связывает данное со сценарием. Хранит информацию о том, что конкретное данное на самом деле является результатом применения определённого метода к определённому данному.
- **Survey**. *Опрос*, задаёт совокупность данных и методов, которые может оценить пользователь.
- **SurveyMethod**. *Тип опроса*, задаёт конкретную методологию сбора оценок со ссылкой на автора.
    - survey_rating – опрос по методу [7]
    - survey_preference – опрос по методу [6]
- **SurveyToPreset**. Задаёт набор сценариев, используемых для опроса.
- **SurveyToScene**. Задаёт набор сцен, используемых для опроса.
- **ScoreRating**. *Рейтинговая оценка*, хранит рейтинг (целое число), который присвоил определённый пользователь определённому данному в рамках определённого опроса.
- **ScorePreference**. *Сравнительная оценка*, хранит предпочтение определенного пользователя из двух определённых данных в рамках определённого опроса.

Приведённая структура классов позволяет создавать достаточно гибкую систему, в которую при необходимости можно добавлять не только новые примеры и методы тональной компрессии, но и новые виды опросов или типов данных. В определённый момент у авторов работы возникла необходимость добавить в систему методы класса Exposure Fusion, такие как [8] и [9]. Их отличительная особенность в том, что они получают итоговое изображения непосредственно из набора исходных изображений, без синтеза HDR-изображения. Благодаря возможности указывать методам типы входных и выходных данных, эта задача решилась просто созданием дополнительных методов с входными данными типа ldr_bracket и выходными данными типа ldr_image. Отделение сцены от данного позволило унифицировать доступ к классическим ТМО и к методам класса Exposure Fusion.

Заключение

В данной работе мы описали систему GMLePublish, которая позволяет пользователям предоставлять оценки качества работы методов тональной компрессии. За счёт того, что система реализована в виде web-приложения, пользователю требуется минимальное количество времени и усилий для предоставления оценок. Это позволяет исследователям быстро собирать обширные данные, необходимые для развития алгоритмов тональной компрессии.

Описанная система была успешно использована в Лаборатории Компьютерной Графики и Мультимедиа для сбора субъективных оценок алгоритмов тональной компрессии для образцов ювелирной промышленности [10]. В процессе использования она показала высокую надёжность и удобство в использовании, а также простоту доступа к результатам тестирования для их дальнейшего анализа.

Список литературы

[1] Erik Reinhard, Greg Ward, Sumanta Pattanaik, and Paul Debevec, *High Dynamic Range Imaging: Acquisition, Display and Image-Based Lighting*, Morgan Kaufmann Publishers, Dec. 2005.

[2] E. Reinhard, M. Stark, P. Shirley, and J. Ferwerda, “Photographic tone reproduction for digital images,” *ACM Transactions on Graphics (TOG)*, vol. 21, no. 3, pp. 267–276, 2002.

[3] M. Ashikhmin, “A tone mapping algorithm for high contrast images,” in *Proceedings of the 13th Eurographics workshop on Rendering*. Eurographics Association, 2002, pp. 145–156.

[4] T.O. Aydin, R. Mantiuk, K. Myszkowski, and H.P. Seidel, “Dynamic range independent image quality assessment,” in *ACM Transactions on Graphics (TOG)*. ACM, 2008, vol. 27, p. 69.

[5] M. Čadík, M. Wimmer, L. Neumann, and A. Artusi, “Evaluation of hdr tone mapping methods using essential perceptual attributes,” *Computers & Graphics*, vol. 32, no. 3, pp. 330–349, 2008.

[6] P. Ledda, A. Chalmers, T. Troscianko, and H. Seetzen, “Evaluation of tone mapping operators using a high dynamic range display,” *ACM Transactions on Graphics (TOG)*, vol. 24, no. 3, pp. 640–648, 2005.

[7] M. Kuhna, M. Nuutinen, and P. Oittinen, “Method for evaluating tone mapping operators for natural high dynamic range images,” in *IS&T/SPIE Electronic Imaging*. International Society for Optics and Photonics, 2011, pp. 78760O–78760O.

[8] T. Mertens, J. Kautz, and F. Van Reeth, “Exposure fusion: A simple and practical alternative to high dynamic range photography,” in *Computer Graphics Forum*. Wiley Online Library, 2009, vol. 28, pp. 161–171.

[9] R. Shen, I. Cheng, J. Shi, and A. Basu, “Generalized random walks for fusion of multi-exposure images,” *Image Processing, IEEE Transactions on*, vol. 20, no. 12, pp. 3634–3646, 2011.

[10] М.А. Матросов, А.В. Игнатенко, «Субъективная оценка алгоритмов тональной компрессии HDR-изображений для образцов ювелирной промышленности с помощью системы GMLePublish», Сборник трудов научно-практического семинара «Новые информационные технологии в автоматизированных системах-16». М.: ИПМ, 2013 г. сс. 43-50

Субъективная оценка алгоритмов тональной компрессии HDR-изображений для образцов ювелирной промышленности с помощью системы GMLePublish

М.А. Матросов, А.В. Игнатенко
Лаборатория Компьютерной Графики и Мультимедиа,
Факультет Вычислительной Математики и Кибернетики,
Московский Государственный Университет им. М.В. Ломоносова
{matrosov, ignatenko}@graphics.cs.msu.ru

Аннотация. В работе рассматриваются алгоритмы тональной компрессии изображений с широким динамическим диапазоном яркости (HDR-изображений). Такие изображения хранят в себе карту яркости сцены и не могут быть напрямую визуализированы на современных устройствах вывода. Методы тональной компрессии производят интеллектуальное сжатие динамического диапазона HDR-изображений и позволяют получать обычные изображения с низким динамическим диапазоном яркости (LDR-изображения). На данный момент не существует достаточно надёжных объективных метрик для оценки качества работы методов тональной компрессии. Поэтому широко используются субъективные метрики, основанные на результатах опросов некоторой выборки участников. В данной работе мы сравниваем результаты работы ряда алгоритмов тональной компрессии на основе субъективных оценок. Оценки были собраны с помощью системы GMLePublish, разработанной в Лаборатории Компьютерной Графики и Мультимедиа для этих целей.

Ключевые слова: тональная компрессия, HDR-изображения, субъективные метрики, экспертные оценки.

Введение

Современные устройства ввода (такие как цифровые камеры) могут записать лишь небольшой динамический диапазон яркости (dynamic range of luminance). Для потребительских цифровых камер он составляет порядка 2000:1. В то же время, динамический диапазон воспринимаемой человеческим глазом яркости составляет порядка 10^9:1. Если не принимать во внимание возможность аккомодации глаза, то динамический диапазон воспринимаемой в рамках одной сцены яркости составляет порядка 10000:1 (данные взяты из [1, стр. 191]). Таким образом, невозможно в рамках одной фотографии запечатлеть весь динамический диапазон, доступный человеческому глазу.

Для преодоления этой проблемы используется технология изображений с широким динамическим диапазоном яркости (HDR-изображения). Они позволяют сохранять полную карту яркости сцены. Существует несколько способов их

получения (см. [1, гл. 4]). Однако динамический диапазон современных дисплеев составляет порядка 1000:1, и они не могут напрямую отобразить HDR-изображения. Поэтому для визуализации HDR-изображения необходимо выполнить интеллектуальное сжатие его динамического диапазона с сохранением деталей и содержания. Этот процесс называется тональной компрессией (tone mapping, см. [2], [3]). Далее по тексту мы будем сокращать термин «метод тональной компрессии» до ТМО (tone mapping operator).

На данный момент существует очень мало объективных метрик для оценки качества ТМО. Например, в [4] вычисляются карты пикселей, в которых произошли определённые изменения контраста в результате тональной компрессии. Однако итоговая оценка приемлемости полученного LDR-изображения сильно зависит от содержания исходной сцены, и надёжно её может дать только человек. В связи с этим существует потребность в субъективных метриках. Несколько подходов для субъективного тестирования предложены в [5], [6] и [7].

В данной работе мы описываем результаты субъективного тестирования ряда ТМО, проведённого в Лаборатории Компьютерной Графики и Мультимедиа. В качестве участников тестирования выступали сотрудники лаборатории. В качестве примеров для тестирования выступали изображения с ювелирными изделиями. Обычно субъективные тестирования носят общий характер и в качестве примеров выбираются сцены из повседневной жизни: пейзажи, портреты, помещения, улицы, и т.п. Наше исследование показало, что результаты подобных тестирований не могут быть непосредственно применены к нашим данным. Кроме того, существует задача создания эффективного ТМО именно для ювелирных изделий. Например, для создания красивых рекламных изображений.

Сбор оценок осуществлялся с помощью системы GMLePublish [10]. Был выбран ряд ТМО, наиболее подходящих с точки зрения авторов исследования для выбранного класса примеров. Целью тестирования был выбор одного или нескольких ТМО, наиболее подходящих для поставленной задачи. В работе приводится подробное описание различных аспектов исследования и анализ полученных данных.

Примеры для оценки

В качестве примеров использовались фотографии образцов ювелирной промышленности, имеющиеся в распоряжении в лаборатории. Среди образцов присутствовали обработанные алмазы различных огранок и готовые ювелирные изделия. Далее по тексту для краткости вместо «огранённый алмаз» мы будем говорить «камень». Кроме того, в качестве ювелирных изделий выступали только кольца, поэтому далее по тексту вместо «ювелирное изделие» будем говорить «кольцо». Всего было выбрано 20 примеров – 14 камней различных оттенков и 6 золотых колец, инкрустированных маленькими камнями.

Для каждого примера было получено несколько фотографий с различными экспозициями – по 7 фотографий для камней и по 5 фотографий для колец. Шаг между экспозициями составлял 2EV. Съёмка велась с помощью цифровой камеры Canon EOS 5D Mark III. На рисунке ниже три из пяти экспозиций для одного из примеров с кольцом:

Алгоритмы тональной компрессии

В опросе были представлены результаты 8 существующих на данный момент ТМО:

- Best LDR – вручную выбранная экспозиция (без ТМО)
- Ashikhmin – [3]: “A tone mapping algorithm for high contrast images”
- ExpFusion – [8]: “Exposure fusion”
- ExpFusion (L=7) – [8]: “Exposure fusion”, меньшее количество уровней в пирамиде
- Photographic – [2]: “Photographic tone reproduction for digital images”
- Canon (ArtStandard) – встроенный в камеру ТМО
- Canon (Natural) – встроенный в камеру ТМО
- WLS – [9]: “Edge-Preserving Decompositions for Multi-Scale Tone and Detail Manipulation”

Экспозиция для Best LDR выбиралась вручную оператором при съёмке данных для опроса. Этот “псевдо”-ТМО представляет собой одну из исходных фотографий без какой-либо обработки. Два пункта с именем ExpFusion представляют разные настройки одного и того же метода, который по факту не является ТМО. Этот метод получает итоговое изображение работая напрямую с исходными изображениями, без построения HDR-изображения.

Два пункта с именем Canon являются результатами, полученными с помощью проприетарного алгоритма, реализованного в цифровой камере Canon EOS 5D Mark III – разные установки задавались в настройках камеры. Ранее в научных работах не проводилось тестирование ТМО, встроенных в потребительские устройства, такие как цифровые камеры.

На рисунке выше приведены результаты работы шести ТМО для примера с кольцом – все, кроме Best LDR и ExpFusion (L=7).

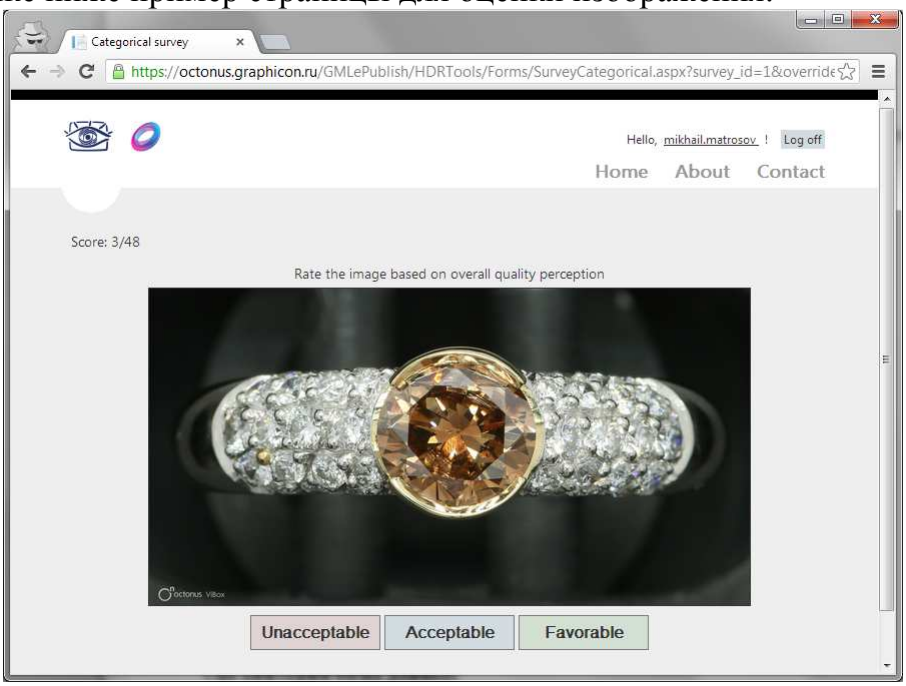
Методология

Тестирование проводилось по методологии, описанной в [7], где она использовалась для предварительного ранжирования ряда ТМО. Наша цель была аналогичной – нужно было отбросить ТМО, дающие заведомо неприемлемые результаты на наших данных. В описании системы GMLePublish [10] эта методология упоминается как «опрос с одиночными оценками».

Участнику последовательно показывались изображения с результатами работы различных ТМО на различных примерах. Каждому изображению участник должен был присвоить одну из трёх оценок, основываясь на правдоподобности изображения:

- Unacceptable – Неудовлетворительно
- Acceptable – Удовлетворительно
- Favorable – Хорошо

На рисунке ниже пример страницы для оценки изображения:



Участники

В тестировании участвовало 7 сотрудников лаборатории. Все участники являются специалистами в области обработки изображений и ранее работали с изображениями алмазов и ювелирной продукции. Т.к. всего в тестирование было включено 20 примеров и 8 ТМО, каждому участнику нужно было оценить 160 изображений. Среднее время оценки одного изображения составило 5.4 секунды, т.е. среднее время на прохождение опроса не превысило 15 минут.

Перед прохождением зачётного опроса, участнику предлагалось пройти тренировочный опрос. В нём участвовали те же самые ТМО, но было меньшее число примеров – 6 вместо 20. Тренировочный опрос нужен был для того, чтобы участник мог оценить примерный вид результатов и откалибровать свою систему оценок.

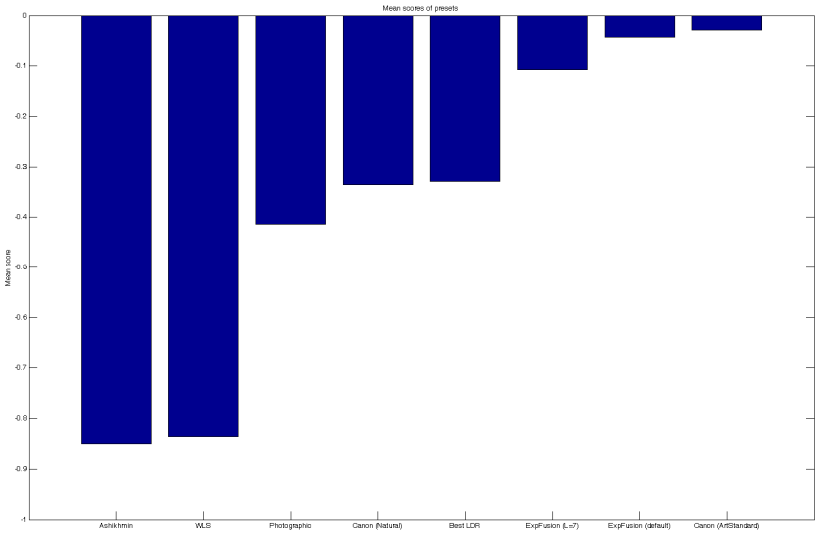
Опрос проводился с помощью системы GMLePublish. Перед прохождением опроса участник попадал на страницу с подробными инструкциями. Все участники

проходили опрос на своём рабочем месте, т.е. условия освещённости были для всех немного разными. Все мониторы были предварительно откалиброваны. Все участники имели нормальное или корректированное до нормального зрение. Все ответы участников были сохранены в системе и затем проанализированы вручную.

Результаты

Результаты опроса представляют собой трёхмерный массив оценок 8x20x7. Измерениями являются ТМО, примеры и участники. Значениями являются числа -1, 0 и 1, где -1 ставится в соответствие оценке Unacceptable, 0 – оценке Acceptable, +1 – оценке Favorable.

Полученный массив суммировался и усреднялся вдоль разных направлений, чтобы оценить результаты с разных точек зрения. На рисунке ниже средняя оценка вдоль примеров и участников для разных ТМО:

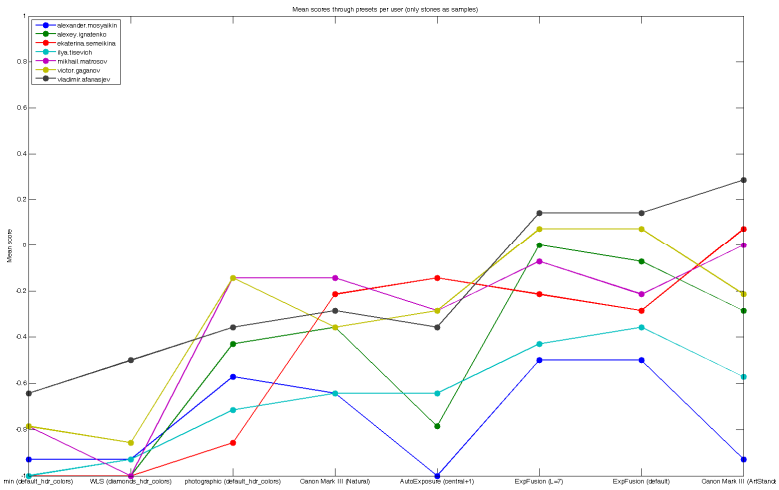


По оси абсцисс отложены различные ТМО. Они отсортированы по возрастанию средней оценки. Средние оценки всех ТМО отрицательные, поэтому график расположен ниже оси абсцисс. Из графика видно, что все ТМО можно условно разделить на три группы:

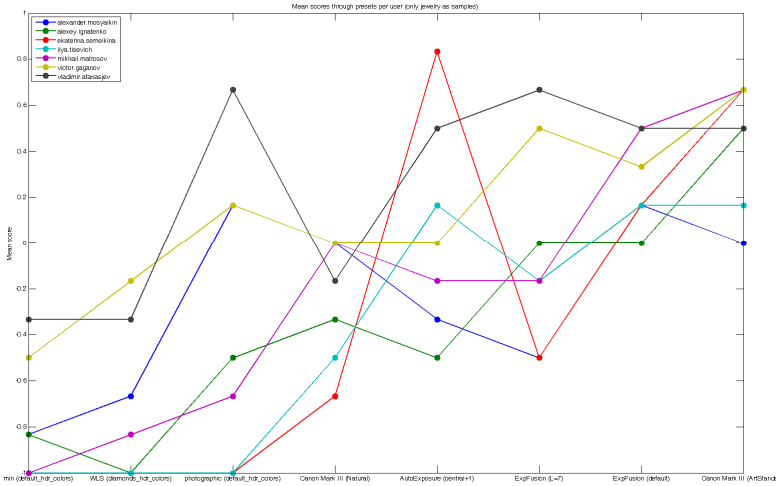
- Фавориты: Canon (ArtStandard), ExpFusion (оба)
- Средние: Photographic, Canon (Natural), Best LDR
- Отстающие: Ashikhmin, WLS

Следует заметить, что в существующих субъективных тестированиях ТМО Photographic и Ahikhmin получили высокие оценки и были в лидерах. Это объясняется спецификой наших данных.

На рисунках ниже средние оценки вдоль примеров для каждого участника в зависимости от ТМО. Усреднение только среди примеров с камнями:



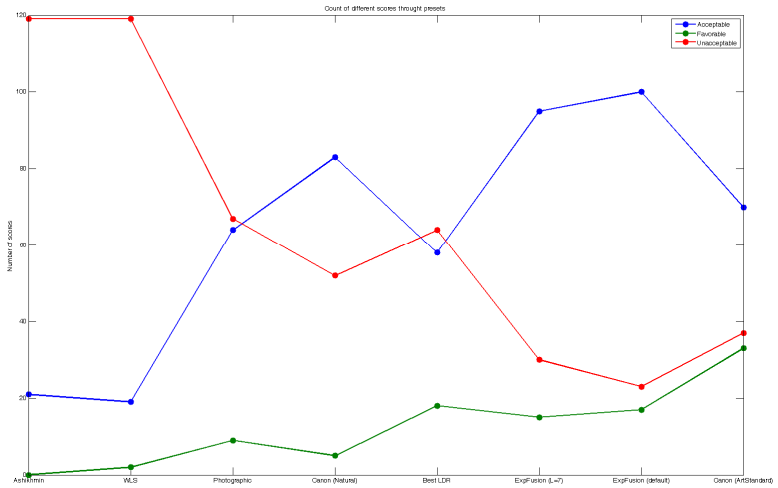
Усреднение только среди примеров с кольцами:



На основании этих графиков можно сделать следующие выводы:

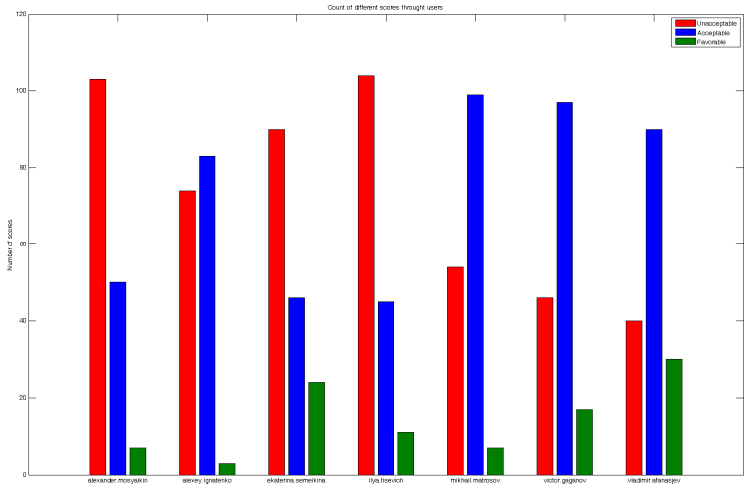
- Canon (ArtStandard) лучше работает для колец, чем для камней
- Самые консистентные оценки у ExpFusion (default) и Canon (Natural)
- Оценки разных настроек ExpFusion на камнях почти не отличаются, что согласуется с тем фактом, что их результаты на этих изображениях абсолютно одинаковы
- Ответы участников для камней более консистентны, чем для колец

Система соответствий оценок числам -1, 0 и +1 спорна, она была выбрана по аналогии с оригинальным исследованием из [7]. На рисунке ниже приведено общее количество различных оценок вдоль примеров и участников в зависимости от TMO:



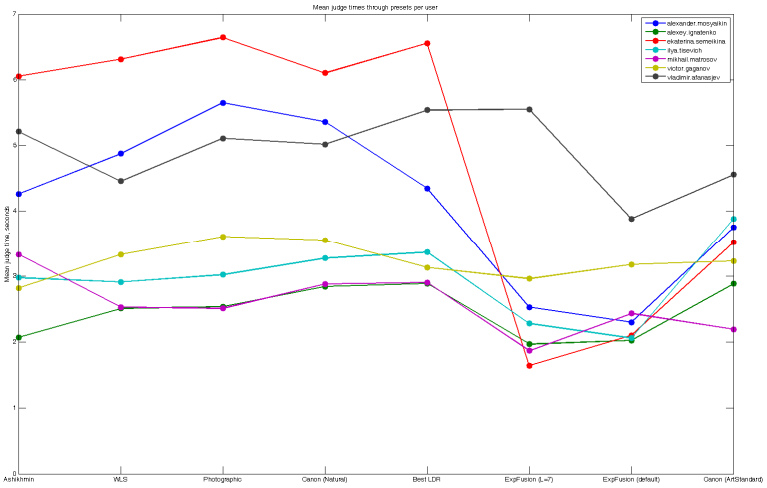
На графике видно, что большая часть оценок – Acceptable и Unacceptable. Меньшая консистентность оценок для Canon (ArtStandard) в сравнении с ExpFusion проявляется в большем количестве оценок Unacceptable и меньшем – Acceptable.

На рисунке ниже приведено количество различных оценок вдоль ТМО и примеров в зависимости от участников:



На графике видно, что количество выставленных участником оценок Acceptable и Unacceptable отличается не более, чем примерно в два раза, и что оценок Favorable всегда меньшинство.

Система сбора оценок запоминала также время, которое потребовалось участнику для оценки определённого изображения. На рисунке ниже среднее время выставления оценки вдоль сцен для участников в зависимости от ТМО:



Для составления графика значения больше 10 секунд были заменены на среднее время по данному участнику. На приведённом графике видно, что для оценки более правдоподобных результатов в среднем потребовалось меньше времени.

Заключение

Проведённое тестирование показало, что для изображений с обработанными алмазами и ювелирной продукцией фаворитами из существующих ТМО являются Canon (ArtStandard) и ExpFusion. Однако ТМО от Canon обладает рядом минусов, в сравнении с ExpFusion:

- Проприетарная разработка – нет возможности анализа и изменения
- Нет возможности использовать на произвольных изображениях

- Ограниченное сжатие динамического диапазона, т.к. выполняется только три снимка

Для проведения исследования была использована система GMLePublish, которая показала хорошую применимость для поставленной задачи и высокую степень надёжности.

Проведённое исследование имело своей целью выделить перспективные ТМО и исключить заведомо плохие. Эта цель была достигнута. Для более точного сравнения новых ТМО в дальнейшем будут использоваться другие методологии, например, опрос с попарным сравнением результатов.

В настоящий момент ведётся работа по созданию автоматизированных объективных метрик, для анализа качества которых будет проводиться сравнение их значений с полученными субъективными оценками. На основе объективных метрик планируется создание новых алгоритмов тональной компрессии, наиболее подходящих для изображений с образцами ювелирной продукции.

Список литературы

[1] Erik Reinhard, Greg Ward, Sumanta Pattanaik, and Paul Debevec, *High Dynamic Range Imaging: Acquisition, Display and Image-Based Lighting*, Morgan Kaufmann Publishers, Dec. 2005.

[2] E. Reinhard, M. Stark, P. Shirley, and J. Ferwerda, “Photographic tone reproduction for digital images,” *ACM Transactions on Graphics (TOG)*, vol. 21, no. 3, pp. 267–276, 2002.

[3] M. Ashikhmin, “A tone mapping algorithm for high contrast images,” in *Proceedings of the 13th Eurographics workshop on Rendering*. Eurographics Association, 2002, pp. 145–156.

[4] T.O. Aydin, R. Mantiuk, K. Myszkowski, and H.P. Seidel, “Dynamic range independent image quality assessment,” in *ACM Transactions on Graphics (TOG)*. ACM, 2008, vol. 27, p. 69.

[5] M. Čadík, M. Wimmer, L. Neumann, and A. Artusi, “Evaluation of hdr tone mapping methods using essential perceptual attributes,” *Computers & Graphics*, vol. 32, no. 3, pp. 330–349, 2008.

[6] P. Ledda, A. Chalmers, T. Troscianko, and H. Seetzen, “Evaluation of tone mapping operators using a high dynamic range display,” *ACM Transactions on Graphics (TOG)*, vol. 24, no. 3, pp. 640–648, 2005.

[7] M. Kuhna, M. Nuutinen, and P. Oittinen, “Method for evaluating tone mapping operators for natural high dynamic range images,” in *IS&T/SPIE Electronic Imaging*. International Society for Optics and Photonics, 2011, pp. 78760O–78760O.

[8] T. Mertens, J. Kautz, and F. Van Reeth, “Exposure fusion: A simple and practical alternative to high dynamic range photography,” in *Computer Graphics Forum*. Wiley Online Library, 2009, vol. 28, pp. 161–171.

[9] Farbman, Zeev, Raanan Fattal, Dani Lischinski, and Richard Szeliski. "Edge-preserving decompositions for multi-scale tone and detail manipulation." In *ACM Transactions on Graphics (TOG)*, vol. 27, no. 3, p. 67. ACM, 2008.

[10] М.А. Матросов, А.В. Игнатенко, «GMLePublish: web-система оценки алгоритмов тональной компрессии HDR-изображений», Сборник трудов научно-практического семинара «Новые информационные технологии в автоматизированных системах-16». М.: ИПМ, 2013 г. сс. 37-42

Пересегментация изображений для применения в задаче трекинга объектов, меняющих форму

К.С. Зипа, С.С.Жулин, А.А. Ильин
МГУ им. Ломоносова, ф-т ВМК, кафедра АСВК, лаборатория КГиМ
kzipa@graphics.cs.msu.ru

Аннотация. В статье представлен алгоритм пересегментации изображений, для которых известна предварительная разметка объект-фон. На основе предварительной разметки строится ее огрубление (овыпукление). Степень огрубления задается параметрически. Полученные сегментации могут использоваться в задаче трекинга объектов, меняющих форму, исчезающих и появляющихся вновь.

Ключевые слова: сегментация, морфология, задача трекинга, дистантное преобразование, компактность.

Введение

Задача трекинга подразумевает определение местоположения, траектории и характеристик движущихся объектов в видеопотоке. Основная проблема трекинга – сопоставление положений целевого объекта на последовательности кадров. В зависимости от специфики искомых объектов вводят для них модель движения. Например, можно отслеживать плоский объект, или же его трехмерную модель. В процессе отслеживания могут возникать различные проблемы: перекрытие объекта, временное исчезновение из области видимости, сильная деформация двумерной проекции объекта. Особенно трудно отследить и локализовать те объекты, о которых имеется мало априорной информации, например, когда не известна приблизительная трехмерная форма объекта, когда форма может сильно меняться, когда объект – это некоторое явление, не имеющее простых физических характеристик (например, оптические эффекты и иллюзии).

Один из подходов – использование алгоритмов сегментации для разграничения изображения на объект-фон. При их использовании может возникать проблема нестабильности разметок от кадра к кадру, это может происходить как из-за ошибок сегментации, так и из-за нестабильности самих данных (резкие скачки формы). В данной работе рассматривается метод пересегментации с огрублением формы до заданного значения. Этот метод позволяет снизить ошибки трекинга для объектов, меняющих форму.

Постановка задачи

В первую очередь опишем модель движения в задаче трекинга. На видео присутствует неизвестное число объектов, которые в процессе движения могут незначительно менять очертания и перемещаться. Также объекты могут исчезать в любом кадра или неожиданно возникать. Известно допустимое значение на период исчезновения объекта. Также допускается перемещение объектов за кадр не больше, чем на заданное число (ограничение по скорости).

Примером может служить отслеживание объектов в инфракрасном диапазоне, в этих случаях их легко отсегментировать, но форма будет выделяться неустойчиво. В статье [1] представлены два алгоритма для отслеживания нескольких объектов в последовательности изображений инфракрасного диапазона. Первый алгоритм использует вычисление фона (background estimation) для идентификации объектов на каждом кадре. Затем между найденными объектами ищутся связи. Этот алгоритм

эффективен, но демонстрирует неустойчивость к включениям, за счет чего чувствителен к повышению сложности сцены. Альтернативный алгоритм использует алгоритм бустинга для классификации на основе сопоставления признаков объектов, что позволяет добиться стабильной работы на сложных сценах.

Другую возможность повысить устойчивость дает пересегментация. Алгоритм трекинга, в который предлагается встраивать алгоритм пересегментации содержит следующие части: первичная сегментация объекты-фон, итерационный алгоритм трекинга траектории объекта с по-объектной пересегментацией. Это значит, что объекты ищутся последовательно в некотором блоке видео. Поиск каждого объекта сопряжен с частичной пересегментацией, повышающей устойчивость трекинга.

Цель пересегментации – добиться устранения искажений первичных сегментаций, абстрагироваться от покадровых деталей объекта, чтобы более точно связать его с проекциями на других кадрах. При этом требуется по возможности избежать ошибок объединения различных объектов на одном кадре.

Обзор существующих алгоритмов

В настоящее время существует довольно много алгоритмов автоматической и полуавтоматической сегментации изображений. Подробное описание базовых направлений можно найти в обзоре [2].

В статье «Алгоритмы детектирования разметки и дефектов дорожного покрытия» [3] рассматривается алгоритм выделения контура дефекта, в основе которого лежит метод активных контуров (“змея”-snake) [4]. На входе имеются только координаты какой-либо внутренней точки объекта. Поиск контура состоит из следующих этапов:

1) Построение инициализатора (начального приближения) для активного контура в виде эллипса с центром в заданной точке. Предполагается, что область однородна по цвету (рассматривается дисперсия интенсивности).

2) Получение результата метода активных контуров (основан на минимизации функционала энергии). В качестве поля внешних сил используется поток вектора градиента, вычисляемый по карте границ.

3) Постобработка: коррекция результата методом минимального разреза графа [5].

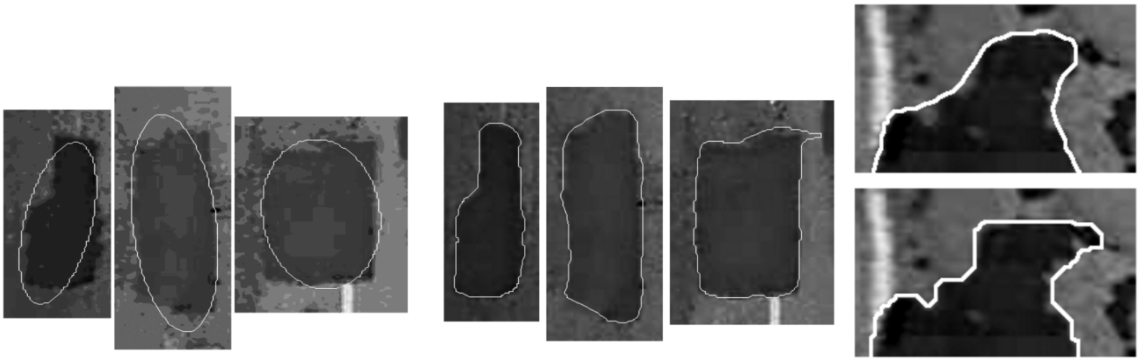


Рис. 1. Результаты работы разных этапов алгоритма выделения контура дефекта

Особенность алгоритма в том, что он позволяет находить точные границы, основываясь только на информации об внутренней точке. Но в задаче трекинга объектов меняющих форму точная граница не требуется, поскольку при сопоставлении она только затруднит идентификацию одного и того же объекта. Тем не менее, в этом алгоритме присутствует интересная идея: описание сложной формы с помощью более простой геометрической фигуры. Излишняя степень упрощения

тоже может навредить при покадровом сопоставлении объектов, поэтому в этой работе предлагается использовать описание с задаваемой степенью абстракции.

Метод разрастания регионов (region growing) [6] работает снизу-вверх (bottom-up). Вначале определяется пиксель-семя (seed pixel) происходит его сравнение с соседями. Похожие соседи добавляются к семени (регион разрастается). После объединения всех похожих пикселей выбирается новый, не принадлежащий региону, и процесс повторяется до исчерпания всех пикселей. В качестве метрики похожести может выступать, например, разница интенсивностей. Метод разрастания регионов дает хорошие результаты для изображений с четкими краями. Однако преимущество при этом получает регион, сегментировавшийся первым. Поэтому разный выбор начальных семян дает разные итоговые результаты. Для преодоления этого недостатка были предложены методы одновременного разрастания регионов, в которых несколько регионов разрастаются одновременно.

Алгоритм, о котором рассказывается в этой работе, базируется на методе разрастания регионов. Однако у него есть несколько отличий от классического алгоритма:

- он нацелен на применение в пространстве бинарных масок (сужение области применимости)
- в качестве семени выбирается пиковое значение дистантного преобразования начальной сегментации
- вместо функционала похожести используется ограничение на компактность всего региона
- разрастание производится в соответствии с дистантным преобразованием
- разрастание на каждой итерации происходит более чем на один пиксель вдоль границы.

Нахождение компактной маски

Четкий контур объекта не всегда необходим, а зачастую и вреден в задачах трекинга изменяющихся объектов. На видео от кадра к кадру он может значительно исказиться, к тому же методы сегментации могут давать ошибки и погрешности. При этом характерная форма объекта с некоторой точностью остается постоянной, но именно этой формы и бывает достаточно для связи одного и того же объекта на различных кадрах.

Задача алгоритма нахождения компактной маски – на основе четкой маски построить приближенную маску, характеризующую форму объекта. Алгоритм неприменим в задачах классификации, только в задачах слежения за объектом. Если требуется детальная классификация, то можно, например, классифицировать по точной маске на первом кадре, а затем следить за объектом.

Алгоритм базируется на дистантном преобразовании [7] предварительной четкой маски и использует идеи метода разрастания регионов. С его помощью можно находить несколько объектов на изображении и четко их разграничивать. У алгоритма есть один параметр – порог максимальной компактности. Компактность (C) понимается как отношение квадрата периметра (P) к площади объекта (S):

$$C = \frac{P^2}{S}$$

При работе с растровыми изображениями компактность сильно зависит от того, как считать периметр. Для четырехсвязной границы и внутреннего периметра значения компактности для разных объектов представлены на рисунке 2.

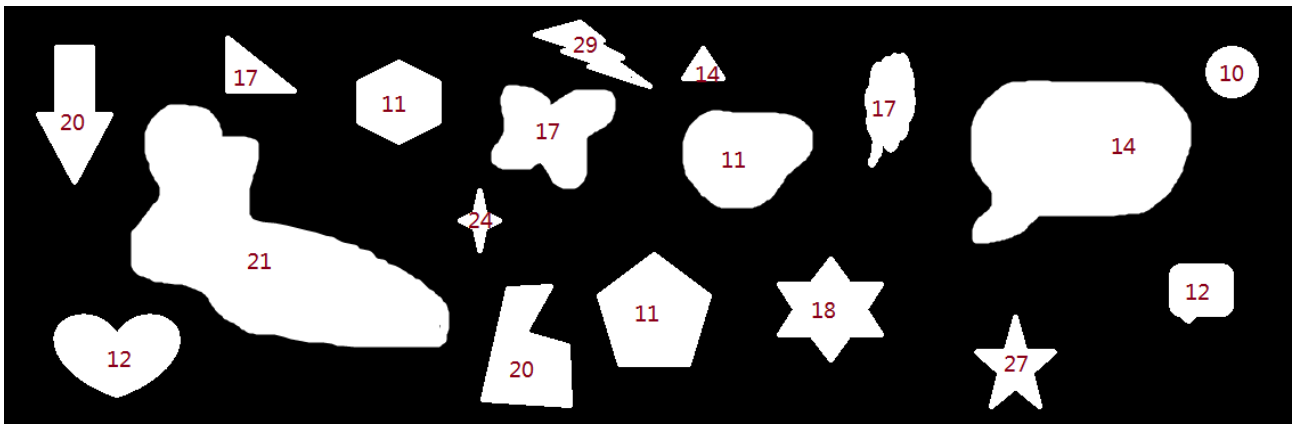


Рис. 2. Пример растровых объектов и их значений компактности

Перейдем к непосредственному описанию алгоритма. На входе – предварительная четкая маска объект\фон. На выходе – сегментация, лежащая внутри предварительной.

1 Для предварительной маски строится евклидово дистантное преобразование `dmap`.
2 В соответствии с алгоритмом трекинга находится инициализирующее семя, являющееся локальным пиком на дистантном преобразовании.
3 В этом пике строится начальный объект – окружность с радиусом `R = dmap(x0,y0)`.
4 Далее этот объект итерационно разрастается, пока не будет достигнут предел компактности, или достигнуты границы исходной маски.

Описание шага роста:

Ищется среднее и максимальное значение (`medV` и `maxV`) дистантного преобразования по всем точкам текущей границы объекта. Каждая точка границы расширяется пропорционально своему значению в дистантном преобразовании и значению среднего: `d = dmap(i,j)*medV/maxV`.

5 Происходит присвоение отдельной метки всем пикселям объекта.
6 Занятая объектом область исключается из рассмотрения в алгоритме трекинга.

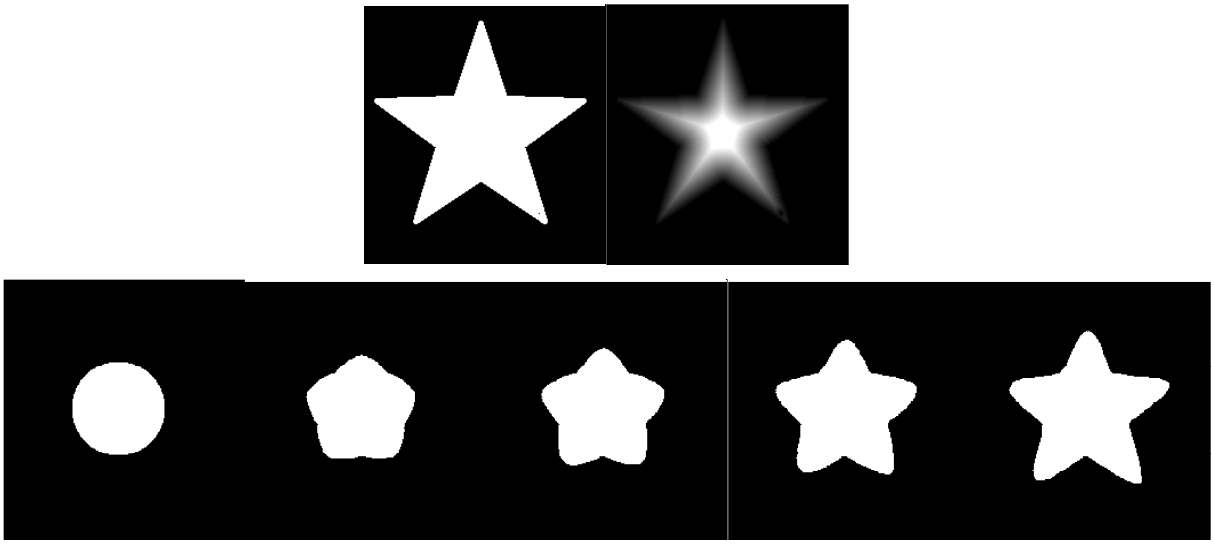


Рис. 3. Пример работы на сегментации в виде звезды. Шаг 1,2,4,7. (Значения компактности на каждом шаге 11,13,17,20)

Алгоритм «умная морфология»

Огрубляет контуры четкой маски при сохранении формы, препятствует нежелательному соединению различных разметок, строит огибающую с внешней стороны (разрастает исходную маску). Может применяться вкупе с методом нахождения неточной маски в тех случаях, когда гарантированно отсутствие внутренней фрагментации. Если возможно подразбиение четкой маски, то его использовать не рекомендуется. Умная морфология уничтожает внутренние неоднородности и помогает бороться с шумом. Параметр алгоритма – число шагов разрастания (степень огрубления) `NSTEPS`.

Описание работы: итерационно присоединяет к маски соседнюю область (как в стандартном методе разрастания регионов).

```
for j=1:NSTEPS
    F1 = imfilter(bframe,[1 1 1]);
    F2 = imfilter(bframe,[1; 1; 1]);
    F3 = imfilter(bframe,[1 0 0; 0 1 0; 0 0 1]);
    F4 = imfilter(bframe,[0 0 1; 0 1 0; 1 0 0]);
    bframe = F1&F2&F3&F4;
end
```

В пределе (`NSTEPS` = ∞) превращает объект в его огибающую прямыми линиями и наклонными линиями (и далее не разрастается).

Рис. 4. Предельная форма для силуэта бабочки

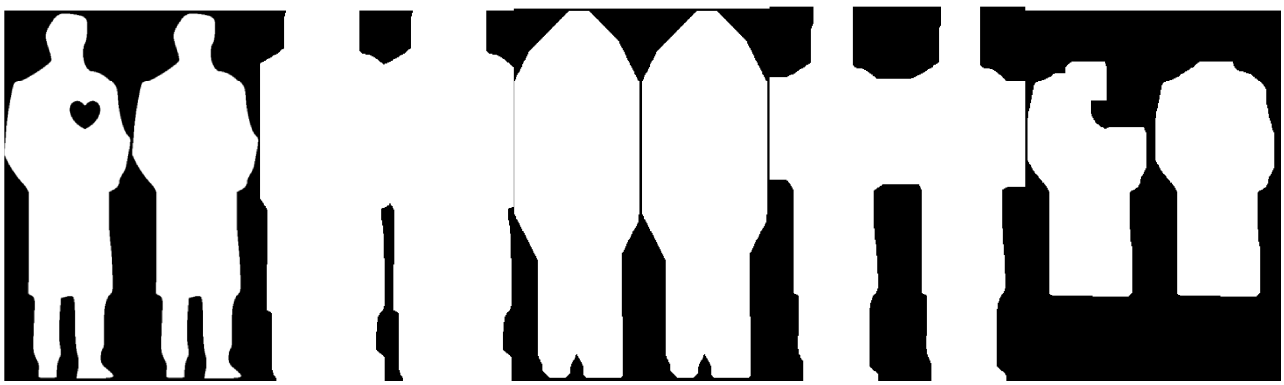


Рис. 5. Пример сегментации с двумя силуэтами людей. Обычное расширение (R = 75). Умное расширение (150 шагов), закрытие (R = 75), открытие (R = 75)

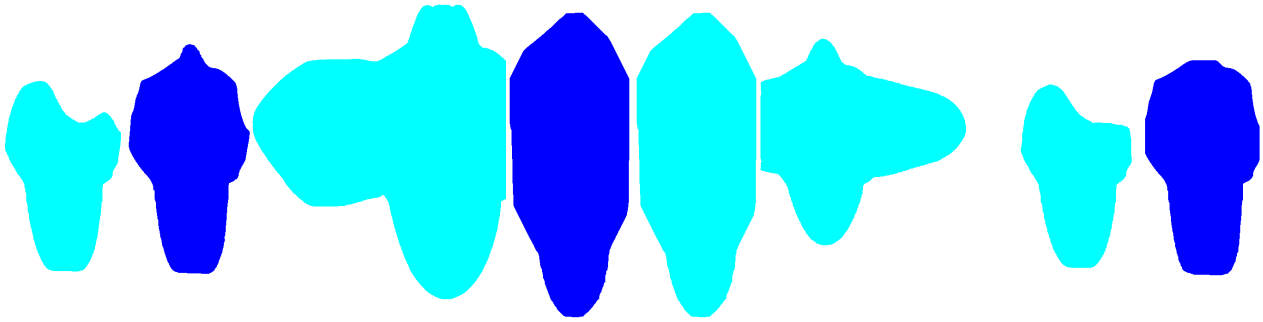


Рис. 6. Результат разметки без морфологии, с обычным расширением, с “умным” расширением, с закрытием, с открытием

Пример использования приведенных алгоритмов в задаче трекинга

Теперь рассмотрим более подробно, как встроить алгоритм пересегментации в имеющийся алгоритм трекинга. Рассмотрим случай постобработки видео. Это значит, что имеется буфер информации с кадрами от 1 до N. Про объекты нам известны ограничения на их относительные и абсолютные перемещения, также известен минимальный размер искомого объекта. Известно, что у объекта довольно хорошая компактность. Подходящие объекты: человек, машина, собака, самолет, футбольный мяч. Не подходящие: змея, пожарный шланг, ленточка гимнастки.

1. Найдем первичные независимые предварительные сегментации для каждого кадра. Если позволяют дополнительные ограничения, применим умную морфологию.
2. Построим дистантное преобразование для каждой предварительной сегментации.
3. Найдем на дистантных преобразованиях локальные пики.
4. Выберем максимальный локальный пик (кадр K) и будем строить цепочки из локальных пиков в направлении кадра 1 и в направлении кадра N. Два локальных пика соединяются в цепочку при минимизации функционала близости. Функционал близости задается для двух точек в пространстве XYT, может использовать информацию о расстоянии, интенсивности, градиенте и т.д.
5. Для каждого узла цепочки и предварительных сегментаций выполним операцию нахождения компактной маски.
6. Удалим из предварительных сегментаций найденные объекты. Пересчитаем дистантное преобразование.

7 Повторим процесс до тех пор, пока не найдем все объекты, большие наперед заданной величины, или не превысим максимальное количество объектов.

Заключение

Представленный алгоритм пересегментации может применяться в задаче трекинга объектов, он позволяет повысить устойчивость. Приведенный алгоритм трекинга не является универсальным, а только служит для демонстрации возможностей алгоритма пересегментации. В частности, можно использовать алгоритмы трекинга, учитывающие примерную форму объекта, и использовать при построении цепочки информацию о потенциальной форме, порождаемой локальным пиком (сравнивать новую форму с текущей).

Список литературы

1. M. Calafut. Multiple-Object Tracking in the Infrared. Stanford University.
2. В. Конушин, В. Вежневец. Методы сегментации изображений: интерактивная сегментация. Компьютерная графика и мультимедиа. Выпуск №5(1)/2007.
3. С. Судаков, А. Семашко, О. Баринова, А. Конушин, В. Киншаков, А. Крылов. Алгоритмы детектирования разметки и дефектов дорожного покрытия. 2008. Графикон’2008, стр. 206-212
4. M. Kass, A. Witkin and D. Terzopoulos “Snakes: Active Contour Models” International Journal of Computer Vision, vol. 1, pp. 321-331, 1987
5. E.V. Zaugolnova, D.V. Yurin “Algorithm for Refinement of Preliminary Segmentation of Images with Smooth, Low Contrast 2D Objects Boundaries” In proc. Graphicon, 2006
6. R.M. Haralick and L.G. Shapiro. Survey: Image. Segmentation. Techniques. CVGIP, vol. 29, 1985, pp. 100-132
7. http://en.wikipedia.org/wiki/Distance_transform

Приложение

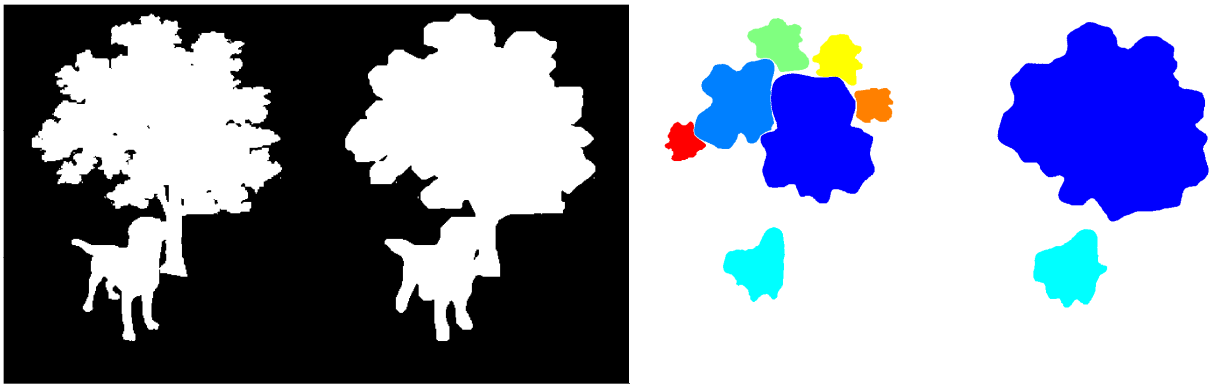


Рис. 7. Пример работы алгоритма для разложения на компоненты. Исходное изображение, умное расширение. Результат разметки с ограничением на компактность 15

Доказательство подлинности фотоизображений встраиванием цифровых водяных знаков

Белобокова Ю.А.
МГУП имени Ивана Федорова, Москва, Россия
Клышинский Э.С.
МИЭМ НИУ ВШЭ, Москва, Россия

Аннотация. В статье ставится задача создания метода для противодействия подделкам цифровых фотографий, описывается алгоритм защиты изображения.

Встраивание в цифровые фотографии невидимых меток, в качестве которых могут выступать последовательности символов или графические изображения, является одним из распространённых способов защиты информационного содержания этих фотографий.

Такие метки были названы цифровыми водяными знаками (ЦВЗ) по аналогии с широко известным способом защиты ценных бумаг от подделок. Метод защиты графической информации с помощью ЦВЗ является составной частью цифровой стеганографии, науки о незаметном сокрытии одних данных в других.

Рассмотрим ситуацию, когда защиту цифровой фотографии производят с целью доказательства её незаконного использования или возможной фальсификации. В этом случае исходное изображение может быть подвергнуто ряду внешних воздействий (атак), к которым относятся зашумление, обрезка краёв изображения, кадрирование, фильтрация, повороты, смена цифрового формата, масштабирование и сжатие с потерями. Также какие-либо фрагменты защищаемой фотографии могут быть добавлены или, наоборот, удалены. Часть атак, таких, как зашумление или фильтрация, направлена непосредственно на извлечение или повреждение возможных встроенных ЦВЗ. Другие воздействия, например, кадрирование, масштабирование, направлены на подготовку изображения к его коммерческому использованию.

Внедряемый в защищаемую фотографию ЦВЗ должен отвечать двум противоречащим друг другу критериям: робастности (устойчивости к различным внешним воздействиям) и скрытности (обеспечения наименьших искажений изображения по сравнению с оригиналом) [1]. Для проверки авторских прав на цифровое изображение осуществляют извлечение встроенной информации, которое при недостаточной устойчивости ЦВЗ к атакам может стать невозможным.

Целью дальнейшей работы является разработка метода, позволяющего гарантировать устойчивость внедрённых в изображение элементов к зашумлению, фильтрации, геометрическим атакам и смене цифрового формата изображения, а также указывать на изменение различных его фрагментов. Предполагается, что при попытке замены нарушителем фрагментов промаркированного изображения другими, после процедуры извлечения ЦВЗ инородные объекты станут заметны, что облегчает доказательство подделки цифрового изображения.

Из существующих форматов цифровых изображений для сохранения полноцветных фотографий хорошо подходят форматы JPEG и JPEG2000, поскольку сохраненные в этих форматах изображения имеют высокие показатели качества при сравнительно небольших размерах файла. Недостаток данных форматов заключается в том, что для уменьшения размера графических файлов применяется алгоритм их сжатия с потерями, а значит, часть информации при сохранении таких изображений

будет необратимо утеряна, и, следовательно, существует вероятность искажения или потери встроенных ЦВЗ.

Для повышения робастности вложения встроенных ЦВЗ к сжатию или масштабированию графических файлов в стегоалгоритмах стараются применять те же преобразования, что и в алгоритмах сжатия этих файлов [3]. Для формата JPEG таким алгоритмом является дискретное косинусное преобразование, для JPEG2000 – вейвлет-преобразование.

Графический формат JPEG2000 имеет ряд преимуществ по сравнению с форматом JPEG. Это лучшее качество изображения при равной степени сжатия, оптимизация качества кодирования, а также возможность сжатия без потерь. Поэтому при выборе формата сохранения цифровой фотографии лучше отдать предпочтение более современному JPEG2000. Разрабатываемый алгоритм встраивания информации должен быть основан на вейвлет-преобразовании, так как в этом случае ЦВЗ будет устойчив к сжатию изображения.

Вообще, для устойчивости ЦВЗ к различным искажениям фотографии его необходимо многократно внедрять в неё, а значит, размеры встраиваемых меток должны быть значительно меньше размеров защищаемого изображения.

В статье [5] разбирался существующий метод встраивания ЦВЗ в фотоизображение в формате JPEG2000 с помощью вейвлет-преобразования маркировкой наиболее существенных участков данного изображения с учётом особенностей человеческого восприятия. Такой метод достаточно эффективен для доказательства подлинности цифровой фотографии, поскольку встроенную таким образом информацию сложно повредить или удалить, не ухудшив визуальных качеств самого изображения. Однако данный метод не вполне подходит для доказательства факта обрезки исходного изображения, а также удаления или замены его значащих частей.

При разработке нового метода необходимо учесть плюсы упомянутого выше существующего, т.е. встраиваемый ЦВЗ должен быть устойчив к зашумлению изображения, его фильтрации и сжатию с потерями. В то же время маркировка областей изображения должна производиться таким образом, чтобы при процедуре извлечения ЦВЗ для проверки подлинности изображения можно было понять, производилась ли обрезка этого изображения, а также были ли удалены или заменены его фрагменты.

Для достижения поставленной цели можно использовать следующий алгоритм.

1. Исходное изображение, которое можно представить как матрицу $f(n,m)$ разбивается на одинаковые блоки a_{ij} размером 64 x 64 пикселя (рис. 1).

2. Выбирается алгоритм расположения ЦВЗ, являющихся матрицей $w(m, n)$ размером 4 x 4 пикселя, в этих блоках. Целесообразно встраивать в каждый блок по 4 ЦВЗ, отступая от края не менее чем на 3 пикселя, но располагая их несимметрично относительно этого блока (рис. 2).

3. ЦВЗ встраиваются в каждый блок изображения согласно выбранному алгоритму.

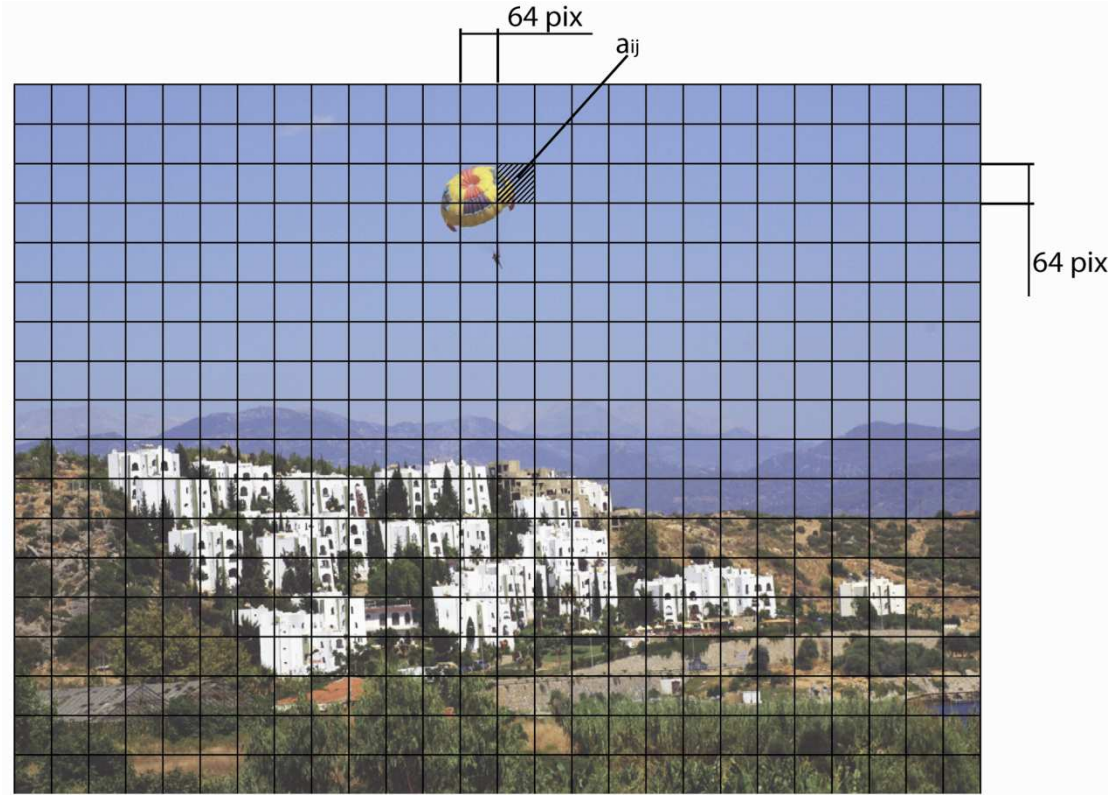


Рис. 1. Изображение, разбитое на блоки

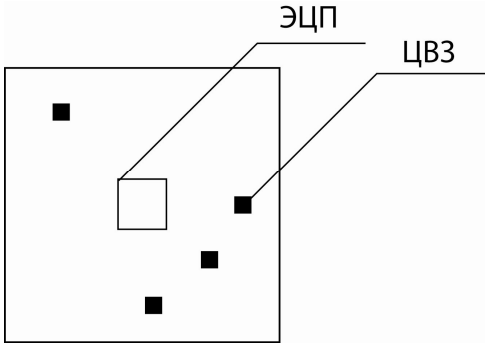


Рис. 2. Пример встраивания ЦВЗ и ЭЦП в блок изображения

Следующие шаги разрабатываемого алгоритма защиты фотоизображения связаны со встраиванием в изображение ЦВЗ, выполняющего роль электронно-цифровой подписи (ЭЦП). В отличие от остальных ЦВЗ, встраиваемые в блоки ЭЦП должны быть различными и характеризовать такие свойства изображения, как, например, его яркость, контраст, фильтрация по цвету и НЧ-фильтрация, а также информация о всём изображении в целом (размер изображения, количество блоков в нём) и соседних с ним блоков.

Каждую ЭЦП необходимо встраивать чётко посередине каждого блока, не пытаясь выбрать наиболее удачные для робастности участки. ЦВЗ, выполняющие роль ЭЦП, должны быть хрупкими. Это связано с тем, что полное или частичное отсутствие в проверяемом изображении этих центральных меток будет свидетельствовать о том, что данная фотография подвергалась каким-либо воздействиям. При этом другие встраиваемые ЦВЗ должны быть максимально робастными к атакам, чтобы можно было говорить о нарушении авторских прав на это изображение.

Отличие встраиваемых ЭЦП друг от друга необходимо для определения применения к изображению процедуры кадрирования.

Вернёмся к алгоритму.

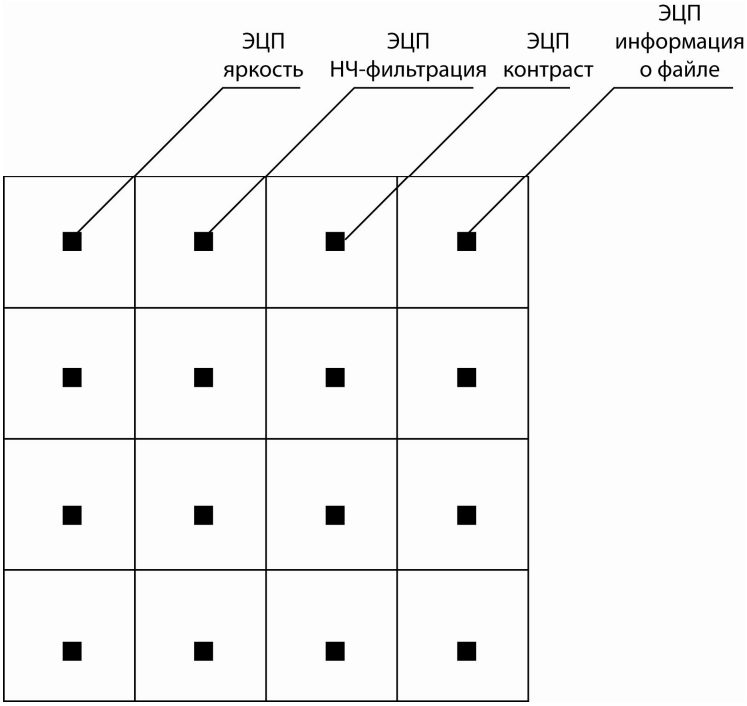


Рис. 3. Пример алгоритма нанесения ЭЦП в блок изображения

4. Выбирается алгоритм нанесения в блоки различных ЭЦП (рис. 3).

5. Составляются ЭЦП в зависимости от их назначения.

6. ЭЦП встраиваются в блоки изображения в соответствии с выбранным алгоритмом.

7. ЭЦП каждого блока корректируется в зависимости от соседних блоков.

Следующим шагом изображение подвергается обратному вейвлет-преобразованию, в результате чего оказывается промаркированным.

Для проверки подлинности изображения необходимо осуществить извлечение встроенных ЦВЗ и ЭЦП, выполняя преобразование над промаркированным изображением, которое могло быть подвергнуто различным воздействиям.

Для этого целесообразно действовать следующим образом.

1. Выполняя преобразование, находим все детектируемые ЦВЗ и проверяем регулярность структуры.

2. По меткам ЦВЗ находим ЦВЗ-ЭЦП и проверяем их регулярность.

Если расположение встроенных ЦВЗ и ЦВЗ-ЭЦП регулярно и соответствует алгоритму их расположения в блоках, можно сделать вывод, что это изображение не подвергалось внешним атакам.

Если при анализе вложений в изображение мы столкнулись с нерегулярностью расположения ЦВЗ или ЭЦП, их искажением, а также с наличием «дыр» в структуре расположения, отсутствия ЭЦП или сдвигов ЭЦП по срезу цифровой фотографии, можно утверждать, что это изображение подвергалось внешним воздействиям.

Например, в случае удаления каких-либо объектов с исходной фотографии на месте удаленного объекта ЦВЗ и ЭЦП детектироваться не будут (рис. 4).

При смещении какого-либо объекта на исходной фотографии при проверке подлинности может быть выявлено смещение ЭЦП, а также перекрытие ЦВЗ (рис. 5).

В случае добавления стороннего объекта при проверке подлинности фотографии на нем будет выявлено отсутствие маркировки (рис. 6).

Рис. 4. Удаление объекта. Слева исходное изображение, справа – модифицированное, красной чертой обведено место с нарушением структуры вложений, блоки разбиения изображения показаны квадратами

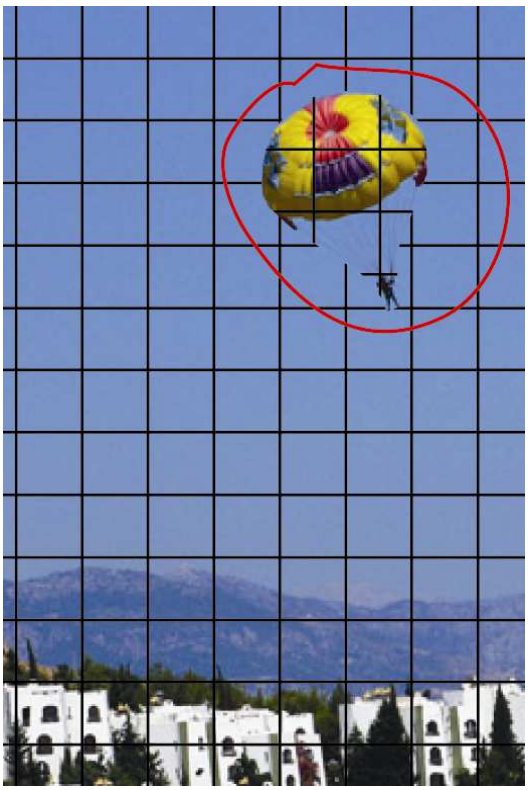


Рис. 5. Смещение объекта. Красной чертой обведено место с нарушением структуры вложений, блоки разбиения изображения показаны квадратами

В случае если извлеченные ЭЦП и ЦВЗ извлечены корректно (не деформированы), необходимо проверить, соответствует ли расположение всех ЭЦП алгоритму их встраивания. В случае обрезки промаркированного изображения они будут смещены.

В случае небольших поворотов промаркированного изображения предполагается, что хрупкие ЭЦП будут разрушены, а ЦВЗ будут детектироваться.

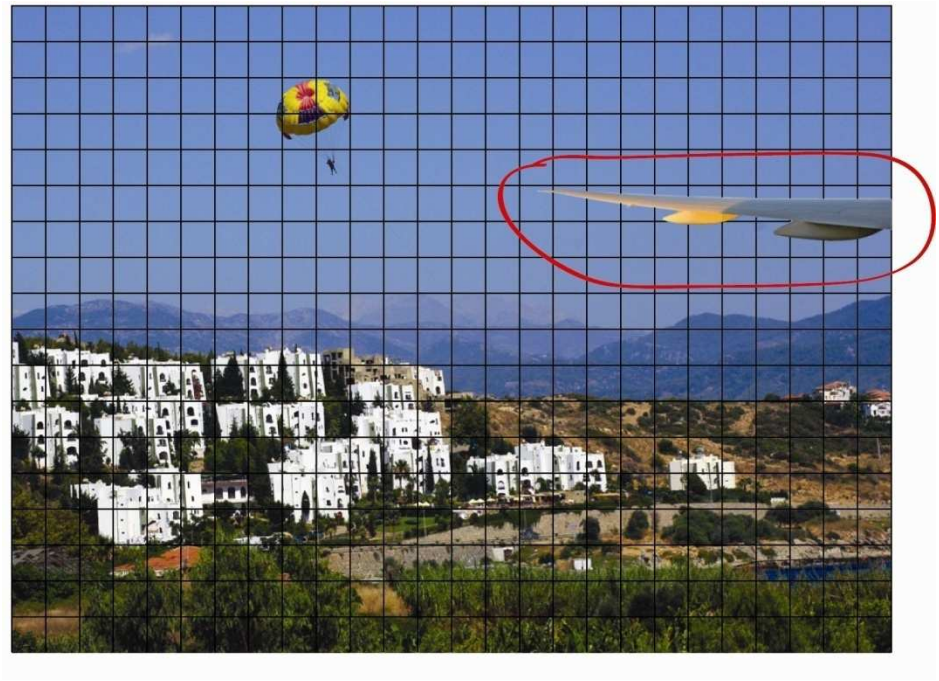


Рис. 6. Вставка стороннего объекта. Красной чертой обведено место с нарушением структуры вложений, блоки разбиения изображения показаны квадратами

Благодаря разбиению исходной цифровой фотографии на блоки, несимметричной маркировке блоков ЦВЗ и наличию в центре каждого блока ЭЦП, расположенных согласно выбранному алгоритму, при проверке подлинности изображения можно определить, подвергалось ли оно обрезке, фильтрации или другим атакам.

Список литературы

1. Цзянь Ван. Исследование устойчивости цифровых водяных знаков-логотипов, внедряемых в статические изображения. Автореферат. С.-Пб, 2010. – 20с.: ил.

2. О.В. Михайличенко. Методы и алгоритмы защиты цифровых водяных знаков при jpeg сжатии. Автореферат. С.-Пб, 2009. – 17с.: ил.

3. А.В. Аграновский, А.В. Балакин, В.Г. Грибунин, С.А. Сапожников Стеганография, цифровые водяные знаки и стеганоанализ. М.: Вузовская книга, 2009 г

4. В.Г. Грибунин. Вейвлеты в стеганографии. Статья. С.-Пб, 10с.: ил.

5. А.В. Немцов, Ю.А. Белобокова. Защита изображений встраиванием цифровых водяных знаков.

«Моделирование процессов»

Модель искусственного интеллекта на реляционных множествах

А.П. Гусев
Компьютерные системы и сети, МГТУ им. Н.Э. Баумана.
xitren@ya.ru +7 (965) 204 12 88

Аннотация. Представлена новая модель искусственного интеллекта на реляционных множествах. Для разработки этой системы рассматривались основные понятия логических операции и аксиомы теории множеств. Также представлена и описана разработанная модель. Во-первых была рассмотрена структура данных в виде реляционных множеств, правила ее организации и визуализации действий производимых для их достижения. Во-вторых представлен алгоритм генерации ответа на внешнее событие и рассмотрены модели действий составляющих предложенный алгоритм.

Ключевые слова: модель, искусственный интеллект, реляционная модель, теория множеств, логические операции.

Развитие технологий во всем мире приходит к созданию по большей части интеллектуальных устройств. Однако за развитием понятия «интеллекта» стоят сложные и зачастую эвристические алгоритмы. Они часто ошибаются и зачастую требуют значительных вычислительных ресурсов.

Для построения логических цепочек любого вида, современные системы используют логику, основанную на нескольких базовых операциях:

- Отрицание («не») ($\neg$)
- Конъюнкция («и») ($\wedge$ или &)
- Дизъюнкция («или») ($\vee$)
- Импликация («следует») ($\rightarrow$)
- Эквивалентность («равны») (=)
- Неэквивалентность («не равны») (!=)

Их достаточно для отображения любой логики принятия решения для конечного алгоритма, и возможность их выполнения берется за показатель состоятельности системы ИИ (Искусственного Интеллекта). Однако на вычисления громоздких логических выражений уходит огромное количество времени, тогда как их можно выполнить побочно с более существенными преобразованиями над данными.

Углубимся в понятие логических операций, согласно определению, это операция над высказываниями, позволяющая составлять новые высказывания путем соединения более простых[2]. Чтобы избежать неточностей в представлении логических операций, проиллюстрируем их на примере множеств (Рисунок 1).

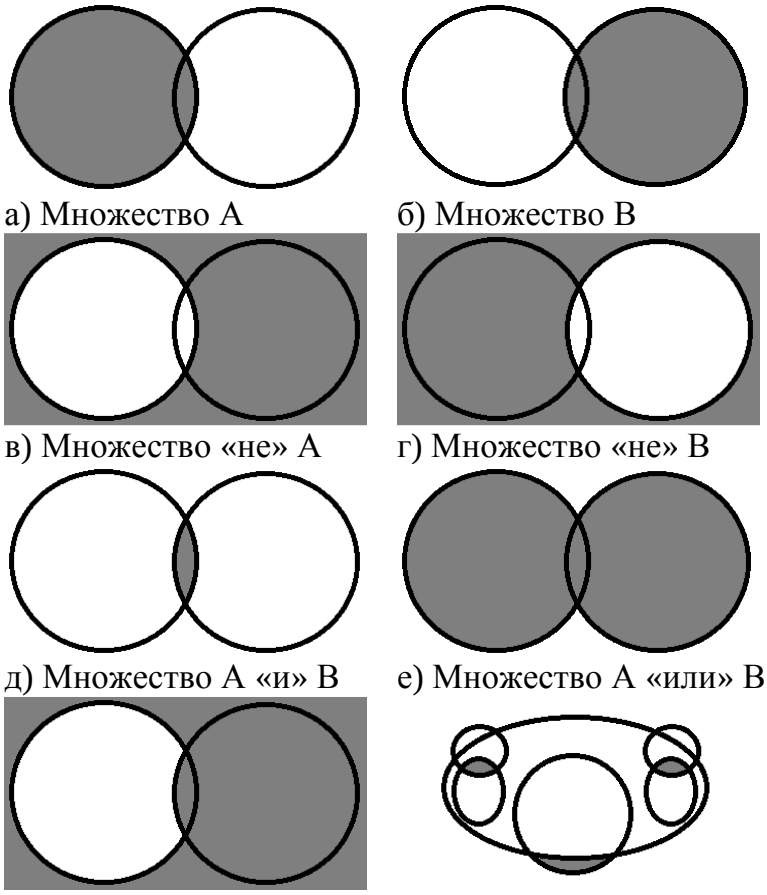


Рис 1. Наглядное изображение логических операций

Из этого простейшего примера достаточно хорошо заметно, что определенная совокупность таких операций может выделить для нас любую фигуру из образованных этими множествами. В данном случае остается лишь обеспечить, чтобы круги множеств образовали необходимую нам картину.

Чтобы максимально приблизить разрабатываемую систему к функционированию памяти человека, рассмотрим операции, выполняемые нашей памятью с точки зрения психологии.

Сравнение выделенных объектов. Чтобы отразить с помощью мышления какие-либо связи и отношения между предметами или явлениями объективного мира, необходимо, прежде всего, в восприятии или представлении выделить эти явления.[3] Например, чтобы понять причину неудачного выполнения спортсменом данного физического упражнения, необходимо сосредоточить свою мысль на этом упражнении и на тех условиях, при которых оно выполнялось. Это выделение всегда связано с осознанием задачи, оно предполагает предварительную постановку вопроса, который и определяет собой выделение интересующих нас объектов. Сравнивая явления друг с другом, мы отмечаем как сходство, так и различие их в определенных отношениях, их тождество или противоположность. Например, низкий или высокий старты сходны между собой по своему назначению, являясь начальным моментом упражнения, но различаются по положению тела спортсмена.[4] Сравнивая выделенные в процессе мышления явления, мы точнее познаем их и глубже проникаем в их своеобразие.

Абстракция. Чтобы осуществился процесс мышления, необходимо не только различать отдельные свойства предметов, но и мыслить эти свойства отвлеченно от самих предметов.[3] Такая мыслительная операция называется абстракцией (от лат. абстракций — отвлечение). Процесс абстрагирования есть мысленное (временное) отвлечение одного свойства вещи от других ее свойств, одного предмета от других предметов, с которыми он в действительности связан. Так, исследуя закономерности процесса реакции спортсмена на старте, психолог-экспериментaлист выделяет только один элемент этого процесса — латентный период, отвлекаясь (пока, на время) от таких побочных явлений, как влияние на спортсмена зрителей, его личное отношение к данному соревнованию и т. д.[4] Абстракция позволяет проникнуть «вглубь» предмета, выявить его сущность, образовав соответствующее понятие об этом предмете.

Обобщение. Абстракция всегда соединяется с обобщением; абстрагированные свойства предметов мы сейчас же начинаем мыслить в их обобщенном виде. [3]Например, разбираясь в характерных особенностях удара боксера при нокауте, мы выделяем такое его свойство, как резкость; при этом мы мыслим это свойство в его обобщенной форме, пользуясь понятием резкости, сложившимся у нас на основании знакомства с этим явлением во многих других случаях (не только в боксе, но и в фехтовании; не только при ударе, но и при отбивании мяча и т. д.), т. е. как соединение силы с кратковременным прикосновением к поражаемому объекту. [4] Уже одна эта умственная операция позволяет нам отразить в своем сознании сущность явления: поражающая сила удара при нокауте заключается именно в его резкости.

Конкретизация. Абстракция всегда предполагает противоположную ей мыслительную операцию — конкретизацию, т. е. переход от абстракции и обобщения обратно к конкретной действительности. В учебном процессе конкретизация часто выступает как приведение примера для установленного общего положения. В соединении с абстракцией конкретизация является важным условием правильного понимания действительности, так как она не позволяет мышлению отрываться от живого созерцания явлений. Благодаря конкретизации наше мышление становится жизненным, за ним всегда чувствуется непосредственно воспринимаемая действительность. Отсутствие конкретизации приводит к тому, что знания становятся голыми абстракциями, оторванными от жизни, а потому и бесполезными.[3]

Анализ. Анализом называется мысленное разложение какого-либо сложного предмета или явления на составляющие его части. В практической деятельности анализ приобретает форму фактического расчленения предмета на составляющие его части. Возможность практически выполнить такое расчленение лежит в основе мысленного расчленения предмета на его элементы.[3] Например, думая о сложной структуре прыжка, мы мысленно выделяем в нем следующие основные части: разбег, толчок, фаза полета, приземление. Этот мысленный анализ облегчается тем, что и в действительности мы можем выделить эти моменты и совершенствовать в процессе тренировки скорость разбега, силу толчка, правильность группировки в полете и т. д.[4]

Синтез. Синтезом называется обратный анализу процесс мысленного воссоединения сложного предмета или явления из тех его элементов, которые были познаны в процессе его анализа. Благодаря синтезу мы получаем целостное понятие о данном предмете или явлении, как состоящем из закономерно связанных частей. Как и при анализе, в основе синтеза лежит возможность практически выполнить такое

воссоединение предмета из его элементов. При анализе выделяются не всякие части, а лишь те, которые имеют для данного предмета существенное значение. В таком физическом упражнении, как прыжок, можно отметить много разных элементов: движение рук, движение головы, мимику лица и т. д. Все эти элементы в той или иной степени связаны с данным упражнением, и мы их выделяем. Однако в процессе научного анализа мы опираемся не на эти, а на существенные части целого, без которого это целое не может существовать. [3] Существенными для прыжка являются не мимика лица или движения головы и рук, а разбег и толчок.[4]

Выделение существенных элементов при анализе сложного явления происходит не механически, а в результате понимания значения отдельных частей для целого. Прежде чем мысленно выделить существенные признаки или части, мы должны иметь хотя бы смутное общее синтетическое понятие обо всем объекте в целом, в совокупности всех его частей. Такое понятие возникает в результате предварительного, образующегося еще до детального анализа общего представления о предмете на основе практического знакомства с ним. [5]

Рассмотрев вышеописанные операции, можно предложить модель данных. В качестве примера, рассмотрим представление некоторого предложения данной моделью, изображенного на рисунке 2.

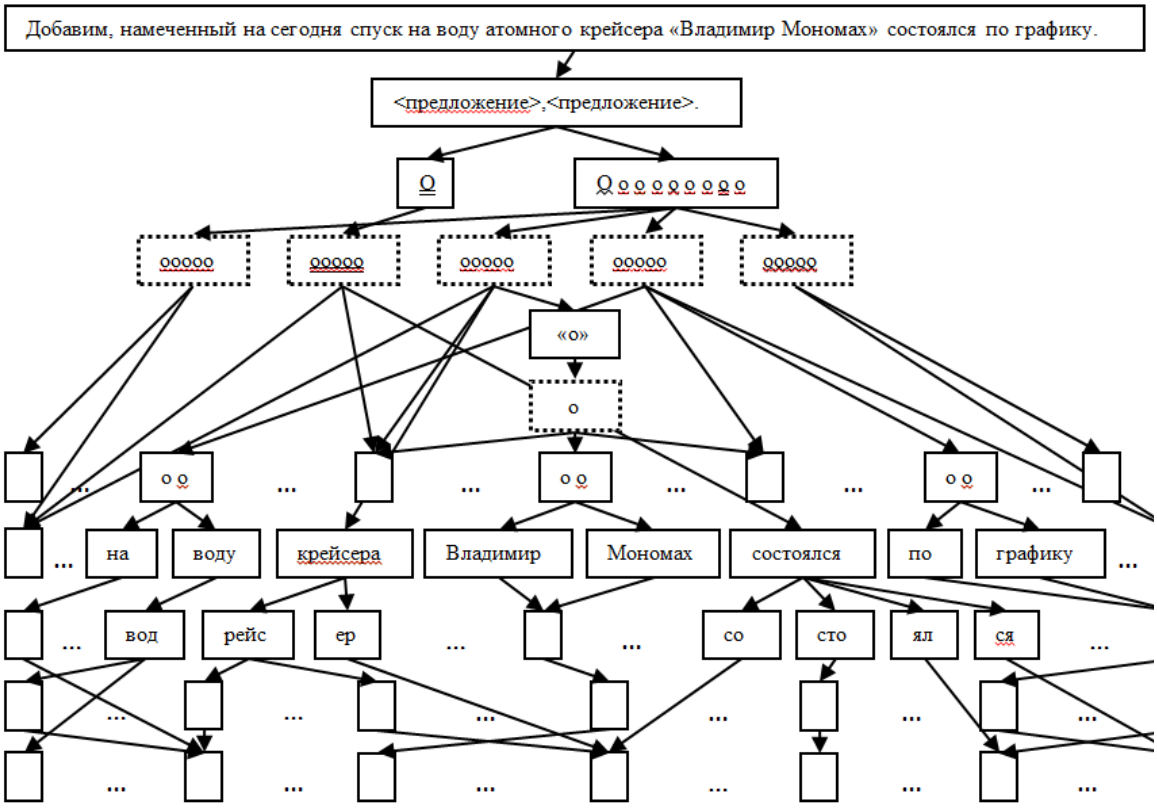


Рис 2. Представление предложения структурой реляционных множеств

Представление такой структурой, а именно реляционной структурой множеств, использует шесть основных аксиом множеств[1]:

- аксиому объёмности (два множества a и b равны тогда и только тогда, когда они имеют одни и те же элементы);

$$\forall a \forall b (a = b \leftrightarrow \forall c (c \in a \leftrightarrow c \in b))$$

- аксиому о существовании «элементарных множеств»;

$$\exists a \forall b (b \notin a) \wedge \forall a \exists c \forall b (b \in c \leftrightarrow b = c) \wedge \forall a_1 \forall a_2 \exists c \forall b (b \in c \leftrightarrow b = a_1 \vee b = a_2$$

- схему выделения (любому множеству a и свойству φ отвечает множество b, элементами которого являются те и только те элементы a, которые обладают свойством φ);

- аксиому множества подмножеств (для любого множества a существует множество b, состоящее из тех и только тех элементов, которые являются подмножествами множества a);

- аксиому объединения (для любого семейства a множеств существует множество , называемое объединением множества a, состоящее из тех и только тех элементов, которые содержатся в элементах множества a);

- аксиому выбора (для каждого семейства непустых непересекающихся множеств существует (по меньшей мере одно) множество d, которое имеет только один общий элемент с с каждым из множеств b данного семейства);

Ставя им в соответствие операции, выполняемые человеческой памятью (Рисунок 3):

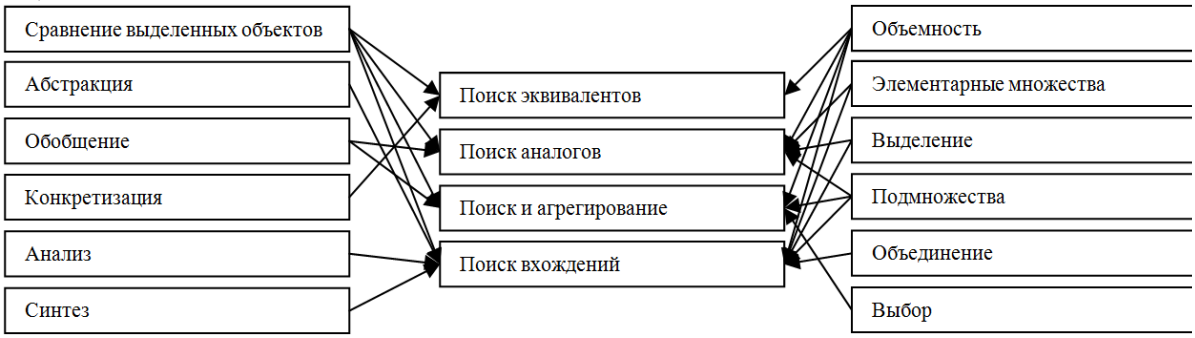


Рис 3. Соответствие мысленных операций и аксиом программным операциям

Довольно сложным вопросом является классификация и представления данных в нашем с вами сознании, можно считать, что разработка правильной модели является одним из главных параметров при построении моделей искусственного интеллекта.

Как известно в поставленной нами задаче, необходимо хранить огромные объемы данных, учитывая и сохраняя их взаимосвязи, предоставлять быстрый и удобный доступ. В современных базах данных часто используют реляционную модель данных, мы же усовершенствуем ее под нашу конкретную задачу и добавим свойства связей между таблицами данных, а в качестве элементов этих таблиц будем использовать множества двух типов:

- Вариантное (устанавливает отношение аналогии между членами)
- Объединяющее (устанавливает отношение совокупности между членами)

Реляционная модель данных (РМД) — логическая модель данных, которая является приложением к задачам обработки данных таких разделов математики как теории множеств и логики первого порядка.

Термин «реляционный» означает, что теория основана на математическом понятии отношения (relation). Для лучшего понимания РМД следует отметить три важных обстоятельства:

- модель является логической, то есть отношения являются логическими (абстрактными), а не физическими (хранимыми) структурами;
- для реляционных моделей верен информационный принцип: всё информационное наполнение модели представлено одним и только одним способом, а именно — явным заданием значений атрибутов в кортежах отношений; в частности, нет никаких указателей (адресов), связывающих одно значение с другим;
- наличие реляционной алгебры позволяет реализовать декларативное программирование и декларативное описание ограничений целостности, в дополнение к навигационному (процедурному) программированию и процедурной проверке условий.

Реляционная модель данных включает следующие компоненты:

- Структурный аспект (составляющая) — данные представляют собой набор отношений.
- Аспект (составляющая) целостности — отношения отвечают определенным условиям целостности. РМД поддерживает декларативные ограничения целостности уровня домена (типа данных), уровня отношения и уровня данных.
- Аспект (составляющая) обработки (манипулирования) — РМД поддерживает операторы манипулирования отношениями (реляционная алгебра, реляционное исчисление).

Над множествами, как элементами базы данных нашего ИИ, определены следующие операции оптимизирующие структуру связей между ними по следующим правилам:

- Поиск вхождений (правило,
  $\forall A \forall B, \nexists A \subset B: (\forall a_{i+k} = b_i / a_{i+k} \in A, b_i \in B, k \in D, i = \overline{0, |B|}\,))$

Рис 4. Представление поиска вхождений

- Поиск эквивалентов (правило,
  $\forall A \forall B, \nexists A = B: (\forall a_i = b_i / a_i \in A, b_i \in B, i = \overline{0, |B|}))$

Рис 5. Представление поиска эквивалентов

- Поиск аналогов (правило,
  $\forall A \forall B, \nexists A \cap B: (\forall a_i = b_i / a_i \in A, b_i \in B, i = \overline{n, m}: n < m, n > 0, m < |B|))$

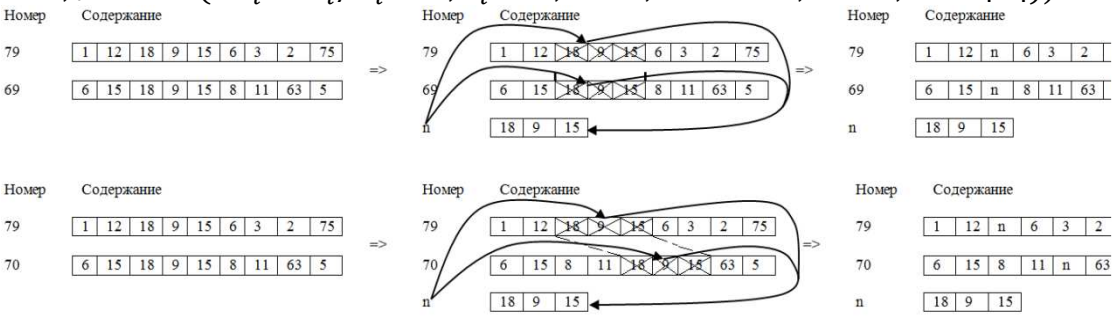


Рис 6. Представление поиска аналогов

- Поиск и агрегирование (правило,
  $\forall A \forall B, \nexists A \approx B: (\forall a_i = b_i / a_i \in A, b_i \in B, i = \overline{0,k} \& \overline{k,|B|}: 0 < k < |B|))$

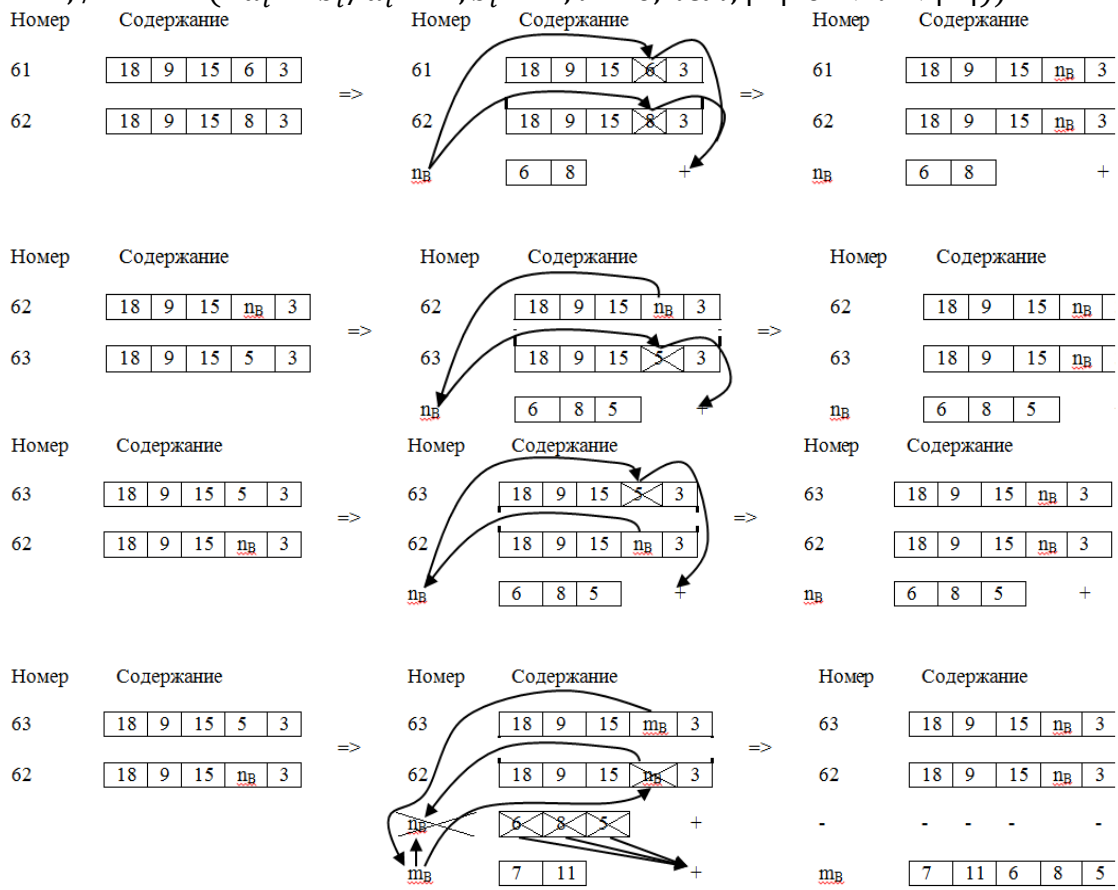


Рис 7. Представление поиска и агрегирования

Процесс генерации решения разбивается на подзадачи, для упрощенного рассмотрения, которые могут выполняться параллельно, т.к. алгоритмически являются независимыми:

- Сначала необходимо найти эквивалент введенной в обработку части данных в списках генерационных листов.
- Следует провести нисходящий генерационный анализ вхождений ячеек вариантов и прибавить их значения текущей темы.
- Выполняется восходящий анализ вариантов и после приемлемого выбора система переходит к генерации следующей ячейки в листе.
- Осуществляется сборка ячейки нисходящей генерацией.

На рисунках 8,9 и 10, предоставлены модели описанных подзадач.

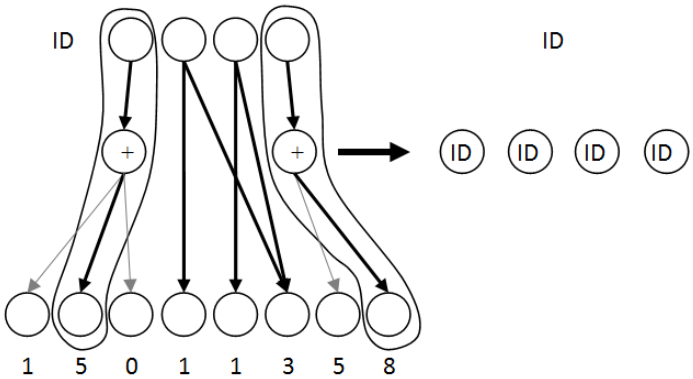


Рис 8. Нисходящая сборка в глубину по максимальным параметрам темы (аналогично поиску по вхождению)

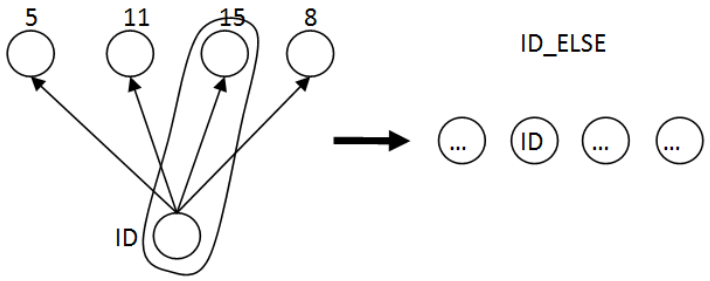


Рис 9. Восходящий поиск вхождения по максимальным параметрам темы

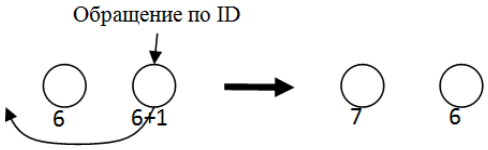


Рис 10. Продвижение операции по приоритету обращений

Помимо вышеописанных операций, каждое обращение к листу генерации продвигает его в списке листов на одну позицию.

Список литературы

1. К. Куратовский, А. Мостовский Теория множеств / Перевод с английского М. И. Кратко под редакцией А. Д. Тайманова. — М.: Мир, 1970. — 416 с.
2. А. Френкель, И. Бар-Хиллел Основания теории множеств / Перевод с английского Ю. А. Гастева под редакцией А. С. Есенина-Вольпина. — М.: Мир, 1966. — 556 с.
3. Коган А. Б. Нейрофизиологические механизмы мышления человека // Основы физиологии высшей нервной деятельности. — второе, переработанное и дополненное. — Москва: Высшая школа, 1988. — С. 335—350. — 368 с. — 10 000 экз. — ISBN 5-06-001444-4;
4. Маланов, С. В. Психологические механизмы мышления человека: мышление в науке и учебной деятельности / С. В. Маланов — М.: Издательство Московского психолого-социального института; Воронеж: Издательство НПО «МОДЭК», 2004. — 480 с.
5. Тихомиров О. К. Психология мышления. М.: 1984.
6. Dias F.M., Mota A.M. Comparison between Different Control Strategies using eural Networks // 9th Mediterranean Conference on Control and Automation. – Dubrovnik, Croatia, 2001.
7. Venayagamoorthy G.K., Harley R.G., Wunsch D.C. Implementation of Adaptive Criticbased Neurocontrollers for Turbogenerators in a Multimachine Power System”, IEEE Transactions on Neural Networks. – 2003. – Vol. 14, Issue 5. – P. 1047 – 1064.
8. McCalloch W.S., Pitts W. A logical calculus of the ideas immanent in nervous activity.// Bull. Math. Biophys. – 1943. – v.5. – pp. 115–133.

Использование генетических алгоритмов в задачах анализа данных и построения систем правил

Солодовников В.И.
Центр информационных технологий в проектировании РАН
Россия, Московская область, г. Одинцово, ул. Маршала Бирюзова, д.7а
Тел.: (495) 596-02-19, E-mail: info@ditc.ras.ru

Введение

Генетические алгоритмы (ГА), предложенные в 1975 году Джоном Холландом (John Holland) из Мичиганского университета, предназначены для решения задач оптимизации и отыскания глобального экстремума многоэкстремальной функции, но также нашли свое применение при решении задач «интеллектуального» анализа данных. Эти задачи характеризуются отсутствием априорных предположений о структуре выборки и виде распределений значений анализируемых показателей. В свою очередь генетические алгоритмы работают с кодовыми последовательностями (КП) безотносительно к их смысловой интерпретации и не привязаны к предметной области. Кодовая последовательность и ее структура задается понятием генотипа, а его интерпретация, с точки зрения решаемой задачи, понятием фенотипа. Каждая КП представляет, по сути, точку пространства поиска и называется особью или индивидуумом, а их набор образует исходное множество решений или популяцию. Наибольший интерес представляет совместное использование генетических алгоритмов с другими методами «интеллектуального» анализа данных, в частности с методами поиска логических закономерностей и нейросетевым подходом, что способно повысить качество получаемых решений.

1. Функционирование генетического алгоритма

Функционирование ГА состоит из нескольких этапов. На первом, происходит кодирование признаков характеризующих объект. Эти закодированные признаки называются генами, а совокупность таких признаков – хромосомой. В свою очередь, набор хромосом особи формирует генотип, т.е. особями популяции могут быть генотипы либо единичные хромосомы, в случае, если генотип состоит из одной хромосомы.

Формирование исходной популяции *K* происходит с использованием некоторого случайного закона, например, равномерного, на основе которого выбирается нужное количество точек поискового пространства. Исходная популяция может быть также результатом работы некоторого алгоритма оптимизации.

На каждом шаге работы генетический алгоритм обновляет исходное множество *K* путем создания новых КП и уничтожения бесперспективных, не удовлетворяющих критерию целевой функции. Каждое обновление интерпретируется как смена поколений и обычно идентифицируется по заданному размеру. Популяция обрабатывается с помощью процедур скрещивания, мутации и инверсии. Эти процедуры имитируют биологические процессы. В ходе работы процедур на каждой стадии эволюции получаются популяции со все более совершенными индивидуумами.

Критерием останова может быть одно из трех событий:

- сформировано заданное число поколений;

- популяция достигла заданного уровня качества;
- достигнут некоторый уровень сложности, при котором улучшение популяции не происходит.

После завершения работы ГА из конечной популяции выбирается та КП (особь), которая дает оптимальное значение целевой функции, и которая является, в итоге, результатом работы ГА.

2. Поиск логических закономерностей

Существует целый ряд методов для определения логических закономерностей. В большинстве случаев задается множество возможных событий $T = \{T_1, \ldots, T_n\}$, и из этого множества строятся цепочки конъюнкций. Среди них тем или иным способом (перебор, вероятностные методы и т.д.) отбираются наиболее характерные для одного из классов и не характерные для других. Основной проблемой является, как определить возможные события. К таким методам могут быть отнесены алгоритм Кора и случайный поиск с адаптацией (СПА). Другим подходом для поиска логических закономерностей являются методы, основанные на применении деревьев решений. Их построение обычно осуществляется с использованием алгоритмов обработки примеров (CLS, ID3 (Interactive Dichotomizer), CART (classification and regression trees) и др.). Здесь следует отметить, что путь от вершины корня до вершины листа также представляет собой цепочку конкатенаций элементарных событий, и проблема их определения остается.

2.1. Формулировка задачи

Рассмотрим задачу, в которой логические закономерности используются для отнесения объекта к одному из N классов $A = \{A_1, \ldots, A_N\}$. Определим количество популяций равное количеству классов N. Выбор такого количества популяций обусловлен тем, что оценку показателей точности и полноты правил при выделении логических закономерностей целесообразно производить для каждого класса. При единственной популяции происходит усреднение данной оценки. Тем самым рассматриваемый подход можно отнести к так называемым островным алгоритмам [1].

Особью (хромосомой) является само правило. Количество особей, в отличии от большинства ГА, в популяции переменно и определяется с одной стороны минимально возможным числом для образования родительских пар, а с другой – максимально возможной полнотой популяции. Обучающая последовательность из n объектов задается в виде пар: $< X_i, t_i >, i = 1 \div n$ где X_i - m–мерный вектор параметров $X_i = (x_{1i}, \ldots, x_{mi})$, $t_i \in A$ - принадлежность к классу, тогда хромосома имеет постоянную длину равную $m*2$, где m – число параметров (признаков), по которым проводится классификация и представляется в виде: $((x_{1,\min}, x_{1,\max}), \ldots, (x_{m,\min}, x_{m,\max}))$. Здесь $x_{i,\min}, x_{i,\max}$, соответственно верхняя и нижняя граница (область определения) значения i-го признака. Будем задавать генотип особи как множество пар $< w_{ijk}^1, w_{ijk}^2 >$, для которых выполняется условие: $x_{i,\min} \le w_{ijk}^1 \le w_{ijk}^2 \le x_{i,\max}$, $i = 1 \div m$, $j = 1 \div N$, $k = 1 \div K$, где K – количество особей в популяции. Таким образом, каждая особь в популяции представляет собой m – мерный гиперкуб в пространстве признаков со сторонами $< w_{ijk}^1, w_{ijk}^2 >$, который соответствует некоторому правилу. Исходные популяции строятся случайным образом, так чтобы максимально покрыть пространство признаков.

Объем общей популяции (состоящей из особей всех N популяций) может быть получен из решения задачи поиска количества гиперкубов со сторонами, размер которых равновероятно распределен в диапазоне $(0, x_{i,\max} - x_{i,\min})$, причем этот набор с вероятностью $P \to 1$ должен закрывать все точки пространства признаков. Необходимо учитывать тот факт, что если популяция мала, то генетического материала может не хватить для решения имеющийся задачи. Размер популяции также влияет на коэффициент применения операторов мутации и скрещивания.

Качество решения также зависит от методики создания популяции. Например, вместо инициализации популяции случайным образом, разработчик может добавить хромосомы, которые изначально представляют хорошее решение проблемы. Но это может привести к быстрому схождению популяции, т.е. потере разнообразия в генетическом фонде.

2.2. Целевая функция

Критерием качества получаемого решения являются точность и полнота. Однако, они не объединяются единой оценочной функцией, а применяются последовательно и зависят от возникающей ситуации. Так при отборе родительских пар решающую роль, как это будет показано ниже, играет точность, а при выполнении редукции основным показателем является полнота и в меньшей мере учитывается точность. Критерием останова является ситуация, когда точность и полнота достигают наперед заданного числа в каждой популяции. Будем считать наилучшей популяцию, в которой находятся особи, лучшим образом удовлетворяющие логическим закономерностям для данного класса. Тогда целевую функцию можно определить:

$$F_{kj} = \frac{r_{kj}}{\sum_{l=1}^{N} r_{kl}}$$

Здесь r_{kj} - количество объектов, удовлетворяющих критерию, заданному в генотипе особи k (т.е. объектов, попавших в гиперкуб, соответствующий данной особи) и принадлежащих классу A_j для популяции j (расстояние по Хэммингу). Фактически целевая функция определяет точность правила. Тогда целью генетического алгоритма является нахождение такой популяции, в которой содержится подмножество особей $K_R^j \subseteq K^j$, где K^j - общее количество особей для популяции, моделирующей принадлежность к классу j, для которого $\sum_{k \in K_R^j} F_{kj} \to 1$ и число объектов, принадлежащих данному классу в обучающей выборке максимально, то есть $\bigcup_{k \in K_R^j} \chi_{kj} = SSO_j$. Здесь χ_{kj} - множество объектов, удовлетворяющих критерию и принадлежащих классу A_j, сформулированному для особи k, SSO_j - множество объектов, принадлежащих к классу A_j среди множества исходных объектов [1].

2.3. Выбор родительских особей.

Одним из наиболее важных факторов функционирования ГА является выбор родительских пар для получения потомков. Могут быть выделены случайный и селективный механизмы отбора. В первом случае особи выбираются случайным образом, причем любая особь может стать членом нескольких пар. Во втором случае родительские пары выбираются в соответствии с некоторым условием. Например, значение приспособленности особи не должен быть меньше среднего значения по

популяции или способ формирования пар на основе близкого и дальнего родства. Под родством понимается расстояние между членами популяции как в смысле геометрического расстояния между особями популяции (для фенотипа), так и в смысле хемминговского расстояния между хромосомными наборами для генотипов.

Наиболее распространен способ, когда в качестве родительских пар отбираются особи с учетом значения целевой функции (в нашем случае с наибольшим значением F_{kj}). Поэтому в качестве вероятности выбора родительской КП можно использовать:

$$P_{kj} = \frac{F_{kj}}{\sum_{i=1}^{K} F_{ij}}$$

Это означает, что КП с наибольшим значением точности правила могут входить в несколько родительских пар.

2.4. Основные операторы

Непосредственная генерация новых КП осуществляется из двух родительских особей за счет оператора скрещивания (кроссовера, случайно–детерминированного обмена). Вероятность применения оператора скрещивания является одним из параметров генетического алгоритма и обычно выбирается достаточно большим, например от 0.9 до 1.

Различаются одноточечные, двухточечные и равномерные операторы скрещивания, в зависимости от вида разбиения КП. При использовании одноточечного оператора случайным образом выбирается точка разрыва, т.е. участок между соседними битами в строке. Обе родительские строки разрываются на два сегмента по этой точке. Затем соответствующие сегменты различных родителей склеиваются и получаются два генотипа потомков.

В двухточечном операторе выбирается две точки разрыва, и родительские хромосомы обмениваются сегментом, который находится между этими точками. В равномерном операторе скрещивания каждый бит первого родителя наследуется первым потомком с заданной вероятностью, в противном случае этот бит передается второму потомку. В дальнейшем, часть этих пар может перейти в популяцию следующего поколения.

После этапа скрещивания, выполняется оператор мутации. Мутация представляет собой случайное изменение КП. В нашем случае осуществляется случайное изменение какого-либо элемента пары $< w_{ijk}^1, w_{ijk}^2 >$, которая также выбирается случайным образом. Данные изменения должны удовлетворять требованиям: сохранение упорядоченности пары и попадание в диапазон области значений для заданной переменной. Оператор мутации позволяет находить локальные экстремумы с одной стороны и позволяет «перескочить» на другой локальный экстремум с другой. Вероятность выполнения этого оператора является еще одним параметром алгоритма.

Порождение потомков сопровождается уничтожением неперспективных особей. Формирование нового поколения производит оператор отбора, а выбор особей для уничтожения – оператор редукции.

Самыми распространенными операторами отбора являются:

- случайный равновероятный отбор;
- ранговый пропорциональный отбор, когда новая популяция формируется из родителей и потомков по принципу «побеждает сильнейший»;

- отбор пропорционально значению целевой функции, в котором чем выше здоровье хромосомы, тем больше вероятность того, что она будет выбрана для формирования нового поколения, но, если применять этот метод в чистом виде, то возможен отсев здоровых особей, а поэтому можно формировать новое поколение, используя генетический материал предшествующих поколений;
- элитный отбор, который заключается в переносе некоторого количества (например, 10%) самых здоровых хромосом в следующее поколение;
- отбор с вытеснением или метод турнира, в котором из популяции отбирается *k* особей, которые затем соревнуются за право попасть в следующее поколение; побеждает та хромосома, здоровье которой выше; турнир повторяется *l* раз.

Оператор редукции имеет ряд особенностей. В частности хромосомы в популяции упорядочиваются по точности, но при выполнении редукции может быть исключена особь, имеющая высокую точность, но не увеличивающая полноту популяции.

Еще одной особенностью алгоритма является введение оператора миграции, который позволяет переносить кодовую последовательность (особь) из одной популяции в другую. Он выполняется с некоторой вероятностью, если в популяции *j* встречается особь *k*, для которой $F_{kj} < F_{kp}$, $j \neq p$. Перенос осуществляется в популяцию *p*. Смысл этого оператора заключается в том, что правила, имеющие низкую точность для одного класса, могут иметь более высокую точность для другого класса. Такая ситуация наиболее часто возникает в начальных поколениях.

Следует отметить, что в отличии от большинства существующих генетических алгоритмов в рассматриваемом подходе популяции имеют переменный размер. Это обусловлено двумя причинами:

1. При описании различных классов может потребоваться различное количество правил, чтобы достичь требуемой точности и полноты.
2. Популяция в ходе выполнения должна иметь по размеру ограничения снизу, что связано с необходимостью наличия достаточного генетического материала для порождения новых потомков.

Однако, в этом виде алгоритм не лишен ряда недостатков, присущих другим подобным алгоритмам:

1. Любое правило всегда содержит число событий равное количеству признаков, по которому проводился анализ;
2. Могут встречаться подобные, фактически совпадающие правила.

Поэтому представляется целесообразным внесение в ГА набора дополнительных операций, рассчитанных на повышение качества получаемых результатов. В частности, после генерации *h* поколений (эпохи) выполняется операция, условно названная, совершенствование вида, направленная на повышение разнообразия правил, в ходе которой удаляются повторяющиеся правила и особи преобразуются так, чтобы как можно более компактно представлять правила.

3. Использование генетического алгоритма для обучения нейронной сети

В качестве кодовой последовательности может выступать нейронная сеть (НС), в частности, многослойный персептрон. В этом случае, ГА применимы для обучения нейросети, т.е. минимизации суммарной квадратической ошибки работы сети путем подстройки ее весовых коэффициентов.

Соотнесение основных структурных характеристик нейронной сети и параметров ГА осуществляется следующим образом:

- ген – весовой коэффициент нейронной сети;
- хромосома – набор генов или упорядоченный набор весовых коэффициентов нейронной сети, при этом каждая хромосома является возможным решением (т.е. таким набором весовых коэффициентов, который лучше подходят для решения имеющейся задачи);
- популяция – множество хромосом, вариантов наборов весовых коэффициентов (множество вариантов нейросети);
- эпоха – итерация, соответствующая созданию нового поколения хромосом.

Над хромосомами осуществляются операции скрещивания, мутации, отбора и редукции, таким образом, происходит параллельная обработка множества альтернативных решений. Стоит отметить, что операция мутации решает одну из проблем, присутствующих в обучении многослойной нейронной сети методом обратного распространения ошибки. Она позволяет популяции преодолевать локальные экстремумы и способствует защите от преждевременной сходимости. После обучения к полученной НС может быть применен алгоритм для извлечения правил и представления результата в виде дерева решений, например TREPAN.

Заключение

Внедрение средств автоматизации в системы «интеллектуального» анализа данных способно сократить сроки, повысить качество и эффективность принимаемых решений. Рассмотрены вопросы совместного использование генетических алгоритмов с методами поиска логических закономерностей и нейросетевым подходом, где ГА выступают в роли средства автоматизации для сегментации данных, поиска оптимального набора элементарных событий при поиске логических закономерностей, выделения наиболее значимых значений признаков и их сочетаний, обучения нейронной сети.

Список литературы

1. Солодовников И.В., Доронин В.А. Генетический алгоритм для поиска логических закономерностей в данных - // Информационные технологии в управлении. № 7. М.:2005 г.

2. Дюк В., Самойленко А. Data Mining. Учебный курс. – СПб: Питер, 2001.

3. Корнеев В.В., Гареев А.Ф., Васютин С.В., Райх В.В. Базы данных. Интеллектуальная обработка информации. – М.: Нолидж, 2000.

4. Назаров А.В., Лоскутов А.И. Нейросетевые алгоритмы прогнозирования и оптимизации систем. – СПб: Наука и техника, 2003.

5. Ежов А., Шумский С., Нейрокомпьютинг и его применение в экономике и бизнесе,1998.

Кластеризация пользователей системы дистанционного образования на основе модели нейронной сети ART1

М.А.Гаврилова, О.Ю.Ерёмин
«Компьютерные системы и сети», МГТУ им.Н.Э.Баумана
mariya.gavrilovaa@gmail.com

Аннотация. Дистанционная форма обучения все увереннее заявляет о себе, особенно в высшем образовании. Давно просчитано, что экономически это более выгодная форма обучения, по сравнению с очной формой. В данной статье рассматриваются наиболее существенные недостатки в сфере дистанционного образования. Особое внимание обращается на проблему кластеризации пользователей с помощью архитектуры сети Гроссберга и модели нейронной сети ART1.

Ключевые слова: дистанционное образование, нейронная сеть, кластеризация, ART1, пользователь.

В современном обществе дистанционная форма обучения является более демократичной, поскольку любой человек при сравнительно небольших материальных затратах может получить профессию, повысить квалификацию, переориентироваться в профессиональной деятельности, дополнить свое образование новыми областями знаний и т.д. [1,2,3,4].

Системы дистанционного обучения на сегодняшний день являются одним из эффективных способов обучения [1,2,3,4].

Они предоставляют ряд преимуществ, таких как:

1) свободное планирование учебных курсов;
2) возможность выбора времени для занятий;
3) индивидуальное обучение.

Также существует и ряд недостатков. Наиболее существенным из них является отсутствие преподавателя, либо его минимальное участие. Так как личное общение с преподавателем позволяет эффективно адаптировать учебный курс к текущему уровню учащегося.

Например, при индивидуальной работе с учеником преподаватель в режиме реального времени может обнаружить пробелы в знаниях, или не совсем полное усвоение учебной информации. За счет большого педагогического опыта преподавателя, при обнаружении подобных ситуаций, учебный курс корректируется с учетом успеваемости ученика и усвояемости им информации. Для этого ему предлагаются дополнительные учебные материалы или изменение подходов к преподаванию (увеличение интерактивного элемента образования, использование наглядных материалов и т.п.).

К сожалению, в системе дистанционного обучения невозможно предоставить каждому учащемуся таких же возможностей, как при индивидуальной работе с преподавателем, но при этом возможно сделать упор на индивидуальное обучение [3,4].

Поэтому *актуальной задачей* является повышения качества учебного процесса при работе с системой дистанционного образования за счет оценки степени освоения единиц учебного материала.

Единицей учебного материала может являться честь учебного курса: раздел, тема, подтема, отдельные рассматриваемые вопросы, вплоть до отдельных параграфов [1].

Поскольку невозможно создать бесконечное количество курсов с разным уровнем сложности и детализации рассматриваемых вопросов, то вполне естественной становится задача разбиения учащихся на отдельные группы (классы или кластеры) с одинаковыми результатами по освоению курса.

Предположим, что учебный курс состоит из трех разделов: раздел 1, раздел 2 и раздел 3.(см. Рисунок 1)

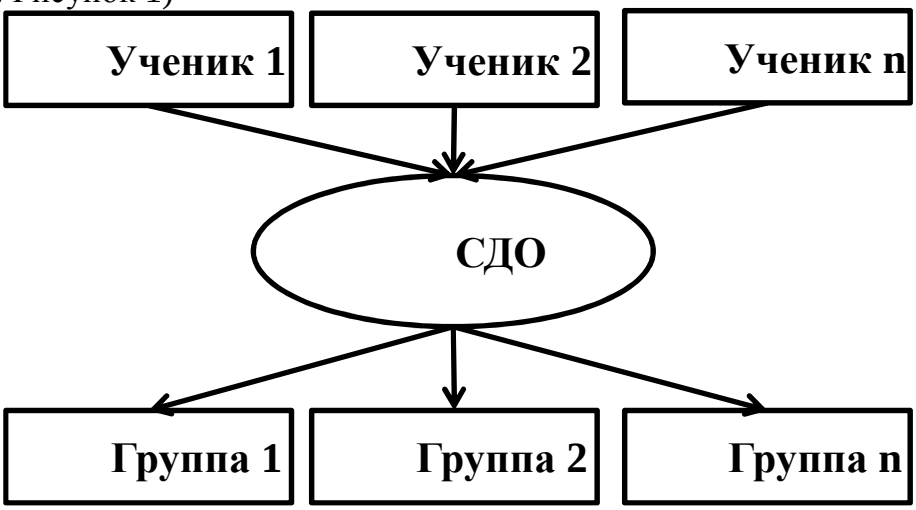


Рис. 1. Процесс разбиения учащихся на отдельные группы (кластеры)

Один учащийся хорошо освоил раздел 1, но у него возникли сложности с разделом 2, поэтому система дистанционного обучения должна проанализировать возникшие сложности и, например, предоставить этому учащемуся расширенную версию раздела 2, которая содержит больше примеров, иллюстративного материала и дополнительных объяснений.

Вопрос анализа трудностей, возникающих перед учащимся, позволяют его определить в группу, которая содержит людей, имеющих такие же сложности, и поэтому курс будет в дальнейшем адаптироваться к потребностям текущего ученика. Конечно, ученик может какие-то части курса усвоить быстрее, а какие-то покажутся ему слишком «избыточными», поэтому необходимо учесть этот фактор и подобрать для него более подходящую группу, либо создавать новую (если процесс обучения и усвоения материала существенно отличается от уже существующих групп). Если отклонения в процессе обучения студента от членов его группы несущественны, то, возможно, необходимо ввести коррекцию на учебный план этой группы. То есть, система должна не только создавать рекомендации для текущего ученика, но и адаптивно подстраивать его курс в соответствии с его успехами.

Таким образом, с одной стороны, необходимо дать рекомендацию ученику, то есть определить его принадлежность к определенной группе, но, с другой стороны, предлагается изменять подаваемый материал в течение процесса обучения конкретного ученика. Подобная ситуация является разновидностью, так называемой, *дилеммы стабильности-пластичности*.

Существующие методы кластеризации (статистические алгоритмы, персептроны и т.д.) обладают рядом недостатков, таких как, требование начального задания количества кластеров, необходимость участия человека при обучении, не позволяют модифицировать данные в кластерах в процессе своей работы, требуют большого количества параметров для настройки [7,8].

Тем не менее, искусственные нейронные сети позволяют решать задачу распознавания за счет своей способности к обобщению. Поэтому необходимо определить требования, к используемой нейронной сети, при которых:

1) нейросеть должна иметь способность к адаптации;

2) возможность работы без обучающей выборки, то есть обучение нейросети должно осуществляться в процессе работы;

3) нейросеть, при имеющейся адаптивности, должна быть стабильной;

4) нейросеть должна формировать устойчивые кластеры;

5) нейросеть должна иметь возможность менять свою чувствительность.

Подобной нейронной сетью, обладающей всеми вышеперечисленными свойствами, является модель нейронной сети адаптивно-резонансной теории ART1, которая позволяет проводить кластеризацию двоичных входных векторов, и при этом решает *дилемму стабильности-пластичности* [5,6,7,9,10,11].

Нейронная сеть адаптивно-резонансной теории ART1 работает с входными векторами, компоненты которых являются бинарными (двоичными), то есть могут принимать значения 0 и 1.

Размер входных векторов определяет структуру нейронной сети ART1, так как количество компонентов во входном векторе должно быть равно количеству нейронов во входном (первом) слое нейронной сети.

Структурная схема нейронной сети ART1

Нейронная сеть ART1, позволяющая стабилизировать процесс обучения, является логическим развитием архитектуры сети Гроссберга [12, 13, 14].

ART1 представляет собой двухслойную нейронную сеть, в которой все нейроны первого слоя связаны со всеми нейронами второго слоя (связи «Слой 1 — Слой 2» или L1-L2), и также обратные связи всех нейронов второго слоя с нейронами первого слоя (связи «Слой 2 — Слой 1» или L2-L1). Также в модель ART1 входит подсистема ориентирования (это один специфический нейрон, который отличается от нейронов в слоях) и управление усилением (либо это два дополнительных нейрона, либо дополнительная обратная связь из второго слоя к первому слою).

Выходы нейронов обоих слоев представляет собой краткосрочную память (STM – short-term memory), а весовые матрицы связей между слоями, в свою очередь, — долгосрочную память (LTM – long-term memory).

Замечание. В сети Гроссберга связи «Слой 2 — Слой 1», подсистема ориентирования и управление усилением отсутствуют.

Структурная схема сети ART1 показана на рисунке 2.

Описание работы сети ART1

Связи «Слой 1 — Слой 2» обеспечивают операцию кластеризации. Когда входной образ предъявлен сети, то он нормализуется в первом слое, а затем умножается на весовую матрицу связей «Слой 1 — Слой 2». Затем во втором слое (который является однослойной нейронной сетью конкурентного обучения) происходит состязание, которое позволяет определить, какая строка весовой матрицы наиболее близка к входному вектору. Именно эта строка будет в дальнейшем замещена текущим входным вектором.

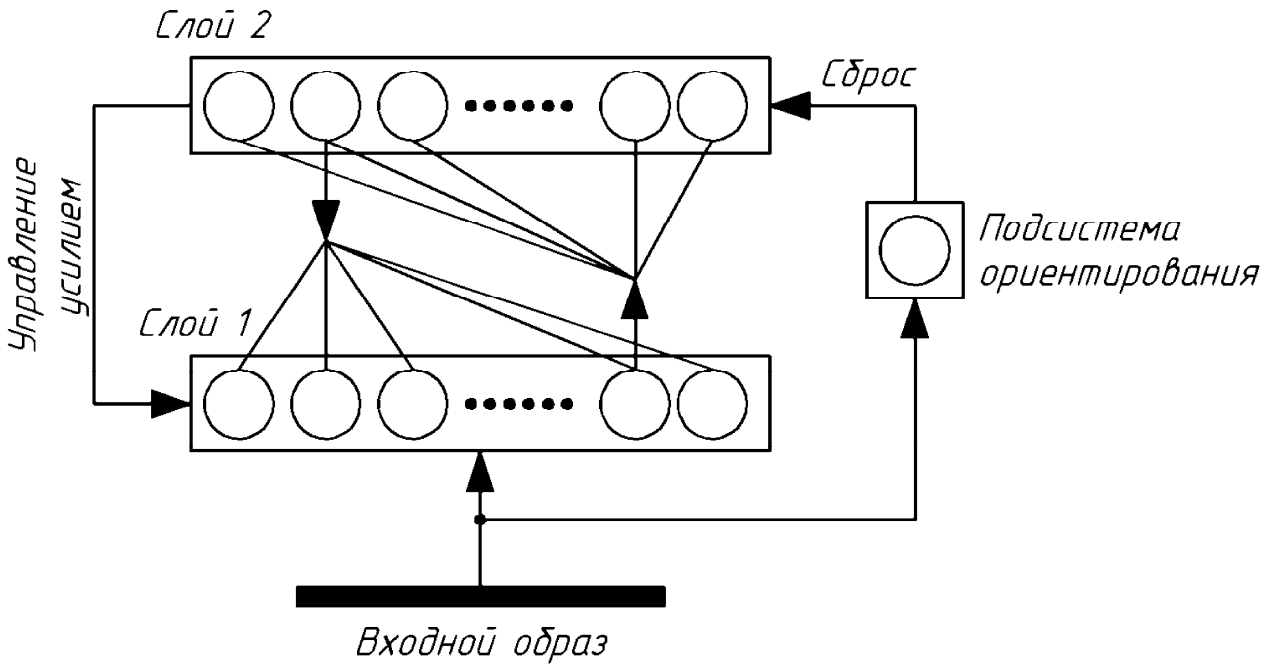


Рис. 2. Структурная схема двухслойной сети ART1

После окончания обучения нейронной сети каждая строка весовой матрицы «Слой 1 — Слой 2» будет содержать образ кластера, так называемого прототипа. В сети ART1 для обучения используются также обратные связи «Слой 2 — Слой 1», которые являются связями выходной звезды Гроссберга (outstars) [10-11], которые обеспечивают повторный вызов образа. То есть, когда узел во втором слое активирован, то он создает сигнал ожидания (expectation), который передается в первый слой. Сигнал ожидания является образом прототипа, соответствующего активному выходному нейрону второго слоя. Далее, первый слой осуществляет сравнение образа ожидания и входной образ.

В случае, когда входной образ существенно отличается от образа ожидания (степень «похожести» определяется параметром бдительности), то в подсистеме ориентирования формируется сигнал сброса (reset) для второго слоя. Этот сброс дезактивирует текущий нейрон-победитель и удаляет соответствующее этому нейрону ожидание. Причем этот нейрон переходит в неактивное состояние до самого окончания работы с текущим входным вектором. Далее во втором слое опять повторяется состязание, и на этот раз выигрывает другой нейрон, который передает новый сигнал ожидания в первый слой через связи «Слой 2 — Слой 1». Этот процесс продолжается до тех пор, пока ожидание в связях «Слой 2 — Слой 1» не обеспечит необходимой близости с входным образом.

Работа нейронной сети строится по следующему алгоритму [5]:

1. На вход сети поступает образ;
2. Входной образ сравнивается с имеющимися образами-прототипами;
3. Выбирается входной нейрон с наибольшим сигналом, который соответствует наиболее похожему образу-прототипу;
4. Если степень схожести входного образа и образа-прототипа не удовлетворяет параметру бдительности, то нейрон, соответствующий этому образу-прототипу, тормозится и далее снова выполняется шаг 2, иначе шаг 5.
5. Если степень схожести входного образа и образа-прототипа удовлетворяет параметру бдительности, то в сети наступает резонанс – все выходные нейроны

тормозятся, кроме того, который соответствует текущему образу-прототипу, и далее выполняется шаг 6.

6. Происходит обновление образа-прототипа с учетом текущего входного образа, с которым произошел резонанс;

7. Входной образ убирается, восстанавливаются сигналы всех заторможенных нейронов и далее снова выполняется шаг 1.

Основные этапы кластеризации учащихся показаны на Рисунке 3.

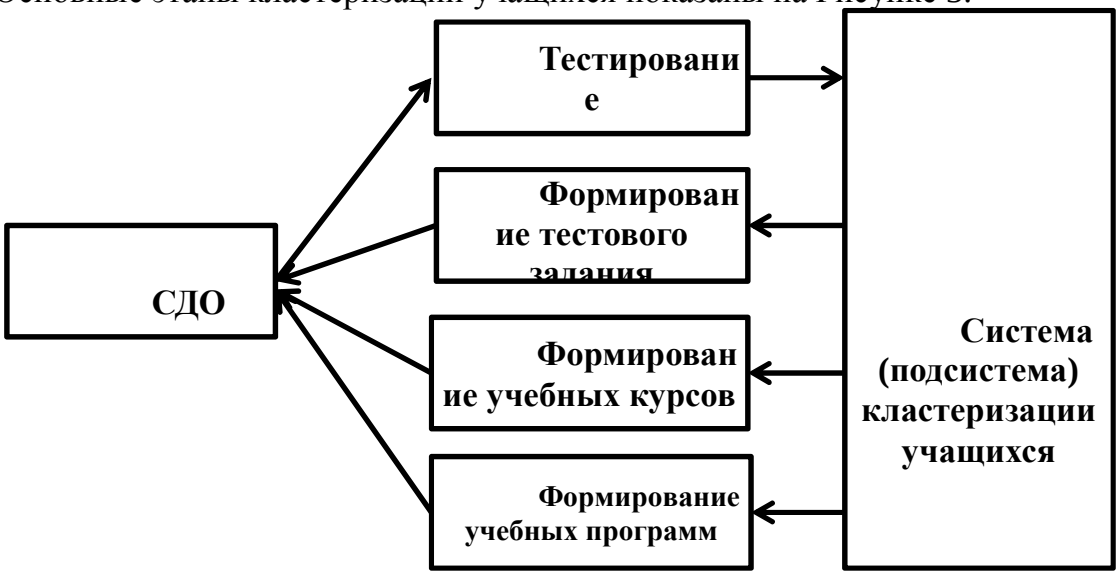


Рис. 3. Основные этапы кластеризации учащихся

Пока есть свободные нейроны в выходном слое нейронной сети, им могут сопоставляться все новые и новые кластеры.

Исходя из принципа работы сети ART1, необходимо изначально задавать количество выходных нейронов больше, чем предполагается иметь кластеров (в идеальном случае в выходном слое бесконечное количество нейронов, что, естественно, не соответствует реальной жизни).

Выводы

В результате применения подхода к кластеризации учащихся, основанного на модели нейронной сети адаптивно-резонансной теории должны быть получены следующие результаты:

- улучшить качество построения учебных курсов для студентов системы дистанционного образования;
- снизить затраты на разработку и внедрение новых курсов за счет более полного использования уже имеющихся ресурсов.

Список используемой литературы

1. Кречетников К.Г., Черненко Н.Н. Дистанционное обучение. Достоинства, недостатки, вопросы организации: аналитический обзор // http://www.hr-portal.ru/article/distantsionnoe-obuchenie-dostoinstva-nedostatki-voprosy-organizatsii-krechetnikov-k-g-cherne (дата доступа: 20.09.2012)
2. Полат Е.С. Педагогические технологии дистанционного обучения [Электронный ресурс] // http://www.distant.ioso.ru/seminary/09-02-6/tezped.htm (дата доступа: 18.09.2012)
3. Полат Е.С, Моисеева М.В., Петров А.Е. Педагогические технологии дистанционного обучения / Под ред. Е.С.Полат. — М., "Академия", 2006.
4. Концепция создания и развития единой системы дистанционного образования в России: утверждена Постановлением Госкомитета РФ по высшему образованию от 31 мая 1995 г. № 6 // КонсультантПлюс: ВысшаяШкола: Программа информационной поддержки российской науки и образования: Специальная подборка правовых документов и учебных материалов для студентов: учебное пособие. - 2007. - Вып.4.
5. Джонс М.Т. Программирование искусственного интеллекта в приложениях/ М.Тим Джонс.- М.: ДМК Пресс, 2006.
6. Гроссберг С. Внимательный мозг [Электронный ресурс]// Открытые системы, №4, 1997. – URL: http://www.osp.ru/os/1997/04/179198/ (дата обращения:06.04.2011)
7. Комарцова Л.Г. Нейрокомпьютеры: Учеб. Пособие для вузов. /Л.Г.Комарцова, А.В.Максимов. – 2-е изд., перераб. и доп. –М.: изд-во МГТУ им.Н.Э.Баумана, 2004
8. Хайкин С. Нейронные сети: полный курс, 2-е изд., испр. – М.: ООО «И.Д.Вильямс», 2006.
9. Carpenter G.A. A Massively Parallel Architecture for a Self-Organizing Neural Pattern Recognition Machine / G.Carpenter, S. Grossberg //Computing Vision, Graphics, and Image Processing, 37. pp. 54-115, 1987
10. Grossberg S. Adaptive Pattern Classification and Universal Recording: I. Parallel Development and Coding of Neural Feature Detectors // Biological Cybernetics, No. 23, 1976
11. Grossberg S. Adaptive Pattern Classification and Universal Recording: II. Feedback, Expectation, Olfaction, Illusion // Biological Cybernetics, No. 23, 1976
12. Barbara M. ART and Pattern Clustering. - Proceedings of the 1988 Connectionist Model Summer. - Published by M.Kaufmann, San Mateo, CA, pp. 174-185.
13. Devaux S. Classification hybride ART-CS: Apprentissage par renforcement, vision et rorotique. Rapport de projet de fin d'etudes [Электронный ресурс] / Tours, [1996]. - URL: http://sde.eduvax.net/artcs/ (дата обращения: 06.04.2011).
14. Hagan M.T. Neural Network Design / M.T. Hagan, H.B. Demuth, M.H. Beale. - Boston, MA: PWS Publishing, 1996.

Исследование многомерных данных в задачах многопараметрической оптимизации

А.Е. Бондарев, В.А. Галактионов
Институт прикладной математики им. М.В. Келдыша РАН, Россия, Москва
bond@keldysh.ru; vlgal@gin.keldysh.ru

1. Введение

Современное развитие вычислительной техники и возможность реализации вычислений в параллельном режиме позволяют решать все более масштабные задачи численного моделирования в механике жидкости и газа. Так, все более востребованными становятся расчеты не только прямых задач, где требуется моделировать явление при известных исходных данных, но и расчеты обратных задач, где необходимо определить при каких определяющих параметрах возникает то или иное явление. Такая постановка требует многократного решения прямых задач и решения задачи оптимизационного анализа. Итоговым решением подобных задач служат многомерные массивы дискретных величин, выражающие зависимость искомой функции (управляющего параметра) от определяющих параметров рассматриваемой задачи. Полученные многомерные численные результаты нуждаются в обработке и анализе.

Изначально методы анализа данных были ориентированы в ходе исторического развития на обработку результатов физических и инженерных экспериментов, а также на обработку результатов статистических наблюдений. Реальные физические эксперименты проводились в трехмерном пространстве. Поэтому методы, предназначенные для работы с одномерными, двумерными и трехмерными результатами, отработаны и общеизвестны. По мере развития вычислительных мощностей и алгоритмов стало возможно обрабатывать огромные массивы данных наблюдений и экспериментов в различных областях. Стали возможны решения обратных и оптимизационных задач, которыми стали многомерные массивы, выражающие зависимости управляющих параметров в многомерном пространстве определяющих параметров. Потребовались инструменты, позволяющие анализировать данные, реализованные не в трехмерном, а в многомерном пространстве.

За последнее десятилетие анализ многомерных данных стал одним из основных направлений прикладной математики, активно развивающимся и применяющимся практически во всех областях исследований. Анализ многомерных данных (АМД или MDA – Multidimensional Data Analysis) является одной из наиболее популярных и востребованных междисциплинарных областей знания и активным инструментом синтеза различных дисциплин [1]. Одним из наиболее известных методов анализа многомерных данных является метод главных компонент и его обобщения для нелинейных случаев [2]. Методы анализа многомерных данных реализуются в тесной взаимосвязи и взаимодействии с методами факторного и кластерного анализа [3].

Существует множество разных определений АМД, диктуемых конкретной областью и целями научного исследования, используемыми алгоритмами и т.п. В нашем случае, где источником многомерных данных служат решения задач оптимизационного анализа нестационарных процессов механики жидкости и газа, мы будем рассматривать анализ многомерных данных как совокупность методов и алгоритмов, позволяющих получить максимально возможную информацию о массиве

числовых данных, расположенных в некоторой области многомерного пространства. Такой подход позволяет формулировать задачу в более общем виде и избегать ограничений, накладываемых использованием конкретных методов, решаемой задачей и прочими подобными факторами.

2. Общая постановка задачи анализа многомерных данных

Рассмотрим функцию $\boldsymbol{F(\Omega)}$, заданную дискретным образом в области $\boldsymbol{\Omega}$, расположенной в n-мерном пространстве $\boldsymbol{X^n}$. Пространство $\boldsymbol{X^n}$ задано с помощью конечного множества ортогональных векторов $\boldsymbol{X = (\overline{X}_1, \overline{X}_2, ..., \overline{X}_n)}$. Многомерные данные – это числовые значения функции $\boldsymbol{F(\Omega)}$, заданные в точках n-мерного пространства $\boldsymbol{(x_1, x_2, ..., x_n)}$. Для определения расстояния в пространстве $\boldsymbol{X^n}$ выбирается метрика. Далее будем предполагать в качестве метрики обычное евклидово расстояние.

Таким образом, многомерные данные представляют собой набор числовых значений, заданный в точках области $\boldsymbol{\Omega}$: $\boldsymbol{F(\Omega) = F(x_1, x_2, ..., x_n)}$.

Задача анализа многомерных данных в этой постановке формулируется как получение максимально возможной информации о свойствах функции $\boldsymbol{F(\Omega)}$ и ее поведении в области $\boldsymbol{\Omega}$.

Проблема анализа любых числовых данных (в том числе и для стандартного числа измерений – одномерных, двумерных и трехмерных) в общем случае реализуется по схеме, представленной на рис. 1. Сначала проводится общая обработка данных – для того, чтобы иметь общую информацию о данных в области $\boldsymbol{\Omega}$ вычисляются глобальные минимумы и максимумы искомой функции, локальные минимумы и максимумы по координатным направлениям, средние значения функции и дисперсия по направлениям. Эта информация позволяет нам получить первичное представление о поведении функции $\boldsymbol{F(\Omega)}$ в области определения и, в случае необходимости, провести нормировку функции, нужную для дальнейших действий.

Рис.1. Общая схема анализа числовых данных

Следующая стадия – это представление исследуемых данных в графической форме с применением всех доступных исследователю средств визуализации. По мере возможности проводится попытка представления полученных геометрических форм в аналитическом виде. По сути, получение аналитической зависимости для любой рассматриваемой в данной постановке функции $\boldsymbol{F(\Omega)}$ является конечной целью анализа данных. Для реализации этой цели можно применить аппроксимацию геометрических форм с помощью простых геометрических элементов – линий, плоскостей, участков сфер и т.д.

Будем называть стандартными те случаи, где количество измерений $n \le 3$, т.е. одномерные, двумерные и трехмерные. Для стандартных случаев способы реализации схемы, приведенной на рис.1, общеизвестны. Различные типы графического представления данных для стандартного числа измерений подробно описаны в [5].

Для стандартного числа измерений реализация общей схемы анализа данных возможна лишь потому, что человек, существуя в трехмерном пространстве, обладает укладывающимися в сознание геометрическими образами и представлениями для пространств с числом измерений $n \le 3$. Для многомерных пространств с числом измерений $n > 3$ подобных геометрических образов у человека нет. Следовательно, не имеется и надежных методов графического представления данных в многомерном пространстве, кроме проецирования во вложенные пространства со стандартным числом измерений. Таким образом, при отсутствии возможности реализовать общую схему анализа данных, для случаев многомерных пространств остаются два основных направления исследования данных:

- методы и подходы, позволяющие понизить размерность исследуемого пространства до стандартной;
- попытки построения визуальной концепции для представления многомерных данных в графическом виде.

3. Оптимизационные задачи как источник многомерных данных

Предположим, что целью численного исследования является изучение условий возникновения некого события, реализующегося в нестационарном газодинамическом процессе. Событием считаем то, что интересует исследователя, например, возникновение новой пространственно-временной структуры (отрывной зоны, вихря), достижение изучаемой величиной определенного значения, достижение частицей определенной точки в счетной области и т.п. Формализованная постановка обратной задачи выглядит в общем случае следующим образом:

Пусть имеется математическая модель нестационарного процесса и надежный численный метод для решения этой модели. В этом случае мы можем решать прямую задачу численного моделирования нестационарного процесса. Допустим, что в моделируемом процессе происходит некое событие (явление, эффект). Численное решение $F = F(x, x_1, ..., x_n)$ выбранной задачи формируется в процессе математического моделирования и определяется управляющим параметром x и конечным набором определяющих параметров задачи $(x_1, ..., x_n)$. Обозначим $\overline{X} = (x, x_1, ..., x_n)$ и введем *функционал события* $\Phi(F(\overline{X}))$, который на решении задачи принимает, подобно логической переменной, два значения: 1 – если событие, интересующее исследователя, наступило (независимо от рода события) и 0 – если событие не наступило.

$\Phi(F(\overline{X})) = 0$ - событие не наступило (1)

$\Phi(F(\overline{X})) = 1$ - событие наступило.

Пусть x' - значение управляющего параметра, при котором наступает изучаемое явление. Тогда общую постановку задачи можно записать формально следующим образом:

- найти $\min\limits_{\Delta x} I(\Delta x)$ при фиксированных значениях определяющих параметров $(x_1^*,...,x_n^*)$, где $I(\Delta x)$ - функционал следующего вида

$$I(\Delta x) = 1 - \Phi(F(\overline{X})), \quad \Delta x = x - x' \qquad (2)$$

Таким образом, наша задача формально состоит в минимизации функционала $I(\Delta x)$ при помощи вариации управляющего параметра. А в реальности, варьируя Δx, мы должны с приемлемой точностью отыскать значение x', то есть то значение управляющего параметра, при котором событие наступает.

Мы получаем одно значение $x'(x_1^*,...,x_n^*)$ для управляющего параметра при фиксированных определяющих параметрах. Но задача исследования состоит в том, чтобы построить зависимость $x'(x_1,...,x_n)$ для всех возможных значений определяющих параметров. Таким образом, если мы имеем в диапазоне разбиения каждого определяющего параметра M точек, то для того чтобы найти значения x' управляющего параметра для всех наборов $(x_1^*,...,x_n^*)$, необходимо решить M^n однотипных задач вида (2). В результате решения этого набора задач находятся все точки в исследуемом пространстве определяющих параметров, где происходит событие.

Рассматривая $(x_1,...,x_n)$ как набор базисных векторов, можно представить пространство определяющих параметров $L(x_1,...,x_n)$, имеющее размерность n. Тогда в общем случае задачу оптимизационного анализа можно сформулировать как нахождение в пространстве L всех подобластей L^*, где наблюдается изучаемое событие, т.е. $\Phi(L^*) = 1$.

Попутно решается задача фильтрации тех точек пространства определяющих параметров, где ожидаемое событие не наступает. Нельзя ожидать при выборе диапазона изменения определяющих параметров, что искомое событие наступит в каждой точке внутри выбранного диапазона. Поэтому, если для конкретной точки $(x_1,...,x_n)$ пространства определяющих параметров для любых значений управляющего параметра x, событие не наступает, данная конкретная точка изымается из рассмотрения.

Общий алгоритм решения задачи оптимизационного анализа в последовательном режиме вычислений представлен в работе [4]. Данный алгоритм в целом предполагает решение очень большого количества обратных задач численного моделирования (M^n), каждая из которых предполагает, в свою очередь решение большого количества прямых задач. Это обстоятельство делает реализацию вышеизложенного алгоритма весьма затруднительной с точки зрения временных затрат. В работе [6] рассматривается наиболее простой и эффективный вариант распараллеливания данного алгоритма. В силу того, что процессы решения ОЗ происходят фактически без обменов информацией между процессорами, распараллеливание здесь сводится к организации интерфейса, управляющего распределением вариантов по процессорам и сбором данных в единый массив результатов. Идеология параллельных вычислений в данном случае принимает форму «многозадачного параллелизма». Таким образом, организация решения рассматриваемой оптимизационной задачи в параллельном режиме позволяет быстро

и эффективно получить дискретную зависимость управляющего параметра от определяющих параметров задачи $x'(x_1,...,x_n)$ в виде многомерного массива.

4. Обработка многомерной информации

Полученный многомерный массив данных в исходном виде не может быть использован для анализа содержащейся в нем научной информации. Он нуждается в предварительной обработке. Для числа измерений, превышающего стандартное ($n > 3$), общая схема анализа числовых данных приобретает вид, представленный на рис. 2.

Рис.2. Общая схема анализа данных большой размерности

Для перехода от стадии общей обработки данных к стадии визуального представления в случае многомерности необходимо предпринимать определенные шаги по дополнительной обработке данных. В общем случае сначала необходимо проводить процедуру кластеризации с целью уменьшения или разбиения рассматриваемой области многомерного пространства. Эта процедура является необходимой при анализе слабоструктурированных данных (результатов экспериментов или наблюдений). В случаях, когда точки многомерного пространства задаются заранее с помощью сеточного разбиения, проведение кластеризации не является необходимым.

В работе [4] рассматривались современные попытки построения визуальной концепции для представления многомерных данных, а также отмечалось отсутствие на сегодняшний день адекватного и надежного способа подобного визуального представления. Следовательно, для анализа информации, содержащейся в полученном многомерном массиве необходимо понизить размерность массива. Рассмотрим наиболее распространенные практические способы понижения размерности.

Первый способ представляет собой поиск координатного направления с наименьшей дисперсией. Согласно общей схеме анализа данных в процессе общей обработки данных вычисляются средние значения функции S_i и дисперсии D_i по координатным направлениям. С помощью полученного набора $D_1, D_2,...D_n$ находим координатное направление j с наименьшей дисперсией данных $D_{\min} = \min\limits_{i=1,n}\{D_i\}$.

Далее проверяем выполнение следующего условия:

$$D_j = D_{\min} \le \varepsilon * D_i, \forall i \ne j \qquad (3)$$

где ε - малая величина, задаваемая пользователем.

Если условие (3) выполняется, то по координатному направлению j значения исследуемой функции заменяются на константу, равную среднему значению функции

по этому координатному направлению S_j. Таким образом, размерность исходного многомерного пространства понижается на единицу.

Изложенный подход обладает рядом недостатков:

- в основе этого подхода лежит допущение о том, что больше всего информации несут те направления, в которых дисперсия входных данных максимальна, т.е. заложена непосредственная связь между величиной дисперсии и информационной ценностью;

- условие (3) выполняются далеко не всегда, например, если данные образуют в многомерном пространстве гиперсферу, то эти условия заведомо не выполняются;

- в выборе малой величины ε заложен произвол, что предполагает наличие определенного опыта работы с данными у пользователя.

Однако, несмотря на эти недостатки, для небольшой размерности пространства $n = 4,5$ во многих случаях в практических задачах данных подход работает успешно.

Второй распространенный способ понижения размерности заключается в построении графических проекций на стандартное число измерений с фиксацией переменных, не участвующих в построении проекции. Далее проводится анализ вида проекций, который во многих случаях позволяет определить визуально координатное направление с наименьшей диперсией, и, исключив его, понизить размерность рассматриваемой области. Также данный подход очень полезен в тех случаях, когда из набора дисперсий по направлениям $D_1, D_2, ... D_n$ нельзя выделить существенно наименьшую. В этих случаях в задачах небольшой размерности $n = 4;5$ для построения аналитической зависимости часто используется метод разделения переменных, приведенный в [6].

Оба вышеизложенных подхода не являются строго обоснованными. Скорее, это алгоритмы выдвижения гипотез, нуждающихся в проверках. Однако эти методы позволяют получать реальные практические результаты. Многие из известных нам законов физики, механики, химии, экономики и т.д. получены именно таким образом.

Более современным способом понижения размерности массива данных является метод главных компонент (англ. Principal Components Analysis, PCA). Суть метода состоит в переходе от исходной системы координат к новому ортогональному базису в рассматриваемом многомерном пространстве, оси которого ориентированы по направлениям максимальной дисперсии массива данных. Различные варианты реализации метода главных компонент и его обобщений для нелинейных случаев подробно представлены в работах [2,7]. Геометрическая постановка задачи нахождения главных компонент формулируется согласно [7] следующим образом. В многомерном пространстве ищется вектор направления, задающий прямую, вдоль которой дисперсия максимальна (или сумма квадратов расстояний от точек данных до прямой минимальна). Таким образом определяется первая главная компонента. Далее рассчитывается множество векторов первых остатков, которое лежит в пространстве, ортогональном первой главной компоненте и имеющем размерность на единицу меньше исходной размерности. Для нового пространства, образованного этим множеством векторов, снова ищется направление с максимальной дисперсией. Так рассчитывается вторая главная компонента. Снова рассчитывается множество векторов вторых остатков и т.д.

Метод главных компонент и его нелинейные обобщения являются мощными развивающимися инструментами анализа данных, однако в задачах анализа данных вычислительной механики жидкости и газа он применялся редко. В настоящее время,

когда одним из основных направлений параллельных вычислений в механике жидкости и газа становится решение оптимизационных задач [4,6], формирующих результаты в виде многомерных массивов данных, применение метода главных компонент содержит в себе большой потенциал. Необходимо заметить, что все вышеперечисленные методы понижения размерности рассматриваемого многомерного массива данных могут применяться комплексно, дополняя друг друга и обеспечивая верификацию проводимых расчетов.

5. Пример решения конкретной задачи

Данный подход к организации параллельного расчета был применен к конкретной задаче о нестационарном взаимодействии сверхзвуковых струй [4]. В качестве события рассматривалось возникновение новой пространственно-временной структуры течения (ПВС), в качестве управляющего параметра использовалась скорость повышения нерасчетности струи V^*, а в качестве определяющих параметров были выбраны характерные числа Маха, Рейнольдса, Прандтля и Струхаля $(M_\infty, \mathrm{Re}_\infty, \mathrm{Pr}_\infty, Sh_\infty)$ для данной задачи. Рассматривались разбиения определяющих параметров по 5 и 10 точек на каждый параметр в диапазоне его изменения, что вело к необходимости расчета 625 и 10000 обратных задач соответственно. Общая задача решалась в параллельном режиме. Для проведения расчетов использовался вычислительный комплекс К100 (ИПМ им.М.В.Келдыша РАН). При организации интерфейса для управления параллельным расчетом использовалась технология MPI [8]. В результате расчетов был получен 4-мерный массив результатов $V^*(M_\infty, \mathrm{Re}_\infty, \mathrm{Pr}_\infty, Sh_\infty)$. К полученному массиву было применено логарифмическое преобразование по координатному направлению Re_∞ и проведено сравнение дисперсий по координатным направлениям. Оценка дисперсий по направлениям показала, что дисперсия по направлению $\lg \mathrm{Re}_\infty$ существенно меньше дисперсий по остальным направлениям. Построение и анализ трехмерных проекций исходного 4-мерного массива данных подтвердило этот результат. На рис.3 представлена зависимость $V^*(M_\infty, \mathrm{Re}_\infty, \mathrm{Pr}_\infty)$ в виде изоповерхностей.

Это позволило понизить размерность и рассматривать уже трехмерный массив результатов $V^*(M_\infty, \mathrm{Pr}_\infty, Sh_\infty)$. На рис. 4 представлена зависимость $V^*(M_\infty, \mathrm{Pr}_\infty, Sh_\infty)$ в виде изоповерхностей.

Полученные данные были также обработаны с помощью программного комплекса ViDaExpert, что позволило определить главные компоненты в исходном многомерном пространстве. По результатам общего визуального представления массива данных после понижения размерности был сделан вывод о том, что для целей грубой усредненной оценки полученное решение оптимизационной задачи можно аппроксимировать с помощью зависимости, представленной в виде плоскости:

$$V^* = V^*(M_\infty, \mathrm{Pr}_\infty, Sh_\infty) = -0.1 M_\infty + 0.115 \mathrm{Pr}_\infty + 0.24 Sh_\infty$$

Таким образом, было проведено исследование многомерного решения задачи оптимизационного анализа возникновения новой пространственно-временной структуры в потоке в полном соответствии со схемой анализа, представленной на рис. 2 с использованием приемов понижения размерности и визуального представления.

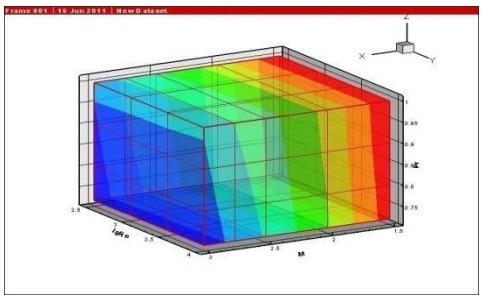


Рис. 3. Зависимость критической скорости повышения нерасчетности от чисел Маха, Рейнольдса и Прандтля

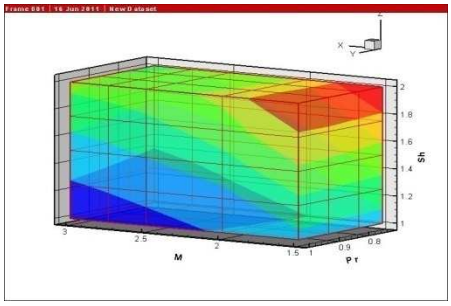


Рис. 4. Зависимость $V^*(M_\infty, \mathrm{Pr}_\infty, Sh_\infty)$

Заключение

Развитие параллельных вычислений выдвигает на первый план в вычислительной механике жидкости и газа решение оптимизационных и обратных задач. Подобные решения, как правило, представлены дискретно в виде многомерных массивов данных. Выработка методики анализа данных подобной структуры, их визуальное представление, обобщение этих данных в виде квазианалитических зависимостей является важной и актуальной задачей. Данная работа представляет проблему анализа многомерных данных как решений оптимизационных задач, формулировку общей постановки задачи, схему обработки и анализа данных и ее способы реализации. Общий подход и способы реализации представленной схемы анализа данных показаны на примере решения в режиме параллельных вычислений конкретной оптимизационной задачи.

Литература

[1] Esbensen K. Multivariate Data Analysis – in Practice. 5-th Edition, 2002, CAMO Process AS, Oslo, Norway.

[2] A. Gorban, B. Kegl, D. Wunsch, A. Zinovyev (Eds.), Principal Manifolds for Data Visualisation and Dimension Reduction, LNCSE 58, Springer, Berlin – Heidelberg – New York, 2007.

[3] Ким Дж., Мюллер Ч. и др. Факторный, дискриминантный и кластерный анализ. М.: Финансы и статистика, 1989. – 216 с.

[4] Бондарев А.Е. Оптимизационный анализ нестационарных пространственно-временных структур с применением методов визуализации / научный электронный журнал «Научная визуализация», Национальный Исследовательский Ядерный Университет "МИФИ", М., 2011, Т.3, N 2, С.1-11.

[5] Бондарев А.Е., Галактионов В.А., Чечеткин В.М. Анализ развития концепций и методов визуального представления данных в задачах вычислительной физики / Журнал вычислительной математики и математической физики, 2011, Т. 51, N 4, С. 669–683.

[6] Бондарев А.Е., Галактионов В.А. Анализ многомерных данных в задачах многопараметрической оптимизации с применением методов визуализации / Научная визуализация. Т.4, № 2, с.1-13, 2012.

[7] Зиновьев А. Ю., Визуализация многомерных данных, Красноярск, Изд. КГТУ, 2000. 180 с.

[8] Pacheco P., Ming W. MPI User's Guide in Fortran, http://parallel.ru/ftp/mpi

Проблемы построения имитационных моделей для проектирования технологических процессов

Зародов А.Ф., Ульянов Н.Г.
Московский институт электроники и математики «МИЭМ ВШЭ», кафедра ИТАС
Zaf_48@mail.ru

Построение и использование имитационных моделей при проектировании технологических процессов требует разработки и использования специальных подходов, учитывающих особенности применяемых физических явлений и реализованных на их основе технологических процессов.

Одним из самых важных этапов является разработка концептуальной схемы модели, при этом необходимо определить структурные особенности исследуемой системы, вид и степень взаимодействия между физикой процесса и технической реализацией [1].

На этом этапе осуществляется формализация системы, т.е. переход от реального объекта к некоторой логической схеме. Такая формализация позволяет осуществить переход от слабоструктурированного, словесного описания реальности к системе принятых терминов и формальных понятий о природе и параметрах физических явлений воздействующих на обрабатываемые объекты. Однако, одной из существенных особенностей построения моделей при разработке технологических процессов является необходимость включения в модель, элементов для которых отсутствует структурно-функциональное описание. При этом предполагается наличие экспериментальных данных, которые могут определять связи «вход-выход». Для построения моделей таких элементов необходимо решить ряд проблем связанных с выделением факторов, играющих существенную роль при реализации процесса.

Эксперимент и построенная на его основе модель, может потерять информативность, если существенные факторы, влияющие на процесс, не будут включены в модель. С другой стороны, включение малозначимых факторов ведет к ее усложнению. Т.е. для выделения значимых факторов необходимо провести предварительные исследования. Одним из основных подходов в данном случае является применение так называемого отсеивающего эксперимента. При этом предполагается, что существенная информация находится в отдельных, как правило, небольших участках области значений значимых факторов. Для извлечения значимой информации необходимо выделить информативные участки, а представление о поведении процесса строить через описание выделенных отдельных классов участков.

Другой важной проблемой является использования программных средств, реализующих имитационную модель, наряду с необходимой точностью получаемых результатов, является скорость моделирования. В отдельных случаях время необходимое для получения окончательных результатов моделирования технологического процесса может лежать в пределах от нескольких секунд, а иногда минут и требовать от пятидесяти до восьмидесяти процентов ресурсов компьютера, в зависимости от количества заданных параметров, что снижает эффективность работы программы. Для увеличения скорости работы моделирующей программы, без потери точности, возникает вопрос о создании логической системы, работающей параллельно с основной программой, которая в процессе своей работы способна накапливать статистику, т.е. создавать базу данных результатов для заданных

параметров [2], на основе которой можно прогнозировать, с достаточной степенью точности, влияние изменений того или иного входного параметра на конечный результат моделирования. В конечном счете использование данного подхода позволит сократить время создания и отладки модели технологических процессов.

Литература

1. Марка Д., МакГоуэн К. Методология структурного анализа и проектирования: Пер. с англ. - М.:1993.
2. Солодовников И.В., Зародов А.Ф. Управление экспериментом с имитационными моделями сложных динамических систем // Тематический сборник научных трудов “Проектирование цифровых систем управления летательными аппаратами”, - Харьков, ХАИ, 1988, с. 111 - 114.

«Автоматизация проектных исследований надежности и стойкости аппаратуры»

Методика оценки накопленной дозы функционального узла БРЭА КА при воздействии низкоинтенсивных излучений

М.А. Артюхова*, С.Н. Полесский**
**Кафедра «Радиоэлектроники и телекоммуникаций»*
***Кафедра «Информационные технологии и автоматизированные системы»*
Московский институт электроники и математики Национального исследовательского университета «Высшая школа экономики»
109028, г. Москва, Большой Трехсвятительский пер., д.3
maya.artyukhova@gmail.com, spolessky@hse.ru

Аннотация: При разработке БРЭА КА важно своевременно оценить стойкость аппаратуры к воздействию ИИ КП. Излучения низких энергий (частицы энергий до 1 МэВ) приводят к радиационным эффектам от накопленной дозы – параметрическим и функциональным отказам элементов вследствие деградации материалов. Это делает актуальным создание метода математического моделирования поля распределения накопленной дозы для учета его при проведении задач оптимизации конструирования функциональных узлов на ранних этапах проектирования.

Ключевые слова: радиация, стойкость, космический аппарат, проектирование.

Одной из двух причин нарушения работоспособности бортовой радиоэлектронной аппаратуры (БРЭА) космического аппарата (КА) при воздействии ионизирующего излучения космического пространства (ИИ КП) является изменение параметров электронной компонентной базы (ЭКБ) вследствие деградации материалов под действием накопленной дозы радиации (под действием частиц низких энергий - до 1 МэВ). Экранирование БРЭА позволяет существенно снизить уровень накопленной дозы (см. рис.1). Однако современные концепция конструирования БРЭА КА направлена на уменьшение массо-габаритных характеристик и создание унифицированных негерметизированных платформ [5, 6].

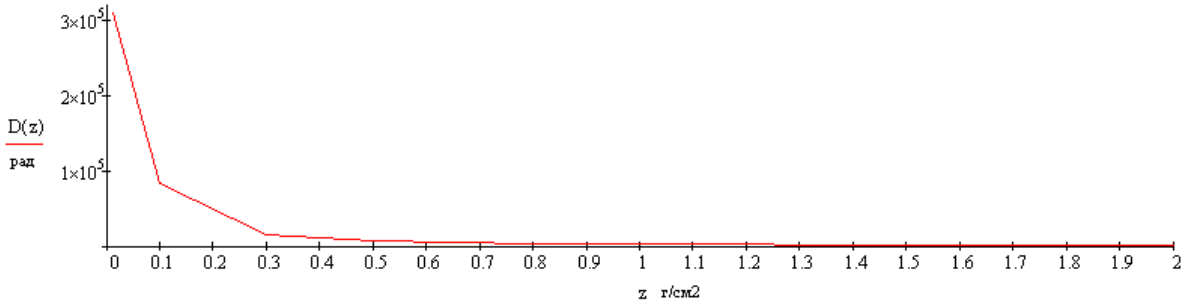


Рис. 1. Вид зависимости уровня накопленной дозы от величины массовой защиты

Задача обеспечения радиационной стойкости БРЭА КА осложняется еще и тем, что применяемая ЭКБ иностранного производства (ИП) довольно часто имеют показатели уровней стойкости к накопленной дозе, граничащие с установленным минимальным уровнем стойкости для конкретной БРЭА. Это делает необходимым с наибольшей точностью оценивать прогнозируемую величину накопленной дозы для каждого конкретного элемента в каждом конкретном устройстве.

В работе предлагается метод оптимизации функциональных узлов из состава БРЭА в процессе проектирования путем использования модели построения поля распределения накопленной дозы на уровне функционального узла. Такое поле позволит непосредственно разработчикам БРЭА получить наглядное представление о прогнозируемой стойкости элементов и, при необходимости, провести мероприятия по улучшению показателей радиационной стойкости БРЭА на ранних этапах проектирования.

Метод основан на учете в оценке величины накопленной дозы влияния «теневой» защиты - защиты создаваемой взаимной экранировкой элементов и элементами конструкции аппаратуры. На рис. 1 приведен вид зависимости уровня накопленной дозы от величины массовой защиты, с увеличением защиты значение накопленной дозы падает. Графика снижения имеет вид экспоненциального закона.

На рис. 2 приведена дочерняя функциональная модель метода построения поля распределения накопленной дозы для радиоэлектронного функционально узла.

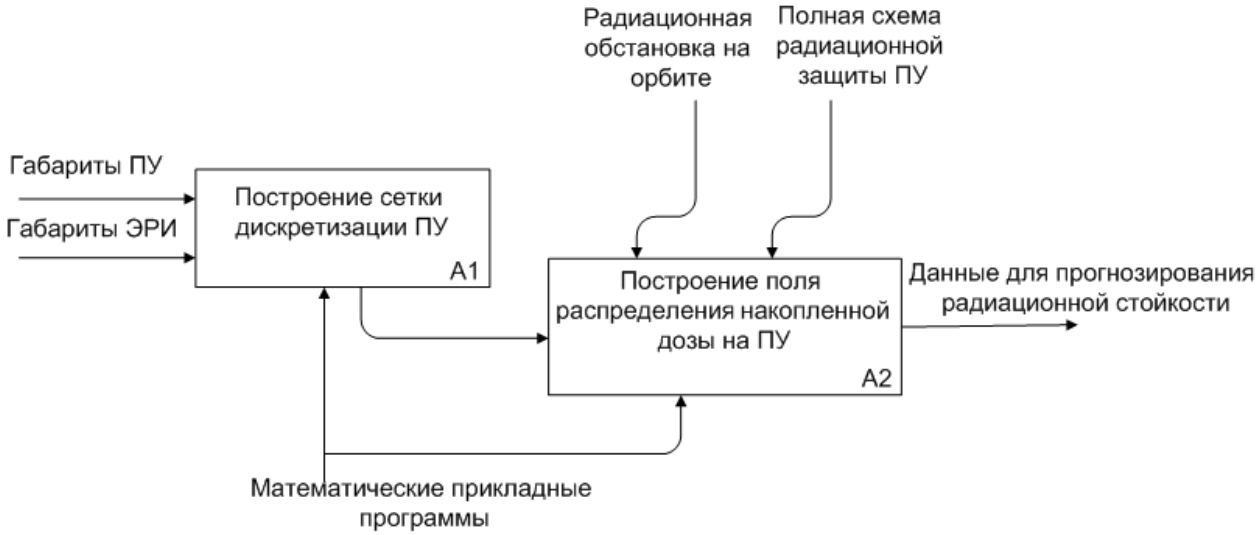


Рис. 2. Дочерняя функциональная модель метода построения поля распределения накопленной дозы

Для построения поля распределения накопленной дозы на функциональном узле необходимо иметь следующие данные:

1. Модель внешних воздействующих факторов (ВВФ);
2. Схема радиационной защиты (чем точнее схема, тем точнее будет поле);
3. Габариты устройства, в которое будет входить функциональный узел.

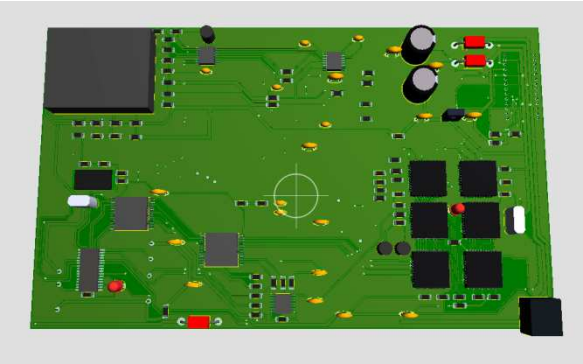
Доминирующая часть БРЭА конструктивно выполняется в виде плат-рамок, то есть радиоэлектронные функциональные узлы (ФУ) упрощенно можно представить в виде параллелепипеда. ФУ представляет из себя набор функциональных узлов, скрепленных вместе шпильками и закрытых с торцов крышками.

Исходя из габаритов устройства, строится сетка дискретизации. Основа функционального узла – печатная плата, которая является двухмерной поверхностью прямоугольной формы, что облегчает задачу. Для построения сетки используется метод конечных элементов. Автоматическое построение сеток конечно-элементных моделей с помощью симплекс-элементов для двумерных задач рассмотрено во многих работах, например, в [1-2]. Размер шага сетки выбирается исходя из габаритов платы и средних значений геометрических размеров ЭРИ.

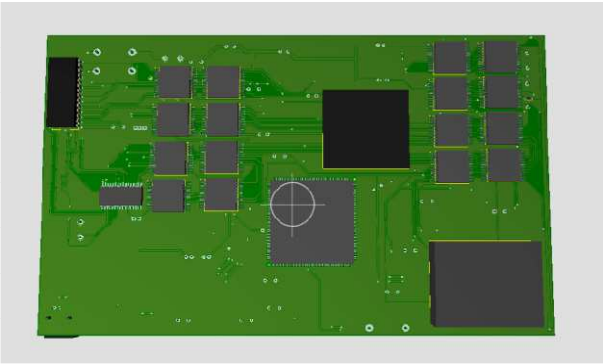
Далее в каждом узле сетки дискретизации рассчитывается величина уровня накопленной дозы. Для этого вычисляются значения шести телесных углов, соответствующих шести сторонам параллелепипеда – модели функционального узла. Задача решается последовательно через необходимые соотношения прямоугольной стереометрии, плоской тригонометрии и сферической тригонометрии. Данный метод носит название лучевого метода, и применим в случае относительной однородности величин массовой защиты в каждом из шести телесных углов.

В случае неоднородности защиты узлов для расчета телесных углов применяется метод секторирования. Сущность метода состоит в том, что ФУ разделяют на ряд сектором с вершинами в узле сетки дискретизации. Сектора выбираются таким образом, чтобы в пределах сектора величина массовой защиты была одинаковой.

На рис. 3 представлена модель функционального узла, выполненного в виде печатной платы с двухсторонним монтажом. Габариты функционального узла 15х25 см.



Сторона А



Сторона Б

Рис. 3. Модель функционального узла

Для разработчиков БРЭА КА, возможно, более полезным будет графическое представление поля распределения уровней накопленной дозы на ФУ, так как, соотнесенное с координатами элементов, оно даст достаточную информацию для оценки стойкости элемента к накопленной дозе.

На рис. 4 приведен пример графического представления результатом расчетов поля распределения накопленной дозы для стороны Б платы с учетом того, что функциональный узел помещен в корпус с равными величинами массовой защиты по всем направлениям. Использована цветовая индикация сравнительно благоприятных и критичных областей. Как видно, область окрашенная зеленым цветом и соответствующая минимальной накопленной дозе соответствует положению DC-DC преобразователя на стороне А платы. Поле распределения накопленной дозы для стороны А будет таким же, с той разницей, что область минимальной накопленной дозы будет соответствовать положению DC-DC преобразователя на стороне Б платы.

Сопоставляя физическое поле распределения накопленной дозы и схему размещения элементов на печатном узле, разработчик имеет необходимые данные для оптимизации конструкции ФУ в процессе проектирования. Одна из главных целей задач оптимизации схемы и конструкции – это снизить погрешность проектирования при выполненных требованиях по регламентируемым характеристикам, изложенным в техническом задании и принятому разработчиком показателю оптимальности.

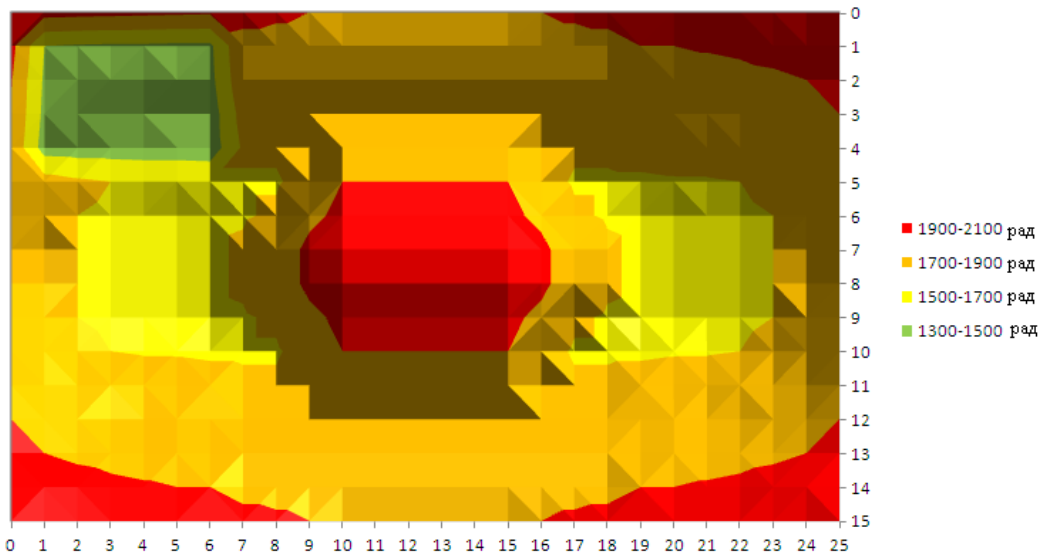


Рис. 4. Графическое представление результатов построения поля распределения накопленной дозы

Критерием стойкости ФУ БРЭА КА к воздействию низкоинтенсивного излучения является коэффициент запаса, определяемый как соотношение предельной накопленной дозы элемента и дозы, накопленной на элементе за срок активного существования КА.

Задача выполняется путем параметрической оптимизации модели физических процессов по параметру величины накопленной дозы с использованием критерия:

$$D = \sum_{j=1}^{N} \int_{\zeta_j^H}^{\zeta_j^B} f_j\left(\zeta_j\right)\left(y_j - y_j^H\right)^2 d\zeta_j$$

где y_j и y_j^H - выходные характеристики модели; $f_j\left(\zeta_j\right)$ - нормирующий множитель; ζ_j^H, ζ_j^B - нижний и верхний пределы изменения независимой величины.

Оптимизация по критерию проводится градиентными методами, в которых вектор градиент с составляющими указывает направления наибольшего возрастания критерия, что повышает эффективность алгоритмов оптимизации:

$$\frac{\partial D}{\partial q_k} = 2\sum_{j=1}^{N} \int_{\zeta_j^H}^{\zeta_j^B} f_j(\zeta_j)\left(y_j - y_j^H\right) A_{q_k}^{y_j} d\zeta_j$$

где $A_{q_k}^{y_j}$ - абсолютная функция чувствительности первого порядка.

Исходя из выше изложенных особенностей процесса обеспечения радиационной стойкости ФУ БРЭА КА, была разработана следующая концепция методики обеспечения спецстойкости при воздействии низкоинтенсивного ИИ.

Общий вид совокупности основных задач, входящих в методику обеспечения радиационной стойкости при воздействии низкоинтеснивных ИИ представлен на рис. 5. На представленной схеме блоки 7 – 10 отражают расчетные задачи и их традиционную последовательность [3], блоки 1 – 6 и 13 – 18 – эвристические задачи, которые необходимо предварительно решить для расчетных задач.

Поля распределения уровней накопленной дозы на уровне функционального узла позволяют на начальном этапе правильно задать граничные условия по накопленной дозы для выбора электронной комплектующей базы, на более поздних этапах – иначе разместить элементы на плате либо рационально применить экранирование. Данные поля позволяют проводить однопараметрическую

оптимизацию, на основе соотношения предельной накопленной дозы элемента и уровня воздействия на него. Основной задачей на данном этапе разработки методики является создание математических моделей для проведения однопараметрической оптимизации для распространенных геометрических моделей исполнения БРЭА и интеграция данных моделей в методику получения проектных решений с последующим слиянием методики с методами расчета надежности.

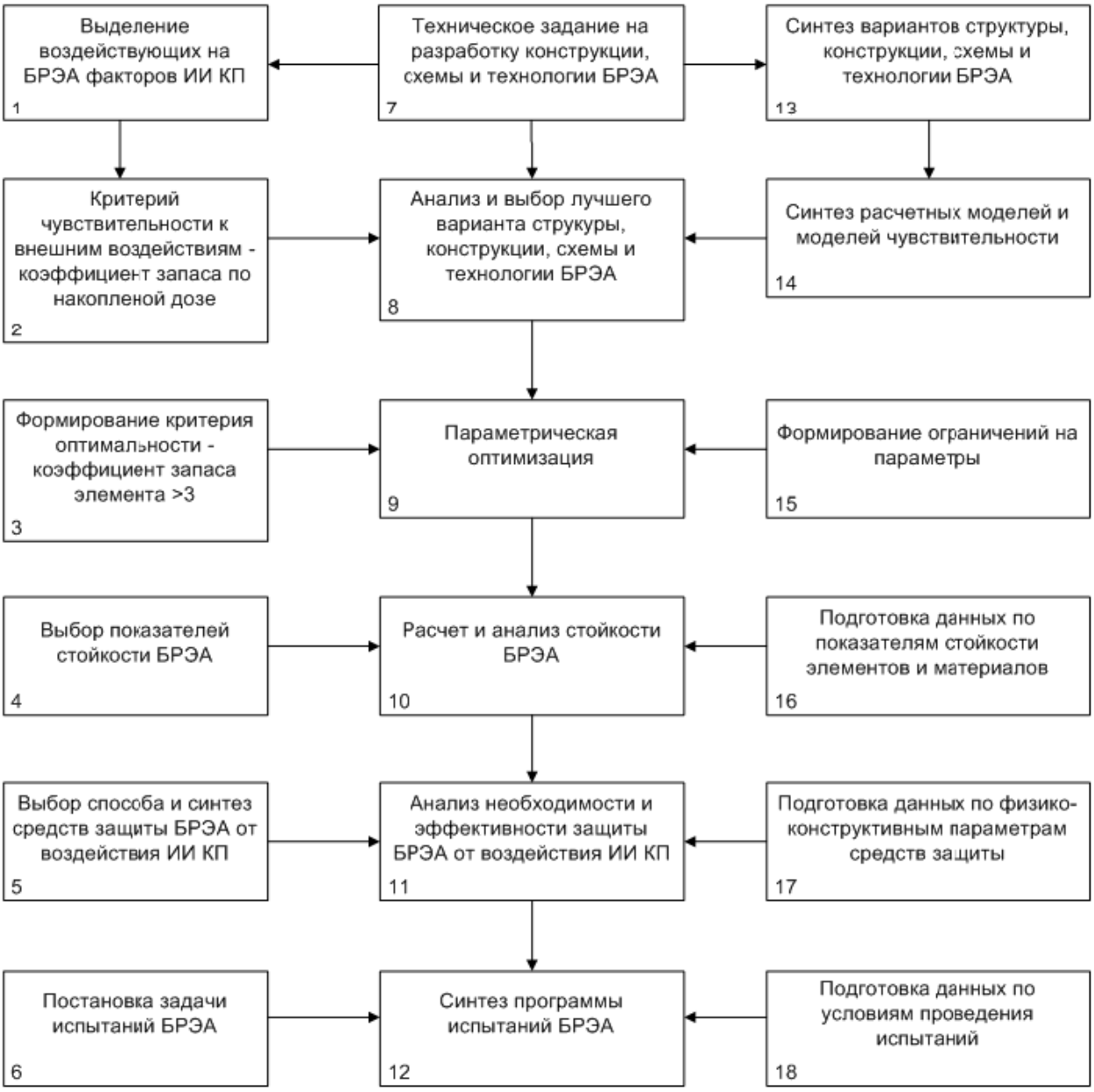


Рис. 5. Синтез, анализ и оптимизация в задачах обеспечения радиационной стойкости проектируемой БРЭА, решаемых на основе математического моделирования радиационных полей

Создаваемая методика позволит разработчику, на основе приведенных на рис. 5 задач, с помощью комплекса моделей получить проектные решения, соответствующие, в том числе и требованиям по радиационной стойкости.

Список литературы

1. Математика и САПР: В 2 кн. / П. Жермен-Лакур [и др.] – М.:Мир, 1989. – Кн. 2. – 264 с.

2. Скворцов А.В. Обзор алгоритмов построения триангуляции Делоне //Вычислительные методы и программирование, 2002, №3.-с. 14–39.

3. Гольдин В.В., Журавский В.Г., Сарафанов А.В., Кофанов Ю.Н. Информационная поддержка жизненного цикла электронных средств: Монография. – М.: Радио и связь, 2002. – 379 с.

4. ГОСТ 18298-79. Межгосударственный стандарт. Стойкость аппаратуры, комплектующих элементов и материалов радиационная. Термины и определения.

5. [Электронный ресурс]. Режим доступа: http://www.iki.rssi.ru/seminar/tarusa200406/1-03.pdf

6. [Электронный ресурс]. Режим доступа: http://144.206.159.178/ft/7719/646383/12961415.pdf

Анализ построения локальной вычислительной сети по критерию надёжности на базе «тонких» и «толстых» клиентов

А.А. Гавриленко, С.Н. Полесский
Кафедра «Информационные технологии и автоматизированные системы»
Московский институт электроники и математики Национального исследовательского университета «Высшая школа экономики»
109028, г. Москва, Большой Трехсвятительский пер., д.3
e-mail: al.gavrilenko@gmail.com, spolessky@hse.ru

Аннотация: Предпосылкой проведения анализа построения локальной вычислительной сети (ЛВС) на базе «тонких» и «толстых» клиентов является возрастающая тенденция перехода на «тонкие» клиенты взамен обычных рабочих станций в последние годы. И, если такие плюсы терминальной станции, как низкая цена, простота администрирования, энергоэффективность известны, то сравнение с «толстыми» аналогами по критерию надёжности до сих пор не проводилось.

Ключевые слова: «тонкий» клиент, «толстый» клиент, локальная вычислительная сеть, надёжность, электронные средства, коэффициент готовности.

Для проведения сравнительно анализа необходимо спроектировать два варианта типовых схем ЛВС, состоящие из «тонких» и «толстых» клиентов, соответственно, построенных по топологии «звезда». Для удобства расчетов достаточно смоделировать небольшую корпоративную сеть отдела, состоящую из 20-25 типичных устройств. Выработанные критерии отказов по схеме ЛВС позволят создать структурную схему расчета надежности и приступить к расчетам. В ходе расчетов необходимо определить коэффициенты готовности, наработки на отказ и времени восстановления для обеих ЛВС.

Первым этапом исследования является построение двух ЛВС, состоящих из «толстых» и «тонких» клиентов, соответственно. Допустим, что исследуемый отдел состоит из 20-и рабочих мест и занимается конструкторскими работами, используя соответствующее программное обеспечение (ПО). Таким образом, для построения ЛВС из «толстых клиентах» понадобятся 20 рабочих станций, сервер, принтер. Все устройства объединяются в сеть через коммутатор (см. рис. 1). Для построения ЛВС из «тонких клиентов» понадобятся 20 терминальных станций, сервер, принтер, а так же терминальный сервер, на котором обеспечивается доступ пользователей через «тонкий» клиент к необходимым для работы ресурсам. Все устройства объединены в сеть через коммутатор (см. рис. 2).

Следующим этапом исследования является построение структурной схемы расчета надежности для разных уровней разукрупнения ЛВС по ОСТ 4Г 0.012.242-84 [2]. Для этого необходимо определить, неисправность каких элементов является критичной для работы системы в целом. Допустим, загрузка отдела позволяет оставить 2 рабочих места в резерве. Остальные 18 рабочих мест используются непрерывно в течение всего рабочего дня. Соответственно, отказ более, чем 2-ух рабочих станций/терминальных станций скажется на отказе системы в целом. Так же, недопустим отказ сервера или одного из серверов, для ЛВС из тонких клиентов, недопустим отказ коммутатора.

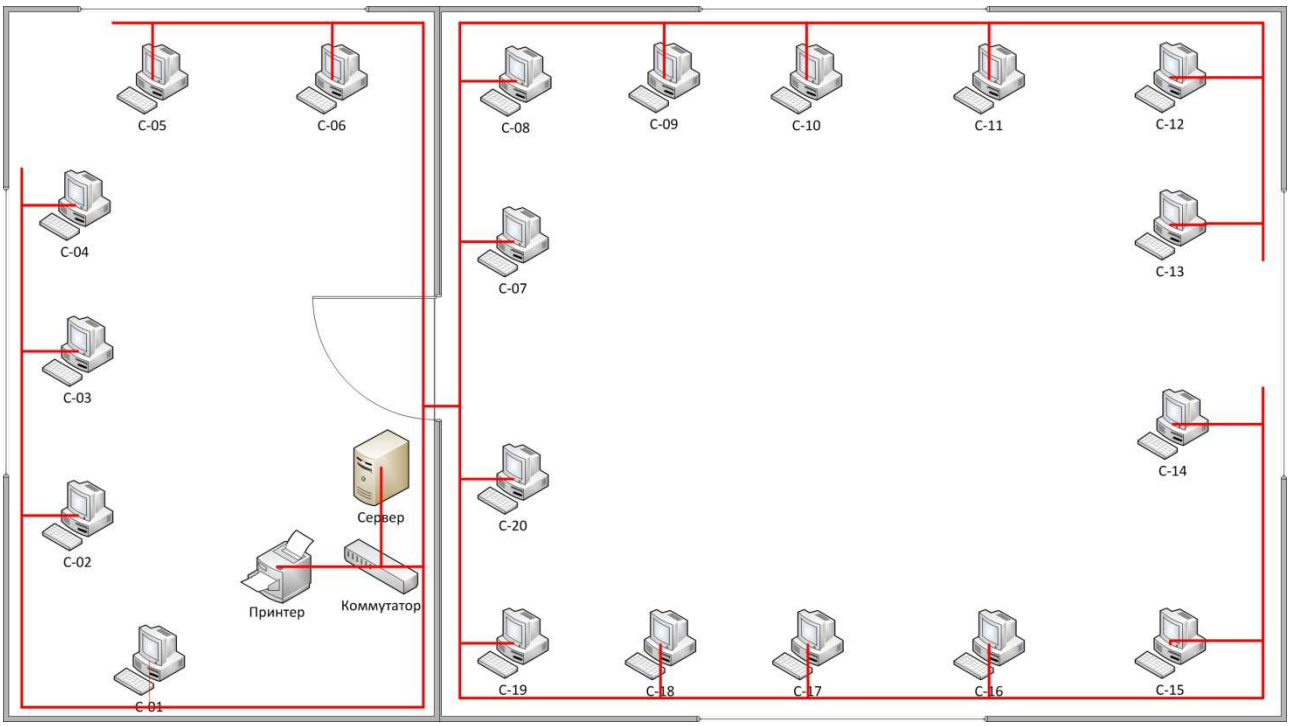


Рис. 1. Схема соединения устройств для ЛВС на базе «толстых» клиентов

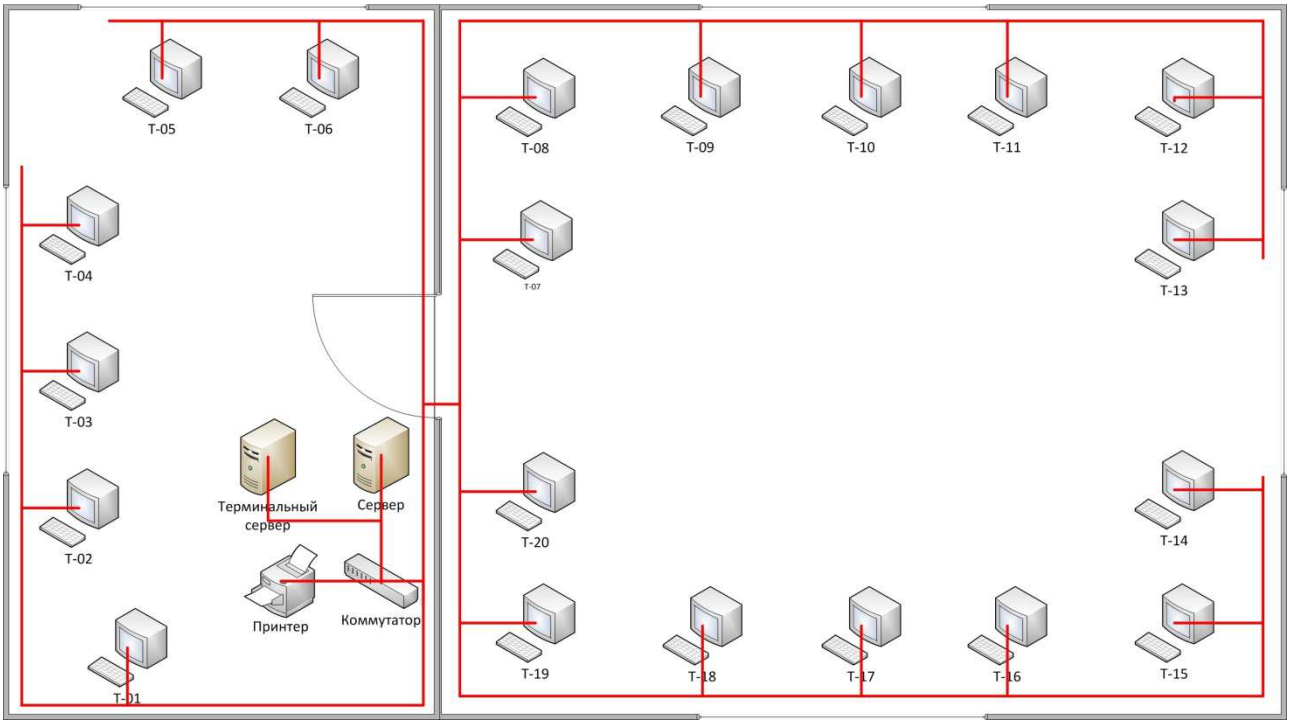


Рис. 2. Схема соединения устройств для ЛВС на базе «тонких» клиентов

Отказ принтера не является критичным, так как задачи отдела напрямую не связаны с непрерывным его использованием. Таким образом, на верхнем уровне, образуется последовательная схема из трех блоков: коммутатор, сервер, рабочая группа, коммутирующая сеть. Однако, отказ коммутирующей сети проводов так не будет включен в расчеты в виду пренебрежимо малой величины интенсивности отказов, а так же вхождения практически одинакового набора соединений в состав обеих ЛВС. На следующем уровне разукрупнения рассматривается рабочая группа, как последовательное соединение n рабочих станций/терминальных станций и параллельно к каждому из них подсоединенных m резервных рабочих

станций/терминальных станций. На следующем уровне рассматривается рабочая станция/терминальная станция, как последовательное соединение элементов, приводящих к отказу работоспособности системы. Структурные схемы надёжности приведены на рис. 3 (для ЛВС на базе «толстых клиентов») и на рис. 4 (для ЛВС на базе «тонких» клиентов). Отказ таких элементов, как: внешнее запоминающее устройство, монитор, клавиатура, мышь, видеокарта, системная плата, процессор, система охлаждения, блок питания, оперативное запоминающее устройство является критичным для системы и ведет к её отказу.

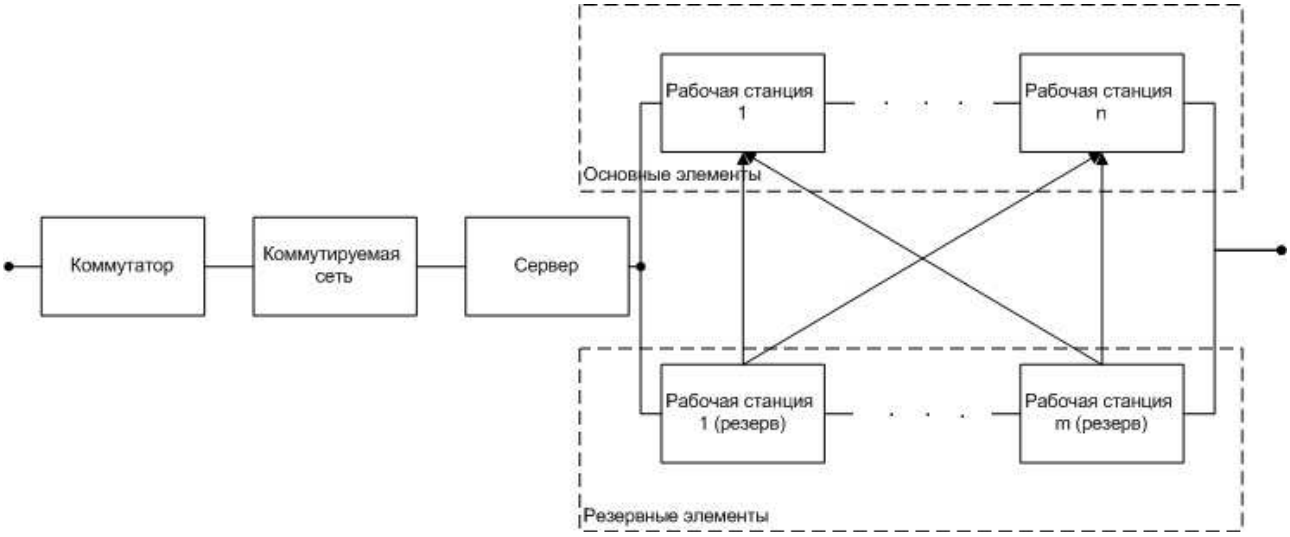


Рис. 3. Структурная схема надёжности ЛВС на базе «толстых клиентов»

Рис. 4. Структурная схема надёжности ЛВС на базе «тонких клиентов»

Расчёт надежности ЛВС производится в два этапа. Сначала рассчитывается (определяется) надёжность элементов в отдельности, затем проводится расчёт надёжности ЛВС в целом. После чего, определяются комплекты запасных частей, инструментов, принадлежностей (ЗИП) для каждой из ЛВС. Схема последовательности действий при расчёте надёжности выполнена в нотациях IDEF0 [13] и представлена на рис. 5.

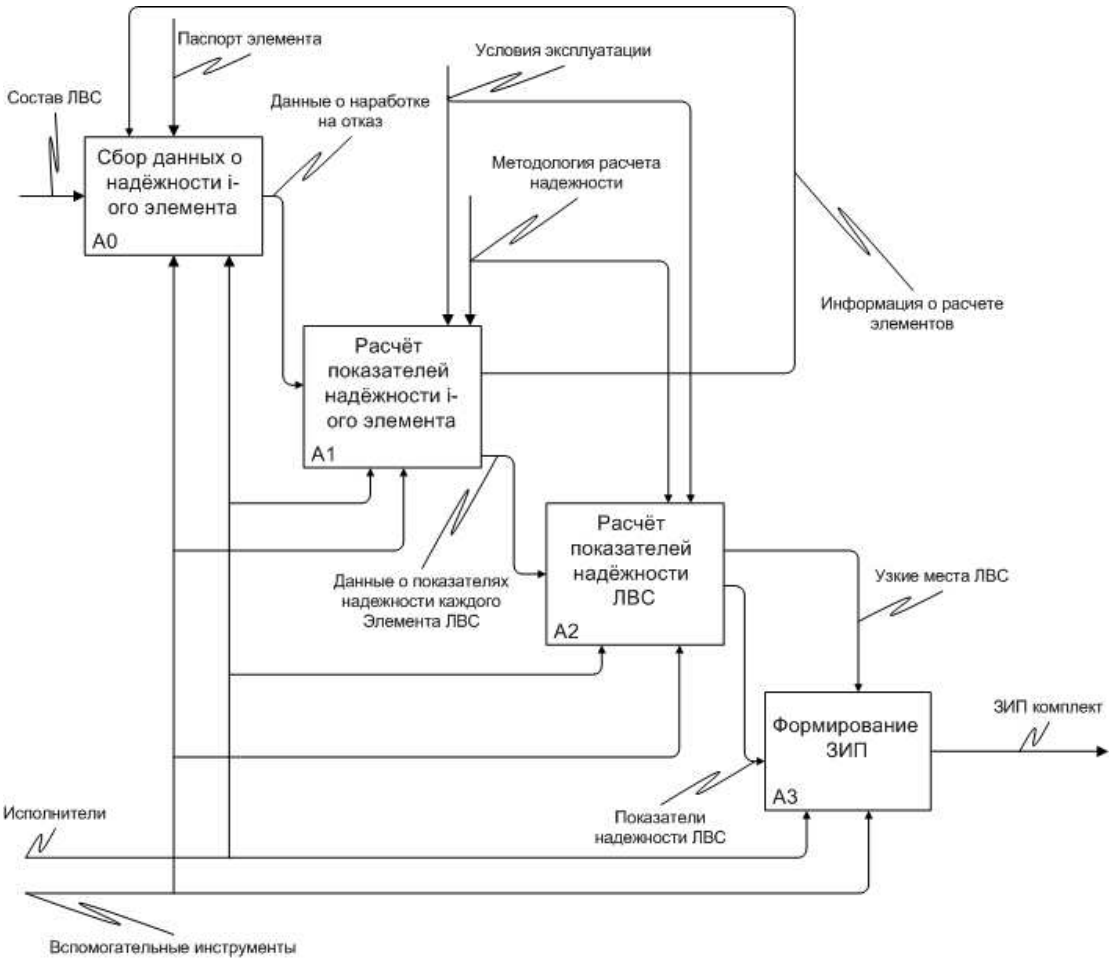


Рис. 5. Схема последовательности расчётов

Расчетные значения наработки на отказ для каждого устройства (элемента ЛВС) возьмем из паспортных данных устройств. Допустим, в состав рабочей станции входят: монитор Samsung 957MB с показателем наработки на отказ 750000 часов [3], мышь и клавиатура HP – по 100000 часов [4], системная плата Intel DP55WP, блок питания и система охлаждения Intel, процессор Intel Core i7 – 100000 часов, 100000 часов, 55000 часов и 550000 часов, соответственно [5], оперативная память Kingston DDR3 KVR1066D3N7/4G 4GB x2 – 500000 часов [6], видеокарта GeForce GT 240 – 55000 часов [7], жесткий диск любого производителя – оценочно 500000 часов [8]. Данные о наработке на отказ для серверов взяты с официального сайта компании Intel и составляют от 45000 до 55000 часов [9]. Что касается коммутатора, то показатели наработки на отказ для модели SRW2024 - 24-Port 10/100/1000 Gigabit Switch составляют 98690 часов [10]. Для подсчета времени восстановления достаточно взять оценочные (экспертные) данные. Так, например, на восстановление рабочей станции могут потребоваться целые сутки, с учетом диагностики, замены частей, перенастройки системы, либо полной переустановки системы. Для восстановления терминальной станции требуется существенно меньше времени из-за меньшего количество компонент и малого времени установки системы (допустим, 3 часа). На восстановление коммутатора требуется в среднем 1 час, на восстановление сервера 4 часа. Спецификация ЛВС на базе «толстых» клиентов представлена в таблице 1. В качестве «тонкого» клиента приведем модель UTC55i Ultra Thin Client фирмы Kingsem. Время наработки на отказ для данной модели составляет не менее 400000 часов [11], что является внушительным показателем относительно данных для элементов рабочей станции. Кроме того, ЛВС на базе «тонких» клиентов включает в

себя терминальный сервер. Его конфигурация совпадает с конфигурацией обычного сервера. Таким образом, время наработки на отказ и время восстановления будет одинаковым.

Далее, проводим расчет в соответствии с ОСТ 4Г 0.012.242-84 по следующей формуле:

$$T_O = \frac{1}{\sum_{i=1}^{n} \frac{1}{T_{Oi}}},$$

где: T_O – наработка на отказ системы, ч.; T_{oi} – наработка на отказ *i*-го отдельного элемента.

Коэффициент готовности рассчитывается по следующей формуле:

$$K_Г = \frac{T_O}{T_O + T_B}.$$

Наработка на отказ совокупности рабочих станций/тонких клиентов, представляющих собой схему с нагруженным резервированием, рассчитывается по следующей формуле:

$$T_O = \frac{1}{\lambda} \frac{\sum_{i=0}^{m} C_N^i (\lambda T_{ВЭ})^i}{C_N^m (\lambda T_{ВЭ})^m},$$

где: m – количество резервных элементов, N-общее количество элементов, λ – интенсивность отказов, $T_{вэ}$ – время восстановления одного элемента (воспользуемся оценочными значениями); при этом время восстановления такой системы рассчитывается по формуле:

$$T_{ВЭ} = \frac{T_{ВЭ}}{m+1}.$$

Расчёт наработки на отказ сервера и всей системы в целом проводится по формуле

$$T_O = \frac{1}{\sum_{i=1}^{n} \frac{1}{T_{Oi}}}.$$

Расчёт производится отдельно для ЛВС из «толстых» и «тонких» клиентов.

Рассчитаем значение наработки на отказ для одной рабочей станции с учётом паспортных данных составных элементов (ссылки приведены выше):

$$T_O ws = \frac{1}{\sum_{i=1}^{n} \frac{1}{T_{Oi}}} = 9643 \text{ часов.}$$

Далее проводим расчёт наработки на отказ для системы рабочих станций (18 основных, 2 резервных):

$$T_O wssys = \frac{1}{\lambda} \frac{\sum_{i=0}^{m} C_N^i (\lambda T_{ВЭ})^i}{C_N^m (\lambda T_{ВЭ})^m} = 2.32 \times 10^4 \text{ часов.}$$

Время восстановления для такой системы рассчитывается следующим образом:

$$T_B = \frac{T_{ВЭ}}{m+1} = 8 \text{ часов.}$$

Далее рассчитывается коэффициент готовности для данной системы из рабочих станций:

$$K_Г wssys = \frac{T_O wssys}{T_O wssys + T_B} = 0.999655$$

Проводим расчёт времени наработки на отказ для одного сервера:

$$T_O serv = \frac{1}{\sum_{i=1}^{n} \frac{1}{T_{Oi}}} = 43662 \text{ часов.}$$

Дальнейшим шагом является проведение расчета времени на работки на отказ для схемы, представленной на рисунке 3. Данные для всех блоков получены выше.

$$T_O 1 = \frac{1}{\sum_{i=1}^{n} \frac{1}{T_{Oi}}} = 13130 \text{ часов.}$$

По аналогии проводятся расчёты для ЛВС на базе терминальных станций, с учетом паспортных данных терминальной станции, согласно которым время наработки на отказ для одной станции составляет 400000 часов [11]. Расчёты для серверов приведены ранее.

$$T_O tssys = \frac{1}{\lambda} \frac{\sum_{i=0}^{m} C_N^i (\lambda T_{ВЭ})^i}{C_N^m (\lambda T_{ВЭ})^m} = 8.78 \times 10^5 \text{ часов:}$$

$$T_B = \frac{T_{ВЭ}}{m+1} = 1 \text{ час;}$$

$$K_Г wssys = \frac{T_O wssys}{T_O wssys + T_B} = 0.999998;$$

$$T_O 2 = \frac{1}{\sum_{i=1}^{n} \frac{1}{T_{Oi}}} = 17520 \text{ часов.}$$

Спецификация ЛВС на базе «тонких» и «толстых» клиентов с указанием рассчитанного времени наработки на отказ, а так же оценочного времени восстановления представлена в таблице 1.

Таблица 1. Спецификация ЛВС

Наименование	На базе «толстых» клиентов (Количество, шт., То/Тв 1 штуки, часов)	На базе «тонких» клиентов (Количество, шт., То/Тв, часов)
Сервер	1, 43662/ 3	1, 43662/ 3
Терминальный сервер	-	1, 43662 / 3
Коммутатор	1, 98690 / 1	1, 98690 / 1
Рабочая станция/ Терминальная станция	20, 9643 / 24	20, 400000 / 3

Коэффициент готовности для ЛВС на базе «толстых» клиентов:

$$K_{Г1} = \prod_{i=1}^{n} K_{Гi} = 0.99955$$

Коэффициент готовности для ЛВС на базе «тонких» клиентов:

$$K_{Г1} = \prod_{i=1}^{n} K_{Гi} = 0.99981$$

Показатель времени восстановления для ЛВС на базе «толстых» клиентов:

$$T_{B1} = T_O 1 \frac{1 - K_{Г1}}{K_{Г1}} = 6.03 \text{ часов.}$$

Показатель времени восстановления для ЛВС на базе «тонких» клиентов:

$$T_{B2} = T_O 2 \frac{1 - K_{Г2}}{K_{Г2}} = 3.26 \text{ часов.}$$

Расчётные значения для показателей наработки на отказ, времени восстановления и коэффициента готовности для двух ЛВС из 20-и машин, с учетом 2-ух резервных представлены в сравнительной таблице 2.

Таблица 2. Сравнение показателей надёжности

№ п/п	Тип ЛВС/Показатель надёжности	T_O, ч	T_B, ч	$K_Г$
1	На базе «толстых» клиентов	1.313×10^4	6.03	0.99955
2	На базе «тонких» клиентов	1.752×10^4	3.26	0.99981

Из полученных результатов видно, что коэффициент готовности (Кг) для ЛВС на базе «толстых» клиентов меньше, чем у аналогичной ЛВС на базе «тонких» клиентов. Время наработки на отказ для ЛВС на базе «тонких» клиентов больше, чем для ЛВС на базе «толстых» клиентов, а показатель времени восстановления ниже. Приведенное сравнение показывает, что система, построенная на базе 20-и «тонких» клиентов, 2 из которых находятся в резерве, окажется надёжнее, чем система, построенная на базе «толстых». Однако, проведённого сравнения может быть не достаточна для заключения. Для более детального анализа, рассмотрим те же самые ЛВС, состоящие из разного количества станций (от 1 до 20). Будем считать, что резервных элементов нет и отказ любой из станций ведёт к отказу системы в целом. На рисунке 6 приведен график, показывающая отношение коэффициента готовности (Кг) к количеству станций в системе. На гистограмме отражены данные как для ЛВС на базе «тонких» клиентов, так и для ЛВС на базе «тонких клиентов».

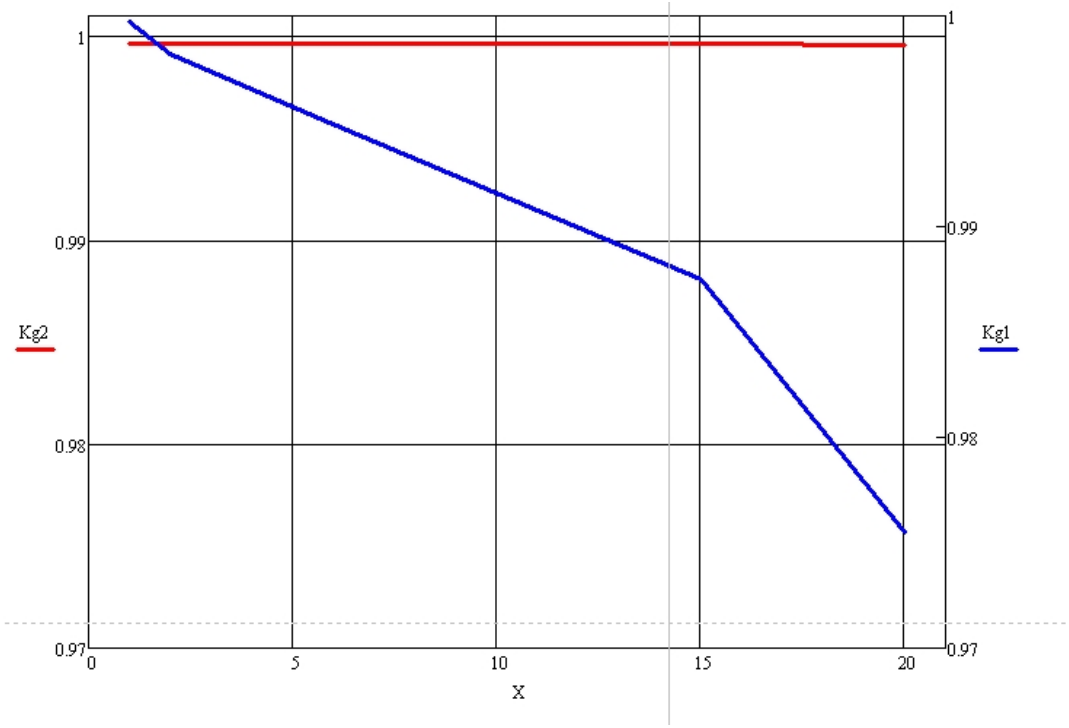


Рис. 6. Отношение коэффициента готовности к числу станций в сети

Kg1 – коэффициент готовности для ЛВС на базе «толстых клиентов», Kg2 - коэффициент готовности для ЛВС на базе «тонких клиентов», X – количество станций. Из рисунка 6 видно, что с увеличением числа станций, готовность ЛВС на базе «толстых» клиентов существенно падает, чего нельзя сказать о ЛВС на базе «тонких» клиентов. Даже при наличии 20-и нерезервированных станций вероятность того, что система на базе «тонких» клиентов будет работоспособна в данный момент времени составляет 99,9996%. График на рисунке 6 показывает, что по сравнению с ЛВС на базе «тонких» клиентов, надёжным является использование ЛВС, состоящей из одной или, максимум, двух «толстых» клиентов. С увеличением числа «толстых» клиентов в группе, Кг демонстрирует планомерное падение и на отметке 20 станций демонстрирует значение 0,9835, что больше, чем на сотую хуже, чем для ЛВС на базе «тонких» клиентов.

Так, как в качестве примера ранее приводилась ЛВС из 18-и основных станций и 2-ух резервных, целесообразно будет проанализировать систему по критерию надёжности с учётом увеличения или уменьшения числа резервных станций. На рисунке 7 приведена гистограмма зависимости коэффициента готовности от числа резервных станций среди 20, входящих в состав ЛВС. Данные приведён для ЛВС на базе «тонких» и «толстых» клиентов.

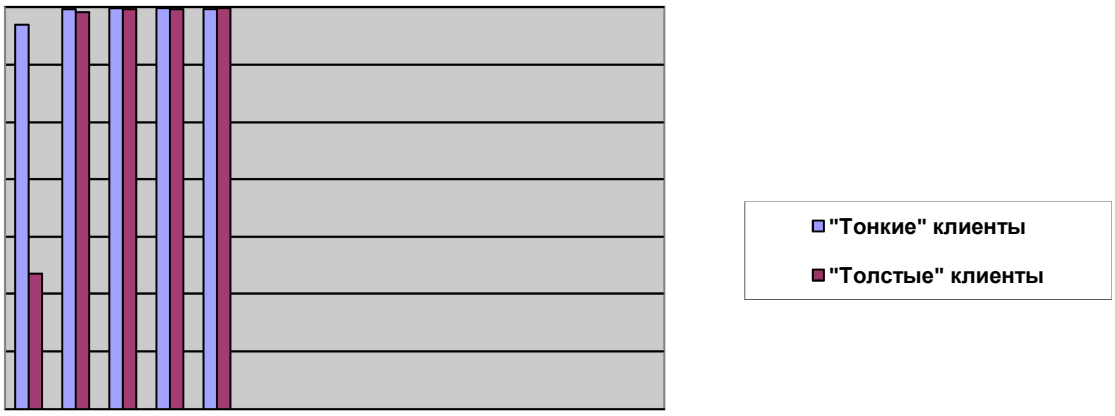


Рис. 7. Отношение коэффициента готовности к числу резервных станций в сети

Гистограмма на рисунке 7 показывает, что при резервировании более 2-ух из 20-и (т.е. более 10%) рабочих станций, сеть на базе «толстых» клиентов демонстрирует такую же высокую «надёжность», как и сеть на базе «тонких» клиентов. Однако, наличие постоянных неиспользуемых станций в качестве резервных далеко не всегда оправдано с экономической точки зрения, причём единичная рабочая станция сама по себе проигрывает в цене аналогичной терминальной. А в отсутствии резервных машин, коэффициент готовности ЛВС на базе «толстых» клиентов составляет всего 0,9755, исходя из чего существует 2,45% вероятность отказа в любой момент времени.

Следующим этапом анализа является расчет запасных комплектов для двух ЛВС. Запасные комплекты необходимы в реалиях современного бизнеса с целью повышения надёжности сети путем своевременной замены вышедшего из строя оборудования. Однако, обратной стороной запасных комплектов являются складские издержки, которые могут привести к моральному устареванию, а так же потере

качества электронных средств. Следовательно, необходимо найти баланс между затратами на хранение и надёжным функционированием системы. В данном случае применимы следующие стратегии пополнения запасов для одиночных комплектов:

1. Периодическое пополнение;
2. Непрерывное пополнение;
3. Пополнение по уровню неснижаемого запаса.

В таблицах 3 и 4 представлена спецификация элементов для ЛВС на базе «тонких» и «толстых» клиентов с указанием необходимых для расчёта параметров.

Таблица 3. Одиночный комплект для ЛВС на базе «толстого» клиента

Наименование	i	m_i	λ_i	α_i	T_{i1}	n_i
Монитор	1	22	$1.33x10^{-6}$	1	8760	2
Мышь	2	22	10^{-5}	1	8760	2
Клавиатура	3	22	10^{-5}	1	8760	2
Коммутатор	4	1	$1.013x10^{-5}$	1	8760	1
Сервер	5	1	$1.754x10^{-5}$	3	8760	1
Системный блок	6	20	$1.037x10^{-4}$	3	336	2

Таблица 4. Одиночный комплект для ЛВС на базе «тонкого» клиента

Наименование	i	m_i	λ_i	α_i	T_{i1}	n_i
Монитор	1	23	$1.33x10^{-6}$	1	8760	2
Мышь	2	23	10^{-5}	1	8760	2
Клавиатура	3	23	10^{-5}	1	8760	2
Коммутатор	4	1	$1.013x10^{-5}$	1	8760	1
Сервер	5	1	$1.754x10^{-5}$	3	8760	1
Терминальный сервер	6	1	$1.754x10^{-5}$	3	8760	1
Терминал	7	20	$2.5x10^{-6}$	3	168	2

где i – Номер составной части;

m_i - Количество основных элементов i –ого типа в объекте;

λ_i - Интенсивность отказов основного элемента i –ого типа в объекте;

α_i - Тип стратегии пополнения элемента i –ого типа в объекте;

T_{i1} - Числовой параметр стратегии пополнения (среднее время ремонта) элемента i –ого типа в объекте;

n_i - Количество запасных элементов i –ого типа в объекте;

α_i = 1 - Периодическое пополнение отказавших элементов.

α_i = 3 - Ремонт(восстановление) отказавших элементов. Одновременно может ремонтироваться любое количество элементов данного типа [12].

Для расчета показателей достаточности такого ЗИП последовательно рассчитаем показатели достаточности комплекта ЗИП-О (запасные части, инструменты, принадлежности - одиночный). Сначала проводится расчёт для ЛВС на базе «толстых» клиентов.

Для элементов с периодическим пополнением (При α_i=1 периодическое пополнение).

$$P_1 = \frac{1}{a} e^{-a} \sum_{j=n+2} (j-n-1) \frac{a^j}{j!},$$

где

$$a = m\lambda T_{П}.$$

Для первой составной части:

$$a = 0.257;$$

$$P_1 = \frac{1}{a} e^{-a} \sum_{j=n+2} (j-n-1) \frac{a^j}{j!} = 5.468 \times 10^{-4};$$

Для второй составной части:

$$a = 0.963;$$

$$P_2 = \frac{1}{a} e^{-a} \sum_{j=n+2} (j-n-1) \frac{a^j}{j!} = 0.014;$$

Для третьей составной части:

$$a = 0.963;$$

$$P_3 = \frac{1}{a} e^{-a} \sum_{j=n+2} (j-n-1) \frac{a^j}{j!} = 0.014;$$

Для четвертой составной части:

$$a = 0.08876;$$

$$P_4 = \frac{1}{a} e^{-a} \sum_{j=n+2} (j-n-1) \frac{a^j}{j!} = 1.202 \times 10^{-3};$$

Для восстанавливаемого элемента (При α_i=3 Ремонт(восстановление) ∞ колвом одновременно ремонтирующихся элементов)

$$P_i = \frac{a^{n+1}}{\left[(n+1)! \sum_{k=0}^{n+1} \frac{a^k}{k!} \right]}; \ (T_{i1} = T_P),$$

где

$$a = m\lambda T_P.$$

Для пятой составной части:

$$a = 0.697;$$

$$P_5 = \frac{a^{n+1}}{\left[(n+1)! \sum_{k=0}^{n+1} \frac{a^k}{k!} \right]} = 0.00068;$$

Для шестой составной части:

$$a = 0.697;$$

$$P_6 = \frac{a^{n+1}}{\left[(n+1)! \sum_{k=0}^{n+1} \frac{a^k}{k!} \right]} = 0.00068;$$

Показатели достаточности рассчитываются по формулам:

$$K = \prod_{i=1}^{N}(1 - P_{i,l_i+1}); \quad K_i = 1 - P_{i,l_i+1};$$

$$\Delta = \frac{1}{\Lambda_{i0}} \sum_{j=1}^{l_i+1} jP_{ij}.$$

Коэффициент готовности комплекта ЗИП-О для ЛВС на базе «толстых» клиентов:

$$K_{ГЗИП-О} = 0.99774;$$

Среднее время задержки в исполнении заявки на запасную часть:

$$\Delta = 40.13;$$

Аналогично проводится расчет для сети на базе «тонких клиентов». Все расчеты совпадут с расчетами для ЛВС на базе «толстых» клиентов, за исключением расчета для терминальных станций, а так же вхождения в систему большего числа устройств ввода/вывода.

Расчет для терминальных станций:

$$a = 0{,}028908;$$

$$P_7 = \frac{1}{a} e^{-a} \sum_{j=n+2} (j - n - 1) \frac{a^j}{j!} = 0{,}00000098\,927585169;$$

Расчет для клавиатуры и мыши:

$$a = 1.0074;$$

$$P_{23} = \frac{1}{a} e^{-a} \sum_{j=n+2} (j - n - 1) \frac{a^j}{j!} = 0.016;$$

Расчет для монитора:

$$a = 0.268;$$

$$P_1 = \frac{1}{a} e^{-a} \sum_{j=n+2} (j - n - 1) \frac{a^j}{j!} = 6.175 \times 10^{-4};$$

Показатели достаточности рассчитываются по формулам:

$$K = \prod_{i=1}^{N}(1 - P_{i,l_i+1}); \quad K_i = 1 - P_{i,l_i+1};$$

$$\Delta = \frac{1}{\Lambda_{i0}} \sum_{j=1}^{l_i+1} jP_{ij}.$$

Коэффициент готовности комплекта ЗИП-О для ЛВС на базе «тонких» клиентов:

$$K_{ГЗИП-О} = 0.99842$$

Среднее время задержки в исполнении заявки на запасную часть:

$$\Delta = 78.62$$

Сравнение характеристик запасных комплектов представлены в таблице 5.

Таблица 5. Сравнение характеристик комплектов ЗИП-О

Тип сети \ параметр	Кг	Δ
На базе «толстых» клиентов	0.99842	78.62
На базе «тонких» клиентов	0.99774	40.13

Из представленных в таблице 5 данных видно, что готовность ЗИП-О для сети на базе «тонких» клиентов выше, а задержки между исполнением заявок на запасную часть меньше, что говорит о меньших затратах на обслуживание ремонта ЛВС, а так же меньшими потребностями в хранении запасных частей на складе.

Подводя итоги анализа, можно утверждать, что более надёжным типом ЛВС является ЛВС, состоящая из «тонких» клиентов. С практической точки зрения это показывает, что отход от привычных корпоративных сетей к технологии «тонкого» клиента является целесообразным с позиции надёжности функционирования системы. Внедрение ЛВС, состоящих из терминальных станций в совокупности с «облачным» ПО существенно сказывается на повышении автоматизации, качества и надежности функционирования предприятий.

Список литературы

1. Козлов Б.В., Ушаков И.А. Справочник по расчету надежности аппаратуры радиоэлектроники и автоматики. – М.: «Советское радио», 1975.
2. ОСТ 4Г 0.012.242-84. Методика расчета показателей надёжности, 1985.
3. Монитор Samsung 957MB. [Электронный ресурс]. Режим доступа: http://www.bjorn3d.com/2003/12/samsung-957mb-19-inch-crt-monitor/#.URgGqx1FU1I.
4. Спецификация USB мыши и USB клавиатуры производства HP. [Электронный ресурс]. Режим доступа: http://h20000.www2.hp.com/bizsupport/TechSupport/Document.jsp?lang=en&cc=us&taskId=120&prodSeriesId=444343&prodTypeId=329290&objectID=c03241057.
5. Спецификация System board DP55WP, Intel Core i7. [Электронный ресурс]. Режим доступа: http://www.intel.ru/content/www/ru/ru/search.html?keyword=mtbf&lstLanguages=ru_RU&topicarea=Support.
6. Спецификация оперативная память Kingston DDR3 KVR1066D3N7/4G 4GB. [Электронный ресурс]. Режим доступа: http://www.kingston.com/dataSheets/KVR1066D3N7_4G.pdf.
7. Обзор видеокарт GeForce. [Электронный ресурс]. Режим доступа: http://www.legitreviews.com/article/1161/3/.
8. MTBF (наработка на отказ) и гарантия в мире компьютеров. Что важно? [Электронный ресурс]. Режим доступа: http://www.hwp.ru/articles/MTBF__28narabotka_na_otkaz_29_i_garantiya_v_mire_kompyuterov_chto_vazhno__64037/?PAGEN_1=2
9. MTBF Intel Servers. [Электронный ресурс]. Режим доступа: http://download.intel.com/support/motherboards/server/sb/s3420gpmtbfcalculationrev10.pdf
10. MTBF figures for switches. [Электронный ресурс]. Режим доступа: http://homecommunity.cisco.com/t5/Switches/Request-MTBF-figures-for-switches/td-p/216907
11. Kingsem UTC55i Ultra Thin Client. [Электронный ресурс]. Режим доступа: http://xn--80akakbjcdfphdy1ackb4nd.xn--p1ai/documentation_pdf/Datasheet_UTCi_rus.pdf.
12. Надежность технических систем: Справочник / Ю.К. Беляев, В.А. Богатырев, В.В. Болотин и др.; Под ред. И.А. Ушакова. – М.: Радио и связь, 1985. – 608 с.
13. Р 50.1.028-2001. Методология функционального моделирования, 2001

Электронный журнал учета отказов радиоэлектронных средств

Цыганов П.А.
Кафедра Радиоэлектроники и телекоммуникаций,
Московский институт электроники и математики национального исследовательского университета «Высшая школа экономики», Москва

Одной из важных обязанностей подразделений предприятий, занимающихся надежностью разрабатываемой аппаратуры, является ведение журнала учета отказов радиоэлектронных средств. В такой журнал заносятся все необходимые сведения об отказе, мерах по его устранению, причинах и результатах устранения и недопущения подобного отказа вновь. На основе журнала также формируются и отчетные документы.

Журнал обычно представляет собой бумажный журнал, с которым неудобно работать. Невозможно быстро найти информацию о конкретном случае отказа, осуществить поиск по заданным критериям. Применение современных информационных технологий [1] диктует необходимость создания электронного журнала учета отказов, который позволит сократить время поиска информации и позволяет улучшить удобство работы с журналом.

Электронный журнал разработан и реализован на основе технологии HTML, языка программирования PHP и MySQL. HTML (от англ. HyperText Markup Language - «язык разметки гипертекста») - стандартный язык разметки документов во Всемирной паутине [2]. Большинство веб-страниц создаются при помощи языка HTML (или XHTML) [3]. Язык HTML интерпретируется браузерами и отображается в виде документа в удобной для человека форме. С помощью HTML выполнен веб-интерфейс журнала учетов отказов, который включает в себя установочную страницу (рисунок 1) журнала и основную страницу журнала.

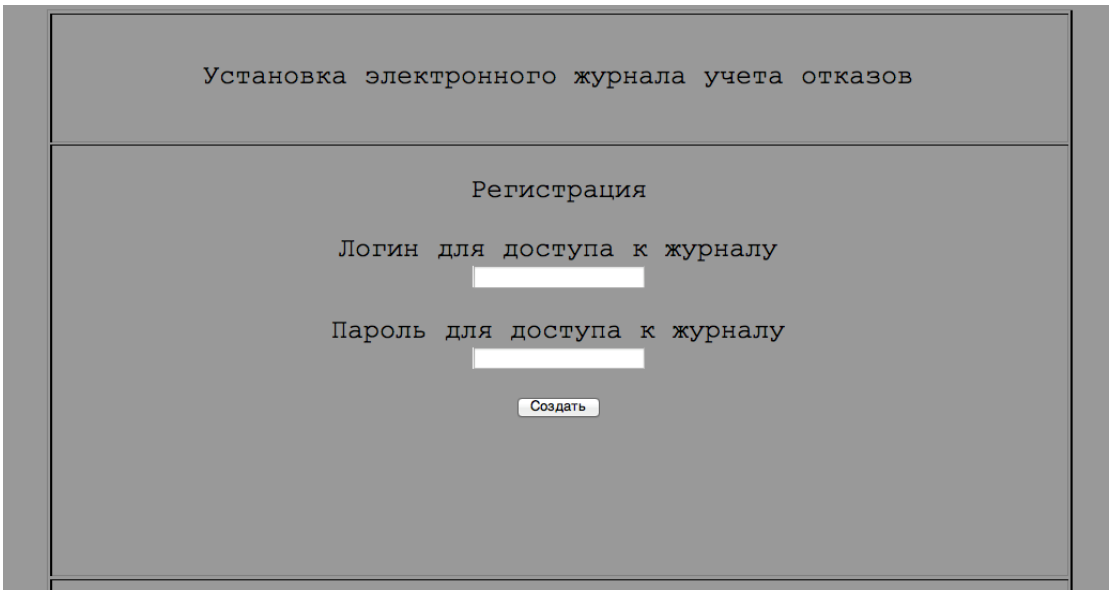


Рис. 1. Установочная страницы журнала

На установочной странице журнала пользователю необходимо указать свой будущий логин и пароль для входа журнал. По завершению ввода информации, программа произведет установку базы данных (БД). Настройка имени базы данных и имени таблиц не предусмотрена. В случае возникновении ошибки происходит вывод

ее описания на экран. В качестве системы управления базой данных (СУБД) используется MySQL. Для удобства администрирования взаимодействие с БД осуществляется через phpMyAdmin. После успешной установки пользователь может приступить к работе с журналом. Для этого на основной странице (рисунок 2) необходимо авторизоваться.

Рис. 2. Фрагмент основной страницы журнала

При успешной авторизации пользователь получит сообщение о том, что он может приступить к работе. Также журнал устанавливает соответствующий флаг. В случае отсутствия флага загрузка форм ввода и вывода информации не осуществляется. Доступа к БД нет.

Пользователь может добавить и новые записи и просмотреть существующие. При нажатии соответствующих кнопок меняется и содержание страницы. Это либо вывод информации из базы данных, так и добавление ее в БД.

В электронном журнале реализован поиск необходимой информации по ключевым словам. Ключевыми словами являются номер акта, дата отказа, дата закрытия наименование изделия, децимальный и заводские номера. Такого количества ключевых слов достаточно для поиска информации по любому отказу в базе данных. Показ результатов выполнен в виде таблицы, что является оптимальным вариантом для подобного рода журналов. Каждая запись выводится после предыдущей. В наименовании столбцов таблицы содержится описание [4].

Кроме того информацию о каждом отказе можно обновлять, так как не вся информация об отказе заносится в журнал сразу.

Для того, чтобы запустить журнал на персональном компьютере, необходимо использовать локальный веб-сервер. Наиболее лучшим решением являются программные решение Denwer и XAMPP. Первый используется в том случае, если используется операционная система семейства Windows. Сразу после установки доступен полностью работающий веб-сервер Apache, работающий на локальном компьютере, на котором может работать неограниченное количество сайтов, что очень эффективно для разработки и отладки сценариев PHP без загрузки его файлов на удаленный сервер. Для запуска практически всех утилит «Денвера» используется приложение Run в подкаталоге /denwer (или /etc) корневого каталога установки «Денвера». Второй вариант будет интересен тем, кто использует операционные

системы семейства Linux и MacOS, так как программный пакет XAMPP является кросплатформеным. Оба пакета бесплатны и не требуют серьезной настройки. После установки они готовы к работе.

Для электронного журнала также написана инструкция, что позволит любому человеку быстро настроить и использовать журнал.

Создание и внедрение электронного журнала позволит существенно сократить время поиска информации, необходимой для прогнозирования надежности радиоэлектронных средств [5]. Заполнение и редактирование журнала происходит более быстро и более легко. Внедрение его на предприятии позволит заменить устаревшие бумажные журналы на электронный вариант.

Список литературы

1. Абрамешин, А.Е. Информационная технология обеспечения надежности сложных электронных средств военного и специального назначения: научное издание. / А.Е. Абрамешин, В.В. Жаднов, С.Н. Полесский; отв. ред. В.В. Жаднов. - Екатеринбург: Изд-во ООО «Форт Диалог-Исеть». - 2012. - 565 с.

2. Фримэн, Э. Изучаем HTML, XHTML и CSS. / Э. Фримэн. - СПб.: Питер. - 2012. - 656 с.

3. Кристиан, Д. PHP и MySQL: создание интернет-магазина. / Д. Кристиан. - СПб.: Вильямс. - 2011. - 632 с.

4. Голицына, О.Л. Основы проектирования баз данных. / О.Л. Голицына. - М.: Форум. - 2012. - 415 с.

5. Жаднов, В.В. Информационные технологии в прогнозировании надежности электронных средств. / В.В. Жаднов. // Информационные технологии в проектировании и производстве. - 2012. - № 1. - с. 20-25.

Метод оценки безотказности структурно-сложных РЭУ

А.Н. Тихменев
Кафедра Радиоэлектроники и телекоммуникаций,
Московский институт электроники и математики национального исследовательского университета «Высшая школа экономики», Москва
E-mail: atihmenev@hse.ru
Моб. тел.: 8-916-267-33-24

Аннотация: В докладе рассмотрены проблемы оценки структурной надежности РЭУ при проектировании. Приводятся описания метода оценки показателей безотказности структурно-сложной РЭУ на основе имитационного моделирования, включающий в себя специализированные модели для описания структуры и алгоритмов функционирования устройств, автоматизированные средства верификации и решения моделей а так же методической обеспечение их использования.

Ключевые слова: показатели безотказности, структурная избыточность, имитационное моделирование.

Исследование осуществлено в рамках Программы фундаментальных исследований НИУ ВШЭ в 2013 году. Одним из часто применяемых методов для обеспечения качества, в том числе и уровня надежности, бортовой аппаратуры является использование реконфигураций различной сложности и на различных уровнях разукрупнения РЭУ [1]. Анализ алгоритмов реконфигурации показал, что структуру бортовых РЭУ часто нельзя свести к набору типовых групп, что одним из наиболее значимых ее особенностей с точки зрения надежности является уникальность конкретной реализации структуры и критериев отказов.

Так как структурная надежность закладывается на ранних этапах проектирования для минимизации риска возвращения к этим этапам необходимо проводить сравнение различных вариантов реализации реконфигураций и резервирования. Критерием выбора с точки зрения обеспечения надежности являются значения показателей безотказности, поэтому важно получить среднее значение ожидаемой структурной надежности, однако существующие методы и средства расчета не позволяют в сжатые сроки провести расчеты различных вариантов реализации структуры с точностью приемлемой для их сравнения.

Наглядной иллюстрацией источника проблем при оценке надежности бортовых РЭУ является структура резервирования бортового интегрированного вычислительного комплекса. При проектировании были учтены особенности функционирования составных частей (СЧ), с целью максимально повысить гибкость структуры и за счет этого добиться повышения «живучести» изделия при отказах. Для цепочек модулей внутри блоков существует такая группа отказов в каждом из них, при возникновении которых другие модули остаются работоспособными и, таким образом, полноценно функционирующая секция может быть построена из «кусочков» разных полукомплектов, то есть при расчете надежности нельзя рассматривать такую группу как последовательное соединение ненагруженных резервов.

Каждый контур резервирования определяет функционирование системы по одной из функций, при этом в итоговую структурную схему надежности различных контуров входят одни и те же (физически) СЧ, что также делает ошибочным рассмотрение БИВК, как несколько последовательно соединенных резервированных

групп. В силу этого неприменимы широко распространенные методы аналитической оценки показателей надежности, базирующиеся на декомпозиции структуры в набор последовательно соединенных типовых групп компонентов. Также неприменим и метод статистического моделирования, основанный на розыгрыше состояния СЧ на момент окончания срока эксплуатации, так как, к примеру, вероятность отказа хранящегося резервного компонента зависит от момента отказа основного:

$$P(t) = e^{-\lambda_x \cdot t_o} \cdot e^{-\lambda_p \cdot (t - t_o)}, \quad (1)$$

где: t_o - время отказа основного компонента, ч.; λ_x - интенсивность отказа в режиме хранения, 1/ч.; λ_p - интенсивность отказа в режиме работы, 1/ч.

Точную оценку ВБР для таких систем можно получить при построении модели расчета по методу перебора гипотез, основанной на формуле полной вероятности. Однако для получения точного результата необходимо учесть все возможный сценарии функционирования системы на протяжении срока эксплуатации, то есть все последовательности и вариации отказов СЧ и реконфигураций не приводящие к отказу РЭУ, которых может быть достаточно много, что сильно затрудняет применение данного метода.

Наличие резервирования на разных уровнях разукрупнения приводит к тому, что интенсивности отказов блоков нельзя считать постоянными (иначе будет допущено занижение показателей надежности), что делает неприменимыми модели и методы, основанные на теории Марковских процессов.

Для реконфигурируемых систем наиболее перспективным методом исследования надежности является имитационное моделирование. Его применение позволяет создать модель, в которой будут учтены все особенности структуры устройства, любые виды зависимости интенсивности отказов и их комбинации во времени. Анализ методов и средств имитационного моделирования показал, что применение имитационного моделирование для расчета надежности ограничено сложностью создания и верификации модели. Так как в модели необходимо учесть все особенности функционирования системы, то для каждой конкретной задачи требуется повторять разработку и верификацию.

Построение модели и ее верификация требует определенных навыков в области имитационного моделирования и представляет собой сложную и длительную работу. Анализ публикаций по данной теме показал, что применение имитационного моделирования, как эффективного инструмента исследования сложных систем для оценки надежности, носит отрывочный характер. Достаточно хорошо освещены модели, связанные с техническим обслуживанием (основанные на теории массового обслуживания), и моделированием физических процессов (деформации, случайные нагрузки, производственные дефекты), однако вопросы надежности сложных, резервированных и реконфигурируемых структур затрагиваются реже и носят отрывочный характер. Рассматриваются только частные случаи, строятся модели отдельных систем, при этом подходе повторно использовать результаты затруднительно, по сути можно лишь повторить разработку моделей и средств верификации для другой системы, ориентируясь на общие подходы предложенные авторами статей.

Несмотря на крайнее разнообразие инструментов имитационного моделирования (одних языков имитационного моделирования более 3 тысяч) в инженерной практике не используются готовые шаблоны, пригодные для построения моделей

реконфигурируемых бортовых РЭУ. На основе этого было принято решения о разработке специализированных моделей и средств их решения для оценки безотказности структурно-сложных РЭУ. Для обеспечения адекватности результатов моделирования модель должна отражать действия реконфигурации на разных уровнях разукрупнения, учитывать зависимость интенсивности отказов от времени, историю функционирования РЭУ и сложные критерии отказов. Для обеспечения требования применения модели к широкому и разнообразному классу бортовых реконфигурируемых РЭУ необходимо гибко описывать состав исследуемого РЭУ, его структуру и критерии отказов. Такие требования можно обеспечить только при моделировании жизненного цикла исследуемого РЭУ от ввода в эксплуатацию до отказа, с учетом всех изменений происходящих в процессе функционирования.

На основе сформулированных требований была предложена форма представления РЭУ как множества отдельных компонентов и возможных событий в системе. Параметрами модели компонента [2] являются состояния и режим функционирования, также модель независимо определяет время, которое компонент пробудет в исправном состоянии при неизменном режиме работы с учетом истории эксплуатации, и информирует о изменениях своего состояния при перемотке модельного времени. При этом разрешается смена режимов и состояний компонента как вследствие случайных процессов (с некоторой интенсивностью) так и при некоторой ситуации в модели (как следствие действий реконфигурации).

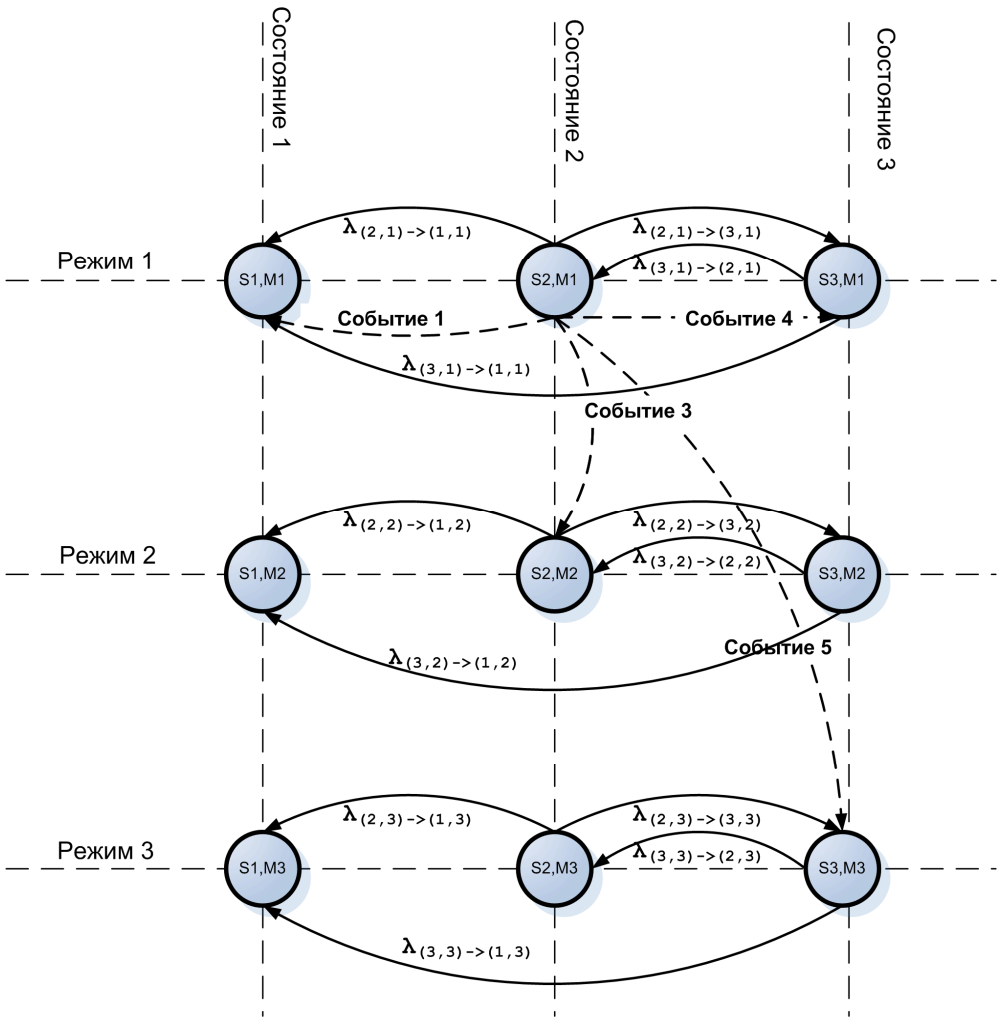


Рис. 1. Диаграмма переходов модели компонентов

Наглядное представление модели одного компонента приведено на рис. 1, это некоторая диаграмма возможных состояний, которые изменяются с определенной интенсивностью (показаны неразрывными стрелками) или вследствие событий (направления обозначены пунктиром).

Таким образом, можно представить множество состояний СЧ, образующих модель РЭУ:

$$U_{R,\tau_0} = \{s_{1,\tau_0}, s_{2,\tau_0}, s_{3,\tau_0}, s_{4,\tau_0} .... s_{n,\tau_0}\}, \quad (2)$$

где: U_{R,τ_0} - множество состояний СЧ РЭУ в момент времени τ_0; каждое s_{i,τ_0} представляет конкретное положение i-й СЧ, на ее диаграмме режимов и состояний.

$$s_{i,\tau_0} \in S_i = \{s_{i0}, s_{i1}, s_{i2} ... s_{im}\}, \quad (3)$$

где: S_i - множество возможных состояний СЧ; $s_{i0}, s_{i1} \ldots$ - конкретные положения на диаграмме состояний (рис. 1).

При этом должно быть известно начальное состояние всех компонентов в модели и их интенсивности переходов в какое-либо другое состояние (как правило, состояние отказа). Тогда путем розыгрыша случайных величин можно определить СЧ, которая первой изменит состояние и ее новое состояние, то есть состояние РЭУ в момент $\tau_{0.9}$:

$$U_{R,\tau_{0.9}} = \{s_{1,\tau_0}, s_{2,\tau_0}, s_{3,\tau_0}, s_{4,\tau_0}, ... s_{k,\tau_{0.9}}, ... s_{n,\tau_0}\}; \quad (4)$$

$$\Delta U_{R,\tau_0} = U_{R,\tau_{0.9}} \setminus U_{R,\tau_0} \cup U_{R,\tau_0} \setminus U_{R,\tau_{0.9}} = \{s_{k,\tau_{0.9}}, s_{k,\tau_0}\}, \quad (5)$$

где: $U_{R,\tau_{0.9}}$ - множество состояний СЧ после смены состояния *k*-ой СЧ; $\Delta U_{R,\tau_0}$ - множество изменений в $U_{R,\tau_{0.9}}$; $s_{k,\tau_{0.9}}, s_{k,\tau_0}$ - текущее и предыдущее состояние *k*-ой СЧ.

В силу того что рассматривается реконфигурируемая РЭУ возможны взаимосвязи в отказах и/или реконфигурации в структуре, то есть окончательно определить состояние РЭУ после отказа (изменения состояния) одной и СЧ можно только через некоторую функцию вида:

$$U_{R,\tau_1} = f_\Delta(U_{R,\tau_0}, \Delta U_{R,\tau_0}), \quad (6)$$

где: f_Δ - функция преобразующая структуру РЭУ и состояние СЧ в соответствии с алгоритмом функционирования на основе предыдущего состояния U_{R,τ_0} и произошедшего события $\Delta U_{R,\tau_0}$.

Функция f_Δ должна однозначно определить состав множества U_{R,τ_1}, то есть состояние всех составных частей РЭУ после изменения состояния одной из них. На основе этого множества можно определить состояние РЭУ в целом, задав для него аналогичную область определения:

$$s_{R,\tau_1} \in S_R = \{s_{R0}, s_{R1}, s_{R2} ... s_{Rm}\}; \quad (7)$$

$$s_{R,\tau_1} = f_s(U_{R,\tau_1}), \quad (8)$$

где: S_R - множество возможных состояний РЭУ в целом; f_s - функция определяющая состояние РЭУ s_{R,τ_1} на основе множества состояний СЧ U_{R,τ_1} в соответствии с критериями отказов РЭУ.

В общем случае набор состояний для РЭУ может быть вида: отказало, работоспособно, исправно. Однако для реконфигурируемой бортовой РЭУ может быть целесообразно вводить другой набор состояний, более четко описывающий специфику функционирования конкретной РЭУ. Иногда необходимо учитывать историю отказов, это легко можно проиллюстрировать на примере

электромеханического компонента - релейного переключателя, его отказы напрямую связаны с кол-вом переключений. В случае если он осуществляет переключение компонентов, то есть его использование связано с действиями реконфигураций то наиболее точно моделировать его отказы через кол-во переключений, то есть при определении его состояния учитывать всю историю жизненного цикла модели РЭУ, тогда функции (7) и (8) приобретает окончательный вид:

$$U_{R,\tau_{i+1}} = f_{\Delta}(U_{R,\tau_i}, \Delta U_{R,\tau_i}, U_{R,\tau_{i-1}}, U_{R,\tau_{i-2}}, ... U_{R,\tau_0}); \quad (9)$$

$$s_{R,\tau_{i+1}} = f_s(U_{R,\tau_{i+1}}), \quad (10)$$

где: U_{R,τ_i} - множество состояний СЧ в *i*-й промежуток времени; s_{R,τ_i} - состояние РЭУ в *i*-й промежуток времени.

Таким образом, можно последовательно определить состояние РЭУ в каждый момент времени и использовать их для определения показателей надежности, таких как интенсивность отказов, вероятность безотказной работы, наработка на отказ, коэффициент готовности и др.

Для улучшения восприятия модели предложено объединение компонентов в группы с индивидуальными критериями отказов (рис. 2).

При этом группа компонентов будет описываться теми же параметрами, что и отдельный компонент (режим, состояние), но без задания интенсивности переходов - ее роль исполняют критерии состояния группы, аналогично виду функцию f_s (рис. 2). Такая организация модели позволяет наиболее точно и близко к действительности отразить структуру исследуемой РЭУ [3], а также разбить функцию f_s на подфункции и тем самым ее упростить без потери точности моделирования реконфигураций.

В качестве решения задачи преобразование действий реконфигураций и критериев отказов в формальный вид, пригодный для автоматизированной обработки и расчета был предложен специализированный язык описания отказов и реконфигураций РЭУ [4].

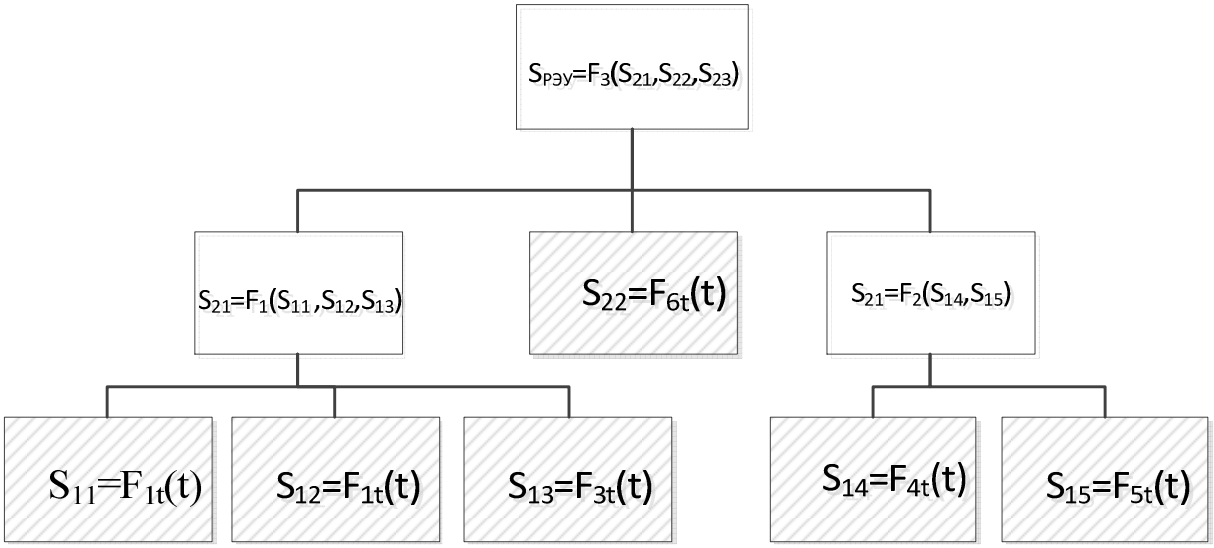


Рис. 2. Представление структуры РЭУ в модели

Его функцией является однозначное задание структуры РЭУ и действий реконфигурации в виде формальной модели, пригодной для автоматизированной обработки и проведения имитационных экспериментов.

Процедурный язык содержит средства для описания моделей компонентов в виде текстовом виде, определяющей вид и интенсивности в диаграмме переходов (в соответствии с концептуальной моделью, представленной на рис. 1). При этом каждый компонент в модели определяется уникальным именем, через которое можно использовать его состояние в вычислительных операциях. В язык вводится несколько основных конструкций - это действие реконфигурации в виде условие-действие и критерий отказа в виде вычислительной процедуры.

В условие события реконфигурации и критерий отказа можно использовать стандартный набор логико-математических операций, как над состояниями компонента так и над переменными, локальными и глобальными, что позволяет учитывать как текущее состояние модели, так и историю возникавших в ходе моделирования событий, то есть формально задать функции f_Δ (9) и f_s (10) и реализовать пошаговое определение состояния РЭУ в каждый момент времени.

Для решения модели применена концепция дискретно-событийного моделирования, то есть функционирование РЭУ представляется как набор событий, между которыми состояние РЭУ и СЧ остается неизменным, а изменения происходят скачкообразно и вызываются изменением состояния одной из СЧ. Для этого был разработан алгоритм имитационного эксперимента, он определяет последовательность вызовов процедур и событий языка и позволяет осуществить корректный перевод модели из одного состояния в другое, то есть программно реализовать функции f_Δ (9) и f_s (10).

Для работы с формальными моделями разрабатывается программное средство, основными модулями которого является компилятор формальной модели, блок верификации и блок выполнения имитационных экспериментов.

В соответствии с принятой практикой имитационного моделирования необходимо перед началом имитационного эксперимента провести верификацию модели на соответствие описанию РЭУ. Для верификации модели в ПС предусмотрена и возможность проведения управляемого эксперимента. В этом случае пользователь сам определяет последовательность отказов компонентов и контролирует состояние модели после каждого отказа. В случае несоответствия реакции модели описанию работы поиск ошибки можно осуществить при помощи анализа лог-файла выполнения шага эксперимента, в котором содержится информация о том, какие условия каких событий реконфигурации были выполнены и какие именно действия с компонентами модели ими были проведены.

Для работы с предложенными моделями и программными средствами разработана инженерная методика оценки надежности бортовой реконфигурируемой РЭУ [7, 8]. В ее основу лег метод имитационного моделирования отказов, способ создания формальной модели реконфигурируемых РЭУ и программное средство имитационного моделирования отказов. Методика представлена в виде на IDEF0-диаграммы на рис. 3.

С соответствии с методикой исходными данными для оценки надежности реконфигурируемой РЭУ является описание алгоритмов функционирования и реконфигурации и данные по интенсивностям отказов СЧ.

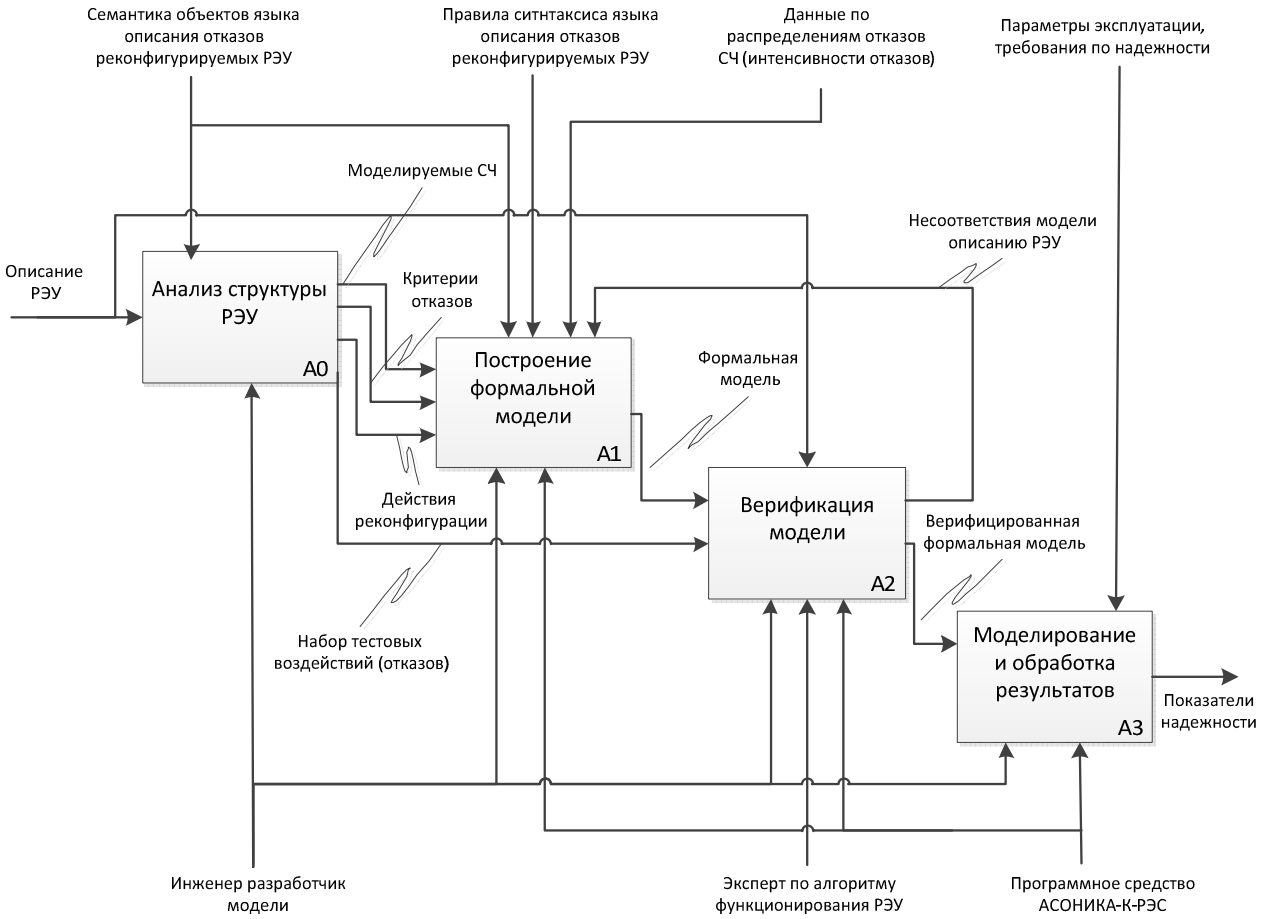


Рис. 3. IDEF0-модель процесса оценки надежности реконфигурируемой бортовой РЭУ

На основе из анализа формируются данные необходимые для построения формальной модели, это списки моделируемых СЧ, списки вводимых групп СЧ, критерии отказов для каждой группы и РЭУ в целом, а также действия реконфигурации в формате «условие-действие». На основе этого создается формальная модель РЭУ которая преобразуется в программную и подвергается верификации.

Верификация производится с использованием программного модуля управляемых экспериментов и представляет собой проверку различных сценариев отказов в составе РЭУ, на каждый из которых модель должна среагировать в соответствии с описанием. При обнаружении несоответствия производится коррекция формальной модели и повторная верификация.

После верификации переходят непосредственно к моделированию, результатом которого являются статистические данные по реализациям отказов РЭУ и статистика по группам СЧ, по которой можно определить какие группы являются проблемными в составе РЭУ с точки зрения надежности и какие наоборот обладают избыточной надежностью.

Экспериментальная проверка разработанных метода, моделей, алгоритма моделирования и программно-методических средств проводилась в два этапа. На первом проводилось моделирование стандартных структур, для которых известны точные аналитические формулы для оценки ВБР. По результатам моделирования были получены значения с ошибкой на уровне 1-3% относительно точных аналитических моделей [4], обусловленной погрешностью конечного числа экспериментов.

На втором этапе проводился расчет показателей надежности сложных реконфигурируемых бортовых РЭУ, одним из которых был БИВК. При этом для верификации моделей привлекались специалисты с предприятий-разработчиков РЭУ, которые экспертно подтверждали соответствие модели алгоритму функционирования. Во всех случаях результаты моделирования давали ожидаемые результаты, которые можно считать ближе к истинным, в сравнении с аналитическими моделями, в которые заведомо были внесены допущения, приводящие к занижению показателей надежности [5, 6]. При этом разница в оценке надежности предложенным методом и аналитическими моделями достигала 15%.

Список литературы

1. Жаднов В.В., Юрков Н.К. Особенности конструирования бортовой космической аппаратуры: учеб. пособие. - Пенза: Изд-во ПГУ, 2012 - 112 с.
2. Жаднов В.В., Тихменев А.Н. Моделирование компонентов электронных средств с реконфигурируемой структурой. / Надёжность и качество: Труды международного симпозиума в 2-х т. // Под ред. Н.К. Юркова. - Пенза: Изд-во ПГУ, 2010 - 1 т. - с. 330-331.
3. Жаднов В.В., Полесский С.Н. Тихменев А.Н. Разработка моделей надежности для проектных исследований надежности радиоэлектронной аппаратуры. / Радиовысотометрия-2010: Сб. трудов Третьей Всероссийской научно-технической конференции. // Под ред. А.А. Иофина, Л.И. Пономарева. - Екатеринбург: Изд-во «Форт Диалог-Исеть», 2010. - с. 200-201.
4. Тихменев А.Н. Язык описания отказов электронных средств с реконфигурируемой структурой. / Науч.-техн. конф. студентов, аспирантов и молодых специалистов МИЭМ: Тез. докл.. - М.: МИЭМ, 2010. - с. 137.
5. Тихменев А.Н., Абрамешин А.Е., Жаднов В.В. Имитационное моделирование в оценке надежности электронных систем с реконфигурируемой структурой для космических аппаратов. / Электромагнитная совместимость и проектирование электронных средств: сб. науч. тр. // Под ред. Л.Н. Кечиева. - М.: МИЭМ, 2012. - с.13-23.
6. Жаднов В.В., Полесский С.Н., Тихменев А.Н. Современные подходы к исследованию безотказности электронных средств циклического применения. // Надежность и качество-2012: труды Международного симпозиума: в 2-х т. // Под ред. Н.К. Юркова. - Пенза: Изд-во ПГУ, 2012 - 1 т. - с. 70-74.
7. Жаднов В.В., Авдеев Д.К., Кулыгин В.Н., Полесский С.Н., Тихменев А.Н. Информационная технология обеспечения надежности сложных электронных средств военного и специального назначения. / Компоненты и технологии, № 6, 2011. - с. 168-174.
8. Абрамешин А.Е., Жаднов В.В., Полесский С.Н. Информационная технология обеспечения надежности электронных средств наземно-космических систем: научное издание. - Екатеринбург: Форт Диалог-Исеть, 2012. - 565 с.

Модуль расчета показателей безотказности резервированных групп для ПК АСОНИКА-К

Авхименко А.М., Колесников С.Н.
Кафедра Радиоэлектроники и телекоммуникаций,
Московский институт электроники и математики национального исследовательского университета «Высшая школа экономики», Москва

Одним из динамично развивающихся программных средств для расчетной оценки надежности радиоэлектронной аппаратуры является программный комплекс (ПК) АСОНИКА-К, который представляет собой визуальную среду обеспечения надёжности. В состав ПК АСОНИКА-К входят специализированные системы (АСОНИКА-К-СИ и АСОНИКА-К-РЭС), предназначенные для расчетной оценки надежности структурно-сложной (резервированной, реконфигурируемой и др.) аппаратуры [1]. Однако, и базовая версия ПК (система АСОНИКА-К-СЧ) давала возможность расчета показателей надежности нескольких типов резервированных групп (для случая общего резервирования) по моделям ОСТ 4Г 0.012.242 [2]. При создании новой (4.12) версии этой системы встала задача расширения номенклатуры таких групп, т.к. опыт эксплуатации системы АСОНИКА-К-СЧ показал необходимость проведения такой экспресс-оценки повышения надежности составных частей (СЧ) путем ведения резервирования.

Анализ выходных данных системы АСОНИКА-К-СЧ и моделей ОСТ 4Г 0.012.242 [2] показал, что необходимо включить в состав модуля расчета показателей надежности резервированных групп следующие модели, приведенные в ОСТ 4Г 0.012.242 [2] для невосстанавливаемой радиоэлектронной аппаратуры (РЭА):

- скользящее нагруженное резервирование. Отказы основных и резервных СЧ обнаруживаются мгновенно. Вероятность отказа переключателя пренебрежимо мала. Интенсивности отказов всех СЧ одинаковые
- скользящее ненагруженное резервирование. Отказы основных и резервных СЧ обнаруживаются мгновенно. Вероятность отказа переключателя пренебрежимо мала. Интенсивности отказов основных СЧ одинаковые; интенсивности отказов резервных СЧ приняты равным нулю
- общее нагруженное резервирование основной СЧ двумя резервными с учетом переключателя. Отказы основной и резервных СЧ обнаруживаются мгновенно. Интенсивность отказов всех СЧ одинаковая. Отказ переключателя делает невозможным подключение любой резервной СЧ вместо отказавшей основной
- мажоритарная схема, работающая по условию «2 из 3» Сигналы с СЧ мажоритарной схемы поступают на элемент М сравнения
- одноярусная иерархическая структура. Общее число СЧ первого яруса (выходных СЧ) - r1. СЧ одинаковые. Допускается отказ не более m выходных СЧ
- мостиковая схема. Интенсивности отказов основных СЧ одинаковые

Алгоритм функционирования модуля расчета показателей надежности резервированных групп приведен на рис. 1.

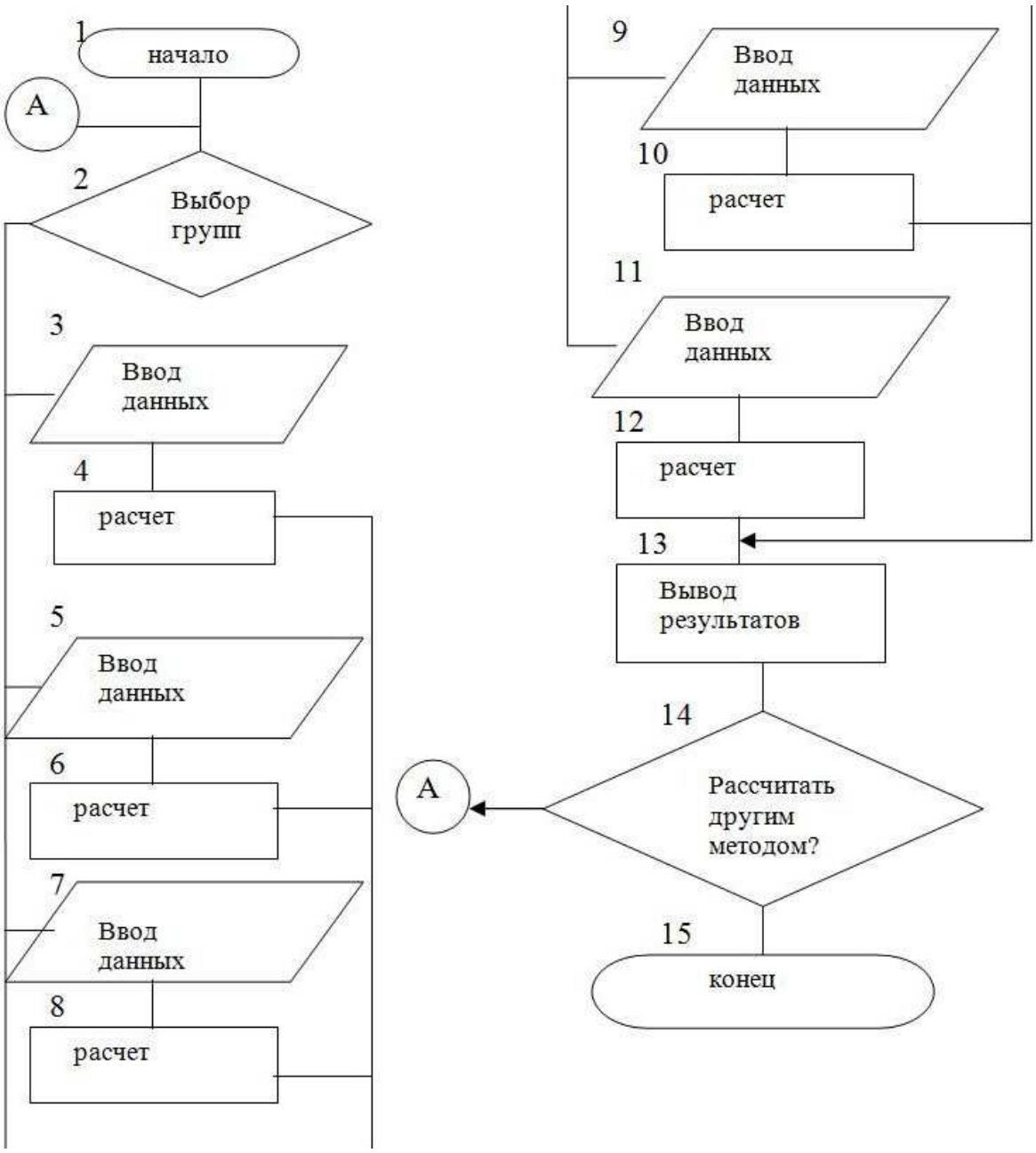


Рис. 1. Алгоритм функционирования модуля расчета показателей надежности резервированных групп

Ниже даны краткие пояснения к блокам алгоритма.

Блок 1. Начало. Вызов модуля расчета резервированных групп из интерфейса системы АСОНИКА-К-СЧ.

Блок 2. Выбор резервированной группы (см. рис. 2).

Блоки 3-12. Ввод данных для расчета и расчет показателей безотказности резервированной группы.

Блок 13. Вывод результатов расчетов (см. рис 3, 4).

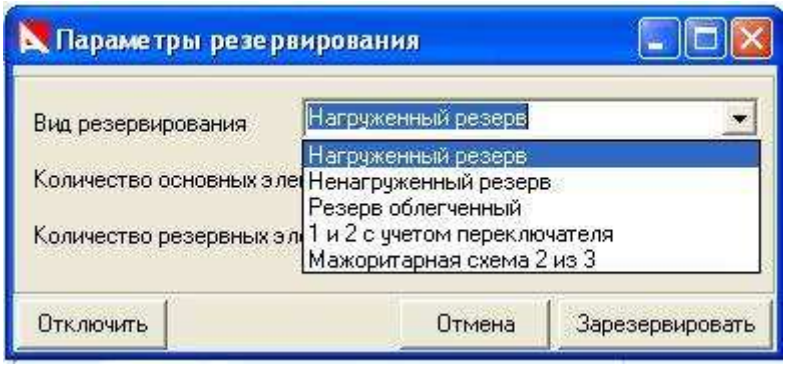


Рис. 2. Система АСОНИКА-К-СЧ: Выбор резервированной группы

Рис. 3. Система АСОНИКА-К-СЧ: Результаты расчета группы ненагруженного резервирования

Рис. 4. Система АСОНИКА-К-СЧ: Результаты расчета группы «мажоритарная схема» с учетом режима хранения

Блок 14. Переход к Блоку 2 для расчета по другому типу резервированной группы.

Блок 15. Окончание. Возврат в интерфейс системы АСОНИКА-К-СЧ.

Модуль расчета показателей надежности резервированных групп был реализован на языке программирования высоко уровня C++ [3].

Тестирование модуля осуществлялась как автономно, так и в составе системы АСОНИКА-К-СЧ (версии 4.12) путем сравнения результатов автоматизированных и «ручных» расчетов.

Список литературы

1. Абрамешин А.Е., Жаднов В.В., Полесский С.Н. Информационная технология обеспечения надежности электронных средств наземно-космических систем: Научное издание. / Отв. ред. В.В. Жаднов. - Екатеринбург: Форт Диалог-Исеть, 2012. - 565 с.

2. ОСТ 4Г 0.012.242-84. Аппаратура радиоэлектронная. Методика расчета показателей надежности.

3. Захарова С.С. Основы программирования на C++: Учебное пособие. / М.: МИЭМ, 2012. - 78 с.

Исследование модели интенсивности отказов пружин скручивания

Монахов М.А., Фокин В.М., Лушпа И.Л.
Кафедра Радиоэлектроники и телекоммуникаций,
Московский институт электроники и математики национального исследовательского университета «Высшая школа экономики», Москва

Исследование осуществлено в рамках Программы фундаментальных исследований НИУ ВШЭ в 2013 году. Наряду с электрорадиоизделиями (ЭРИ) на безотказность радиоэлектронной аппаратуры (РЭА) влияют механические элементы (МЭ). Поэтому при создании новой (4.12) версии программного комплекса (ПК) АСОНИКА-К в его базовую версию (систему АСОНИКА-К-СЧ) было решено ввести возможность расчета надежности РЭА с учетом МЭ. В данной работе приведены результаты анализа математической модели интенсивностей отказов группы «Пружины скручивания» представленной в американском стандарте NSWC-2011/LE10 [1].

Типовая пружина скручивания класса «Пружины» показана на рис. 1.

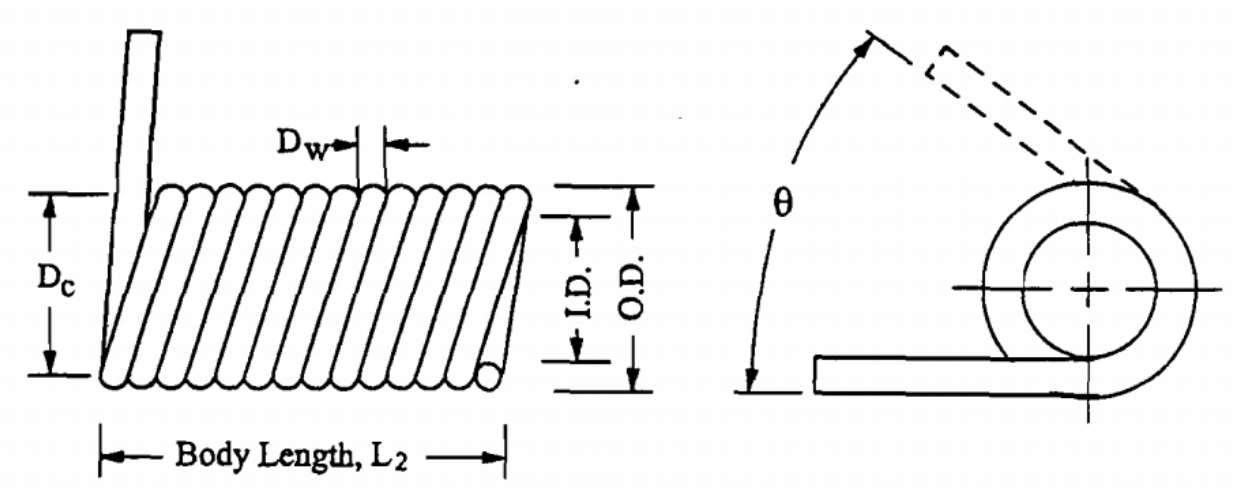


Рис. 1. Пружина скручивания

Интенсивность отказов такой пружины зависит от нагрузки, прикладываемой к пружине и релаксацией, обеспечиваемой материалом.

Математическая модель интенсивности отказов такой пружины, приведенная в стандарте NSWC-2011/LE10 [1], имеет вид:

$$\lambda_{SP} = \lambda_{SP,B} \times C_E \times C_{DW} \times C_{DC} \times C_N \times C_Y \times C_L \times C_{CS} \times C_R \times C_M \qquad (1)$$

где: $\lambda_{SP,B}$ - базовая интенсивность отказов пружины; C_E, C_{DW}, C_N, C_Y, C_L, C_K, C_{CS}, C_R, C_M - поправочные коэффициенты.

Значение коэффициента C_E, учитывающего величину модуля эластичности материала, рассчитывается по формуле:

$$C_E = \left(\frac{E_M}{28.5}\right)^3, \qquad (2)$$

где: G_E - модуль эластичности материала пружины.

Значение коэффициента C_{DW}, учитывающего величину диаметра нити, рассчитывается по формуле:

$$C_{DW} = (\frac{D_W}{0.085})^3, \quad (3)$$

где: D_W - диаметр нити.

Значение коэффициента C_{DC}, учитывающего величину диаметра витка, рассчитывается по формуле:

$$C_{DC} = (\frac{0.58}{D_C})^6, \quad (4)$$

где: D_C - средний диаметр пружины.

Значение коэффициента C_N, учитывающего количество активных витков, рассчитывается по формуле:

$$C_N = (\frac{14}{N_A})^3, \quad (5)$$

где: N_A = количество активных витков.

Значение коэффициента C_Y, учитывающего сопротивляемость растяжению материала пружины, рассчитывается по формуле:

$$C_Y = \left(\frac{190}{T_S}\right)^3, \quad (6)$$

где: T_S - предел прочности материала на разрыв.

Значение коэффициента C_L, учитывающего воздействие закрутки пружины, рассчитывается по формуле:

$$C_L = (\frac{\theta}{61.2})^3, \quad (7)$$

где: θ - угол закрутки пружины.

Значение коэффициента C_K, учитывающего воздействие сжатия пружины, рассчитывается по формулам:

$$C_K = (\frac{K_W}{1.219})^3, \; K_W = \frac{4r-1}{4r-4} \div \frac{0.615}{r}, \; r = \frac{D_C}{D_W}, \quad (8)$$

где: D_C - диаметр витка; D_W - диаметр нити.

Значение коэффициента C_{CS}, учитывающего воздействие частоты нагружения пружины, определяется как:

CR ≤ 30 цикл/минута, C_{CS} = 0.1

30 <CR <300 цикл/минута, $C_{CS} = \frac{CR}{300}$ (9)

CR > 300 цикл/минута, $C_{CS} = \left(\frac{CR}{300}\right)^3$

где: CR - частота нагружения пружины.

Значение коэффициента C_R, учитывающего влияние коррозии, принимается равным 1,0 при наличии антикоррозийного покрытия, в противном случае значение

C_R определяется исходя из коррозионной стойкости материала пружины к воздействию окружающей среды.

Значение коэффициента C_M, учитывающего качество производственного процесса, принимается равным 1,0 при наличии у производителя аттестованной системы менеджмента качества, в противном случае значение C_M принимается равным значению, предоставленному производителем.

Анализ моделей (1)-(9), в соответствии с принятой классификацией [4], позволил сформировать следующую классификацию параметров и коэффициентов модели интенсивности отказов для группы «Пружины сжатия», приведенную в табл. 1.

Таблица 1. Классификация параметров и коэффициентов модели интенсивности отказов

Обозначение	Наименование	Значение	Ед. измерения	Примечание
1	2	3	4	5
Параметры ТУ				
-	Типономинал пружины	-	-	ТУ на пружину
D_w	Диаметр нити	-	in	ТУ на пружину
D_c	Диаметр витка пружины	-	in	ТУ на пружину
N_a	Количество активных витков	-	-	ТУ на пружину
L_2	Длина пружины	-	in	ТУ на пружину
C_R	Коррозионное покрытие	-	-	ТУ на пружину
C_M	Качество производственного процесса	-	-	ТУ на пружину
-	Марка материала пружины	-	-	ТУ на пружину
Параметры режима применения				
CR	Частота нагружения пружины	-	cycles/min	ТЗ на аппаратуру
θ	Угол закрутки пружины	-	revolutions	-
Эмпирические коэффициенты				
$\lambda_{SE,\,B}$	Базовая интенсивность отказов пружины	Из БД	h-1	NSWC-2011/LE10
G_E	Модуль эластичности	Из БД	lb/in2	ТУ на материал
T_S	Предел прочности	Из БД	lb/in2	ТУ на материал

На основании данной таблицы ведется разработка базы данных для этого класса [5-7]. Пользователю необходимо будет выбрать типономинал пружины (номер ТУ), после чего определиться все параметры из ТУ, необходимые для расчета. После этого останется только ввести данные из ТЗ и характеристики режима применения. Если в БД нет необходимых данных, то пользователю будет выведено окно, в котором необходимо внести все данные «вручную».

Список литературы

1. Маркин, А.В. Методы оценки надёжности элементов механики и электромеханики электронных средств на ранних этапах проектирования. / А.В. Маркин, С.Н. Полесский, В.В. Жаднов. // Надёжность. - 2010. - № 2. - с. 63-70.

2. Полесский, С. Обеспечение надёжности НКРТС. / С.Н. Полесский, В.В. Жаднов. - LAMBERT Academic Publishing, 2011. - 280 с.

3. NSWC-2011/LE10. Handbook of Reliability prediction Procedures for Mechanical Equipment.

4. Zhadnov, V. Methods and means of the estimation of indicators of reliability of mechanical and electromechanical elements of devices and systems. / V. Zhadnov. // Reliability: Theory & Applications: e-journal. - 2011. - Vol. 2, No 4. - p. 94-102.

5. Монахов, М.А. Разработка базы данных по характеристикам надежности механических элементов. / М.А. Монахов. // Наука и образование в развитии промышленной, социальной и экономической сфер регионов России. V Всероссийские научные Зворыкинские чтения: сб. тез. докл. Всероссийской межвузовской научной конференции. Муром, 1 февр. 2013 г. - Муром: Изд.-полиграфический центр МИ ВлГУ, 2013. [Электронный ресурс]: 1 электрон. опт. диск (CD-ROM).

6. Монахов, М.А. Разработка баз данных для расчета интенсивности отказов механических элементов в системе АСОНИКА-К-СЧ. / М.А. Монахов. // Сборник трудов VI Международной научно-практической конференции учащихся и студентов: в 2 ч. - Протвино: Управление образования и науки, 2013. - ч. 2.

7. Монахов, М.А. Разработка базы данных программного комплекса АСОНИКА-К для расчета надежности радиоэлектронной аппаратуры с учетом механических элементов. / М.А. Монахов. // Научно-техническая конференция студентов, аспирантов и молодых специалистов МИЭМ НИУ ВШЭ. Тезисы докладов. - М.~: МИЭМ НИУ ВШЭ, 2013.

Сравнительный анализ методов оценки надежности полупроводниковых интегральных микросхем

Жаднов В.В.
Кафедра Радиоэлектроники и телекоммуникаций,
Московский институт электроники и математики национального исследовательского университета «Высшая школа экономики», Москва

Исследование осуществлено в рамках Программы фундаментальных исследований НИУ ВШЭ в 2013 году. Интегральные микросхемы (ИС) широко применяются в современных электронных средствах (в.ч. и в бортовой электронной аппаратуре для диагностики и мониторинга технического состояния распределенных систем) и в значительной степени определяют их показатели надежности.

По классификации ГОСТ 27.003 [1] ИС относится к изделиям общего назначения вида I (высоконадежное комплектующее изделие межотраслевого применения), непрерывного длительного применения, невосстанавливаемое, необслуживаемое, переход которого в предельное состояние не ведет к катастрофическим последствиям, изнашиваемое, стареющее при хранении. Для таких изделий нормируются следующие показатели надежности:

- интенсивность отказов - λ
- средний ресурс до списания (полный) - $T_{р.ср.сп}$
- средний срок сохраняемости - $T_{с.ср}$

Поскольку свойство надежности проявляется при эксплуатации ИС, то для оценки его показателей на ранних этапах проектирования бортовой электронной аппаратуры для диагностики и мониторинга технического состояния распределенных систем применяются расчетные методы. Для обеспечения достоверности и воспроизводимости результатов расчетов, в соответствии с требованиями ГОСТ 27.301 [2] и РДВ 319.01.20 [3], эти методы стандартизованы.

Так, ГОСТ 27.301 [2] для расчета интенсивностей отказов серийно выпускаемых и новых электронных и электротехнических элементов разных типов (в т.ч. ИС) с учетом уровня их нагруженности, качества изготовления, областей применения аппаратуры, в которой используются элементы, рекомендует методики, приведенные в американском военном стандарте MIL-HDBK-217 [4] и в отечественных справочниках по надежности электрорадиоизделий [5, 6].

В справочнике «Надежность ЭРИ» [5] приведены математические модели для расчета эксплуатационной интенсивности отказов отдельных групп (типономиналов) интегральных микросхем (см. табл. 1).

Модель (1), а также значения базовой интенсивности отказов группы интегральных микросхем ($\lambda_{б.с.г}$) используются при расчете эксплуатационной интенсивности отказов $\lambda_э$ всех типономиналов интегральных микросхем. Модель (2) используется при расчете эксплуатационной интенсивности отказов $\lambda_э$ тех типономиналов интегральных микросхем, которые имеют повышенные значения интенсивности отказов.

Значения коэффициента $К_{с.т}$, учитывающего сложность (степень интеграции) ИС и температуру окружающей среды рассчитываются по модели:

$$K_{C.T} = A \cdot e^{B \cdot (t+273)},$$

где: A - коэффициент, зависящий от группы ИС и количества элементов, бит (для ЗУ); B - постоянный коэффициент модели; t - температура окружающей среды.

Математические модели $\lambda_э$, приведенные в справочнике «Надежность ЭРИ ИП» [5] аналогичны методам стандарта MIL-HDBK-217F [4] - см. табл. 2.

Таблица 1. Математические модели эксплуатационной интенсивности отказов ИС

№ п.п.	Группа ИС	Вид математической модели	Примечание
1	2	3	4
1	Микросхемы интегральные полупроводниковые цифровые: логические, арифметические, микропроцессоры и микропроцессорные комплекты, пограммируемые логические матрицы, регистры сдвига, базовые матричные кристаллы и др.	$\lambda_э = \lambda_{б.с.г} \cdot К_{с.т} \cdot К_{корп} \cdot К_V \cdot К_э \cdot К_{пр}$ или $\lambda_э = \lambda_б \cdot К_{с.т} \cdot К_{корп} \cdot К_V \cdot К_э \cdot К_{пр}$	(1) (2)
2	Оперативные запоминающие устройства (ОЗУ)		
3	Постоянные запоминающие устройства (ПЗУ, ППЗУ, РПЗУ)		
4	Микросхемы интегральные полупроводниковые аналоговые		

Таблица 2. Математические модели эксплуатационной интенсивности отказов ИС

№ п.п.	Группа ИС	Вид математической модели
1	2	3
1	Микросхемы цифровые	$\lambda_э = (\lambda_{кр} \cdot К_t + \lambda_{корп} \cdot К_э) \cdot К_{пр}$ (3)
2	Микросхемы аналоговые	
3	Микросхемы памяти	
4	Программируемые логические интегральные схемы	
5	Микропроцессоры	

Значение $\lambda_{кр}$ зависит от сложности ИС. Значения коэффициента $К_t$, учитывающего температуру окружающей среды рассчитываются по модели:

$$K_t = 0{,}1 \cdot e^{\left[-\frac{E_a}{k} \cdot \left(\frac{1}{T_j + 273} - \frac{1}{298}\right)\right]},$$

где: E_a - энергия активации; k - постоянная Больцмана; T_j - температура кристалла ИС.

Область применения моделей (1), (2) и (3) ограничена сложностью ИС, для которой приведены значения коэффициентов А (или $\lambda_{кр}$). Для сверхбольших КМОП ИС, сложность которых выходит за ограничения этих моделей, в стандарте MIL-HDBK-217F [4] приведена следующая математическая модель интенсивности отказов:

$$\lambda_P = \lambda_{BD} \cdot \pi_{MGF} \cdot \pi_T \cdot \pi_{CD} + \lambda_{BP} \cdot \pi_E \cdot \pi_Q \cdot \pi_{PT} + \lambda_{EOS}. \quad (4)$$

В модели (4) первое слагаемое характеризует интенсивность отказов кристалла, второе - интенсивность отказов корпуса, а третье - стойкость СБИС к воздействию электростатического разряда (ЭСР).

Значения коэффициента π_{CD}, учитывающего сложность СБИС, рассчитываются по модели:

$$\pi_{CD} = \left[\frac{A}{0{,}21} \cdot \left(\frac{2}{X_S} \right)^2 \cdot 0{,}64 \right] + 0{,}36 \quad (5)$$

где: A - площадь кристалла; Xs - топологический размер.

Однако, как следует из стандарта MIL-HDBK-217F [4], эмпирические коэффициенты модели (5) получены для СБИС с топологическим размером не менее 0,8 мкм.

Общим свойством моделей (1)-(4) является то, что они позволяют рассчитать $\lambda_э$ периода «нормальной работы» (см. рис. 1), в предположении, что справедлива экспоненциальная модель отказов:

$$F(t) = 1 - e^{-\lambda \cdot t}. \quad (6)$$

Рис. 1. Зависимость λ от времени

Вместе с тем, в ГОСТ 27.005 [7] наряду с моделью (6) рекомендован еще целый ряд моделей (логнормальное, Вейбулла и др.), обоснование выбора которых может основываться не только на статистических критериях согласия, но и на анализе процессов, протекающих в конструкции ИС и приводящих к отказам (на методах «физики отказов»).

Характерными механизмами отказов КМОП ИС являются [8]:

- пробой подзатворного диэлектрика;
- электромиграция;
- генерация «горячих носителей»;
- межслойные дефекты изоляционных диэлектриков;
- дефекты p-n переходов;
- дефекты маскирования;
- образование пустот.

Эти механизмы обусловленными процессами деградации, которые могут быть аппроксимированы марковскими процессами диффузионного типа с немонотонными реализациями (рис. 2).

Для таких процессов в ГОСТ 27.005 [7] приведена модель отказов (DN-распределение) вида:

$$F(t) = \Phi\left(\frac{\alpha \cdot t - 1}{\nu \cdot \sqrt{\alpha \cdot t}} \right) + \exp\left(\frac{2}{\nu^2} \right) \cdot \Phi\left(-\frac{\alpha \cdot t + 1}{\nu \cdot \sqrt{\alpha \cdot t}} \right). \quad (7)$$

где: Ф - функция нормированного нормального распределения; а - средняя скорость процесса деградации (изменения определяющего параметра X); ν - коэффициент вариации процесса деградации (скорости деградации).

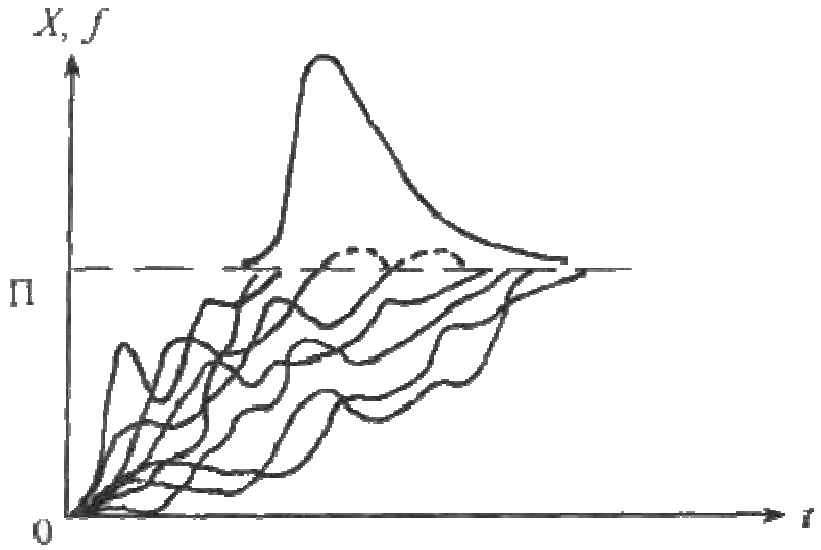


Рис. 2. Реализации случайного процесса изменения определяющего параметра X (П - предельное значение X) и плотность вероятности наработки f

Однако, как следует из (7), эта модель может применяться для «обобщенного» процесса деградации ИС (обобщенного определяющего параметра). В «Appendix B» стандарта MIL-HDBK-217F [4] приведена математическая модель интенсивности отказов КМОП ИС, учитывающая каждый из отказовых механизмов:

$$\lambda_P(t) = \lambda_{OX}(t) + \lambda_{MET}(t) + \lambda_{HC}(t) + \lambda_{CON}(t) + \lambda_{MIS}(t) + \lambda_{PAC} + \lambda_{EOS}, \quad (8)$$

где: $\lambda_{OX}(t)$ - интенсивность отказов кристалла, обусловленная эффектом пробоя подзатворного диэлектрика; $\lambda_{MET}(t)$ - интенсивность отказов кристалла, обусловленная эффектом электромиграции; $\lambda_{HC}(t)$ - интенсивность отказов кристалла, обусловленная эффектом горячих носителей; $\lambda_{CON}(t)$ - интенсивность отказов кристалла, обусловленная эффектом точечных дефектов; $\lambda_{MIS}(t)$ - интенсивность отказов кристалла, обусловленная прочими эффектами; λ_{PAC} - интенсивность отказов корпуса; λ_{EOS} - интенсивность отказов ИС, учитывающая стойкость к воздействию ЭСР.

Подробное описание слагаемых модели (8) и методов их расчета приведено в RADS-TR-89-177 [9]. Кроме того модель (4) представляет собой упрощенную модель (8), позволяющую получить верхнюю оценку значения λ. Как следует из (8), λ является функцией времени, поэтому в этом случае в качестве модели отказов (функции распределения наработки) следует использовать общую формулу:

$$F(t) = 1 - e^{-\int_0^t \lambda_p(t)dt}. \quad (9)$$

Очевидно, что использование моделей (8) и (9) целесообразно при проектировании СБИС, но мало пригодно при расчетах надежности аппаратуры, что обусловлено не столько сложностью и трудоемкостью расчетов, сколько практическим отсутствием в нормативно-технической документации (ТУ и Data Sheet) данных о конструктивных, технологических и др. параметрах СБИС, необходимых для расчетов слагаемых модели (8), за исключением слагаемых λ_{PAC} [10] и λ_{EOS} [11].

Это подтверждается и методологией 217Plus™ [12] прогнозирования безотказности аппаратуры. Так, в RIAC-HDBK-217Plus [12] приведены модели интенсивности отказов ИС (отдельно для ИС в герметичных и негерметичных

корпусах). Например, модель интенсивности отказов ИС в негерметичном (пластиковом) корпусе имеет вид:

$$\lambda_P = \pi_G \cdot (\lambda_{OB} \cdot \pi_{DCO} \cdot \pi_{TO} + \lambda_{EB} \cdot \pi_{DCN} \cdot \pi_{RNT} + \lambda_{TCB} \cdot \pi_{CR} \cdot \pi_{DT}) + \lambda_{SJB} \cdot \pi_{SJDT} + \lambda_{EOS}. \qquad (10)$$

В модели (10) первое слагаемое характеризует интенсивность отказов в режиме работы, ожидания и термоциклирования, второе - интенсивность отказов выводов, а третье - стойкость ИС к ЭСР. Не вдаваясь в подробное рассмотрение модели (10) можно отметить следующее:

- во-первых, λ в этой модели не зависит от времени
- во-вторых, все данные, необходимые для расчетов составляющих модели (10), приведены в соответствующих разделах RIAC-HDBK-217Plus [12].

Следует отменить, что в справочниках [4-6, 12] приводятся «усредненные» значения базовой интенсивности отказов ($\lambda_б$) для групп (подгрупп) ИС. Вместе с тем, ведущие фирмы-производители ИС регулярно публикуют отчеты о надежности своей продукции, полученные по результатам испытаний. На рис. 3, в качестве примера, приведены выдержки из такого отчета фирмы XILINX [13].

Table 2-24: **HTOL Test Results for 0.09 µm Si Gate CMOS Device Type XC4VxXxxx**

Device	Lot Quantity	Fail Quantity	Device Quantity	Actual Device Hours at $T_J \geq 125°C$	Equivalent Device Hours at $T_J = 125°C$	Failure Rate at 60% CL and $T_J = 55°C$ (FIT)
XC4VLX60	3	0	136	230,316	531,451	
XC4VLX80	1	0	75	150,525	410,767	
XC4VLX100	1	0	45	90,855	259,245	
XC4VLX160	2	2(1)	89	174,132	721,939	
XC4VFX12	1	0	45	90,000	116,627	
XC4VFX60	1	0	42	84,630	182,316	
XC4VxXxxx	9	2	432	820,458	2,222,345	18 FIT

Notes:

1. Failure due to substrate defect. New process improvement has been implemented.

Рис. 3. Интенсивность отказов кристаллов ИС семейства XC4VxXxxx

Значение интенсивности отказов рассчитано на основе следующего соотношения:

$$\lambda = \frac{\chi^2(P,m)}{2 \cdot N \cdot \Delta t \cdot A} \cdot 10^9. \qquad (11)$$

где: χ^2 - распределение хи-квадрат; P - доверительная вероятность; N - общее число испытуемых ИС; A - ускоряющий фактор; Δt - время испытаний.

Применение формулы (11) для расчета интенсивности отказов возможно только для случая, когда «ранние» отказы исключены тренировкой ИС, поэтому оценка интенсивности отказов ИС, полученная с использованием статистики χ^2, справедлива только для периода «нормальной работы» (см. рис. 1).

Пересчет значения λ к «нормальной» температуре может быть проведен по модели ускоряющего фактора A:

$$A = \exp\left[\frac{E_a}{k} \cdot \left(\frac{1}{298} - \frac{1}{T_J + 273}\right)\right].$$

Таким образом, очевидно, что использование экспериментальных данных фирм-производителей позволит повысить точность оценки надежности СБИС. Однако

следует помнить о том, что ряд коэффициентов моделей (1)-(4) и (10), учитывающих влияние соответствующих факторов на интенсивность отказов кристалла ИС, «автоматически» войдет в значение λ, полученное по результатам испытаний.

Исходя из вышеизложенного можно сделать вывод о том, что рассмотренные выше методы расчетной оценки надежности полупроводниковых ИС могут применяться для расчетной оценке надежности бортовой электронной аппаратуры диагностики и мониторинга технического состояния распределенных систем при условии использования экспериментальных данных о характеристиках надежности, публикуемых фирмами-производителями [14].

Список литературы

1. ГОСТ 27.003-90. Надежность в технике. Состав и общие правила задания требований по надежности.
2. ГОСТ 27.301-95. Надежность в технике. Расчет надежности. Основные положения.
3. РДВ 319.01.20-98. Положение о справочнике «Надежность электрорадиоизделий».
4. MIL-HDBK-217F. Reliability prediction of electronic equipment.
5. Справочник «Надежность ЭРИ». - М.: МО РФ, 2006.
6. Справочник «Надежность ЭРИ ИП» - М.: МО РФ, 2006.
7. ГОСТ 27.005-97. Надежность в технике. Модели отказов. Основные положения.
8. Сетун А.Н. Физика отказов приборов микро- и наноэлектроники. - Минск: БГУ, 2010. - 125 с.
9. RADS-TR-89-177. VHSIC/VHSIC-LIKE. Reliability prediction modeling.
10. Жаднов В.В., Полесский С.Н., Якубов С.Э. Прогнозирование безотказности микросхем для военной и аэрокосмической электроники. / Электронные компоненты. № 3, 2007. - с. 39-48.
11. Абрамешин А.Е., Жаднов В.В., Жаднов И.В. Расчетная оценка надежности электронных модулей аппаратуры космических аппаратов. / Технологии ЭМС, № 1 (40), 2012. - с. 29-33.
12. RIAC-HDBK-217Plus. Handbook of 217Plus™ reliability prediction models.
13. UG116 (v 9.1). Device reliability report: Second quarter 2012. - XILINX, 2012.
14. Абрамешин А.Е., Жаднов В.В., Полесский С.Н. Информационная технология обеспечения надежности электронных средств наземно-космических систем: научное издание. - Екатеринбург: Форт Диалог-Исеть, 2012. - 565 с.

Сервисная система SaaS-модели программного обеспечения расчетов надежности РЭА

Жаднов В.В.
Кафедра Радиоэлектроники и телекоммуникаций,
Московский институт электроники и математики национального исследовательского университета «Высшая школа экономики», Москва

«Облачная» модель распространения и поддержки программных средств (англ. Software as a Service, сокращенно SaaS) предполагает использование их в режиме удаленного доступа [1]. Суть этой модели заключается в максимальном переносе процесса вычислений и хранения данных на сервер. При реализации модели SaaS практически все данные хранятся на сервере, основной объем вычислений проводятся на сервере, а взаимодействие пользователя с программным средством осуществляется посредством обмена данными через сервер. [2]. При этом на долю рабочей станции пользователя (компьютер, планшет, смартфон и др.) остается задача обеспечения связи с сервером, отображения информации и предоставление минимальных вычислительных ресурсов [3].

Такими свойствами обладает система АСОНИКА-К-СЧ программного комплекса АСОНИКА-К [4]. Однако, входящая в ее состав система администрирования пользователей, лишь отчасти отвечала требованиям к сервисной модели программного обеспечения, которое должно [5]:

- содержать функциональности развертывания сервиса и администрирования пользователей;
- включать механизмы тарификации условий работы пользователей и проведения биллинга в соответствии с выбранными пользователем условиями и потребленными ресурсами;
- обеспечивать удобные механизмы обращения в службу технической поддержки сервиса непосредственно с рабочей станции пользователя.

Поэтому для реализации этих требований была создана сервисная система, отвечающая вышеперечисленным требованиям и реализованная в виде WEB-портала, состав которого приведен на рис. 1.

Ниже приведено краткое описание модулей системы.

Модуль регистрации отвечает за регистрацию пользователя в системе и содержит две функции. Функция вывода формы регистрации выводит форму регистрации в HTML-формате и отправляет данные в функцию регистрации. Функция регистрации получает данные от функции вывода формы регистрации и проверяет введенные значения. Если правильно заполнены все необходимые поля и пользователя с введенным логином не существует, то отправляется SQL-запрос на добавление нового пользователя. В противном случае возвращается сообщение об ошибке.

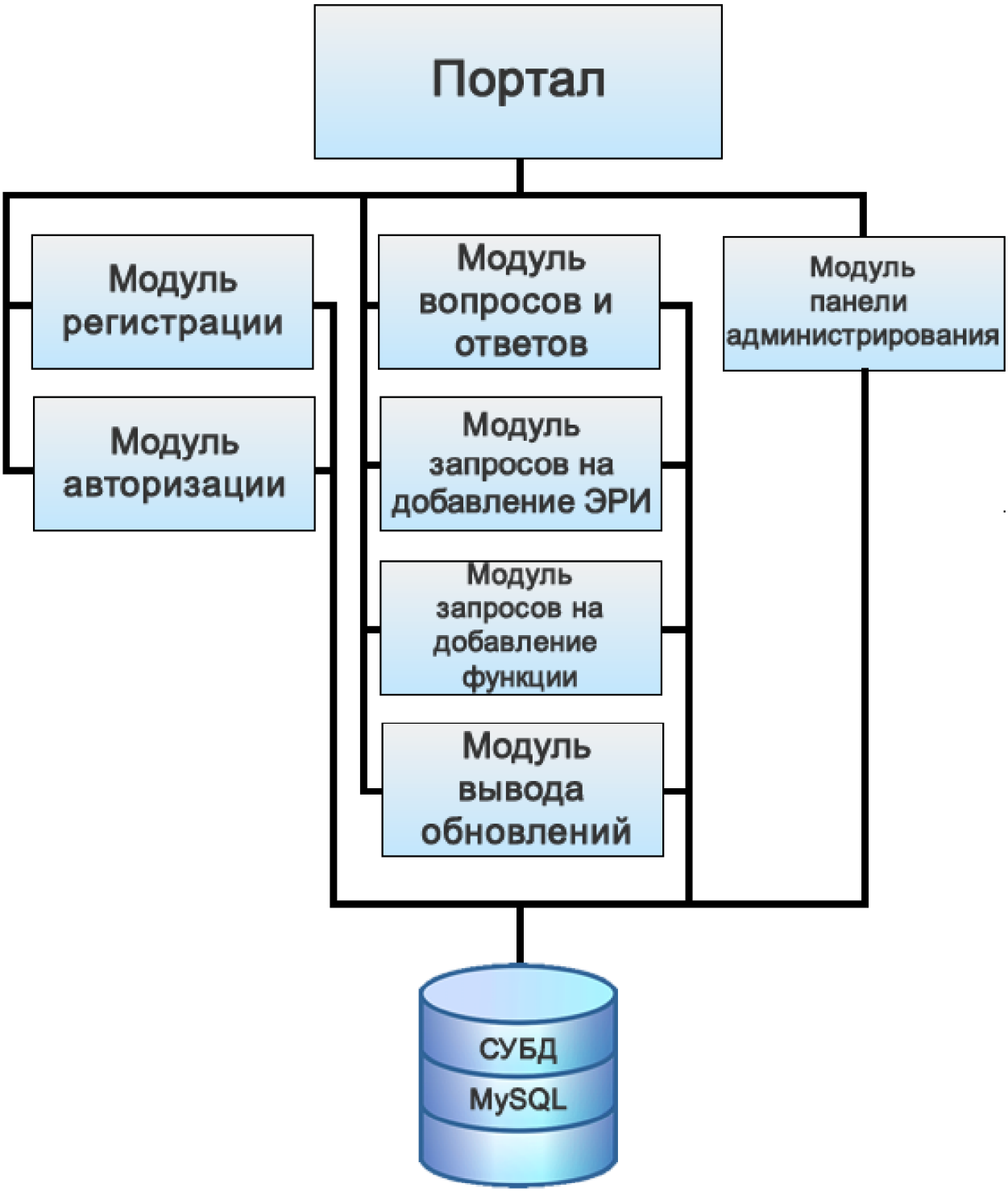


Рис. 1. Состав модулей сервисной системы

Модуль авторизации отвечает за успешную авторизацию в системе и содержит две функции. Функция вывода формы авторизации выводит форму авторизации на портале в HTML-формате и отправляет данные в функцию авторизации. Функция авторизации получает данные от функции вывода формы авторизации и проверяет введенные значения. Формирует SQL-запрос к базе данных. Если пользователь и пароль совпадают с существующими в базе данных, то авторизация проходит успешно. В ином случае выводится сообщение об ошибке и авторизации не происходит.

Модуль вопросов и ответов занимается системой вопросов и ответов со стороны пользователя и содержит четыре функции. Функция показа списка вопросов выводит список вопросов, заданных пользователем и выделяет вопросы, на которые есть ответ. Функция вывода формы вопроса выводит форму в HTML-формате и отправляет данные в функцию добавления вопроса. Функция вывода вопроса выводит заданный вопрос и ответ, если он есть. Функция добавления вопроса отправляет на сервер SQL-запрос и выводит сообщение об успешном запросе.

Модуль добавления ЭРИ в систему АСОНИКА-К-СЧ отправляет запрос на добавление ЭРИ и содержит две функции. Функция вывода формы добавления ЭРИ выводит форму добавления ЭРИ в HTML-формате и отправляет данные в функцию добавления ЭРИ. Функция добавления ЭРИ получает данные от функции вывода формы добавления ЭРИ и проверяет введенные значения. Если все поля заполнены правильно, то функция добавляет запрос в базу данных и выводит сообщение об успехе. В противном случае выводится сообщение об ошибке.

Модуль добавления функции в систему АСОНИКА-К-СЧ отправляет запрос на добавление функции и содержит две функции. Функция вывода формы добавления функции выводит форму добавления функции в HTML-формате и отправляет данные в функцию добавления функции. Функция добавления функции получает данные от функции вывода формы добавления функции и проверяет введенные значения. Если все заполнено правильно, то функция добавляет запрос в базу данных и выводит сообщение об успехе. В противном случае выводится сообщение об ошибке.

Модуль вывода обновления системы АСОНИКА-К-СЧ выдает список файлов и содержит функцию вывода файлов, которая выводит список необходимых для работы архивов системы АСОНИКА-К-СЧ в формате HTML.

Модуль административной панели отвечает за управление порталом и содержит восемь функций. Функция вывода списка запросов ЭРИ выводит список запросов на добавление ЭРИ в HTML-формате. Функция вывода списка запросов на добавление функции выводит список запросов на добавление функции в HTML-формате. Функция вывода списка вопросов выводит список вопросов в HTML формате. Функция вывода запроса ЭРИ выводит запрос на добавление ЭРИ в HTML-формате. Функция вывода запроса на добавление функции выводит запрос на добавление функции в HTML-формате. Функция вывода вопроса выводит заданный вопрос в HTML формате. Функция вывода формы ответа на вопрос выводит форму для ответа на вопрос в HTML формате и отправляет данные в функцию ответа на вопрос. Функция ответа на вопрос получает данные из функции вывода формы ответа на вопрос. Если сообщение не пустое, то добавляет ответ в заданный вопрос. В противном случае выводит сообщение об ошибке.

В сервисной системе для хранения данных используется СУБД MySQL. База данных системы состоят из четырех таблиц, для каждой из которых использован префикс assk_ во избежание случайных совпадений.

Данные пользователей хранятся в таблице ASSK_USERS, описание которой приведено в табл. 1.

Таблица 1. Описание таблицы ASSK_USERS

Поле	Тип	Описание
1	2	3
a_id	int	Целое число, отвечающее за уникальный номер пользователя.
a_mail	text	Текстовое поле. Хранит в себе e-mail пользователя.
a_login	varchar(64)	Строка длиной 64 символа. Хранит в себе логин пользователя, под которым он заходит в систему.
a_pass	varchar(64)	Строка длиной 64 символа. Хранит в себе пароль пользователя.
a_name	text	Текстовое поле, которое содержит имя пользователя.
a_family	text	Текстовое поле, которое содержит фамилию пользователя.
a_secondname	text	Текстовое поле, которое содержит отчество пользователя.
a_firmname	text	Текстовое поле. Хранит в себе название предприятия.
a_jobadress	text	Текстовое поле. Хранит в себе адрес предприятия.
a_mail_index	varchar(256)	Строка длиной 256 символов. Почтовый адрес предприятия.
a_job_position	text	Текстовое поле. Хранит в себе должность пользователя.
a_telephone	varchar(16)	Строка длиной 16 символов. Хранит в себе контактный телефон.
a_fax	varchar(16)	Строка длиной 16 символов. Хранит в себе факс пользователя.
a_know	text	Текстовое поле.
a_desc	text	Текстовое поле.
a_balance	float	Число с плавающей запятой. Хранит в себе число оставшихся запросов.
a_perm	enum(‘0’,’1’)	Перечисление между 0 и 1. Отвечает за уровень доступа пользователя. 0 – простой пользователь, 1 – администратор.
a_status	enum(‘0’,’1’)	Перечисление между 0 и 1. Отвечает за статус пользователя. 0 –пользователь выключен, 1 – пользователь включен.
a_hddnum	varchar(256)	Строка длиной 256 символов. Хранит в себе номер жесткого диска пользователя.
a_lastcount	int	Целое число. Хранит в себе последнее количество запросов к ПК АСОНИКА-К.

Вопросы пользователей и ответы администратора хранятся в таблице ASSK_QA, описание которой приведено в табл. 2.

Заявки пользователей на добавление ЭРИ в СЧБД системы АСОНИКА-К-СЧ хранятся в таблице ASSK_ERI, описание которой приведено в табл. 3.

Запросы пользователей на добавление функций в систему АСОНИКА-К-СЧ хранятся в таблице ASSK_FUNC, описание которой приведено в табл. 3.

Таблица 2. Описание таблицы ASSK_QA

Поле	Тип	Описание
qa_id	int	Целое число. Хранит в себе уникальный идентификатор вопроса.
a_id	int	Целое число. Хранит в себе уникальный идентификатор автора вопроса.
qa_status	enum(‘0’,’1’)	Перечисление между 0 и 1. Отвечает за статус сообщения. 0 – сообщение без ответа, 1 – на сообщение есть ответ.
qa_question	text	Текстовое поле. Хранит в себе вопрос.
qa_answer	text	Текстовое поле. Хранит в себе ответ.
qa_time	int	Целое число. Хранит в себе время отправки сообщения согласно UNIX-времени.

Таблица 3. Описание таблицы ASSK_ERI

Поле	Тип	Описание
e_id	int	Целое число. Хранит в себе уникальный идентификатор заявки на добавление ЭРИ.
a_id	int	Целое число. Хранит в себе уникальный идентификатор автора заявки на добавление ЭРИ.
a_login	text	Текстовое поле. Хранит себе логин автора заявки на добавление ЭРИ.
e_name	text	Текстовое поле. Хранит в себе типономинал (наименование) ЭРИ.
e_func_group	text	Текстовое поле. Хранит в себе функциональную группу ЭРИ.
e_nomer_tu	text	Текстовое поле. Хранит в себе номер ТУ ЭРИ.
e_ktp	text	Текстовое поле. Хранит в себе конструктивно-технологические параметры ЭРИ.
e_ep	text	Текстовое поле. Хранит в себе электрические параметры ЭРИ.
e_up	text	Текстовое поле. Хранит в себе условия применения ЭРИ.
e_file	text	Текстовое поле. Хранит в себе файл технической документации ЭРИ.

Таблица 4. Описание таблицы ASSK_FUNC

Поле	Тип	Описание
f_id	int	Целое число. Хранит в себе уникальный идентификатор запроса на добавление функции.
a_id	int	Целое число. Хранит в себе уникальный идентификатор автора запроса на добавление функции.
a_login	varchar(64)	Строка длиной 64 символа. Хранит в себе логин автора заявки на добавление функции.
f_text	text	Текстовое поле. Хранит в себе текст заявки на добавление функции.
f_status	enum(‘0’,’1’)	Перечисление между 0 и 1. Отвечает за статус заявки. 0 – заявка не была просмотрена. 1 – заявку просмотрел администратор.

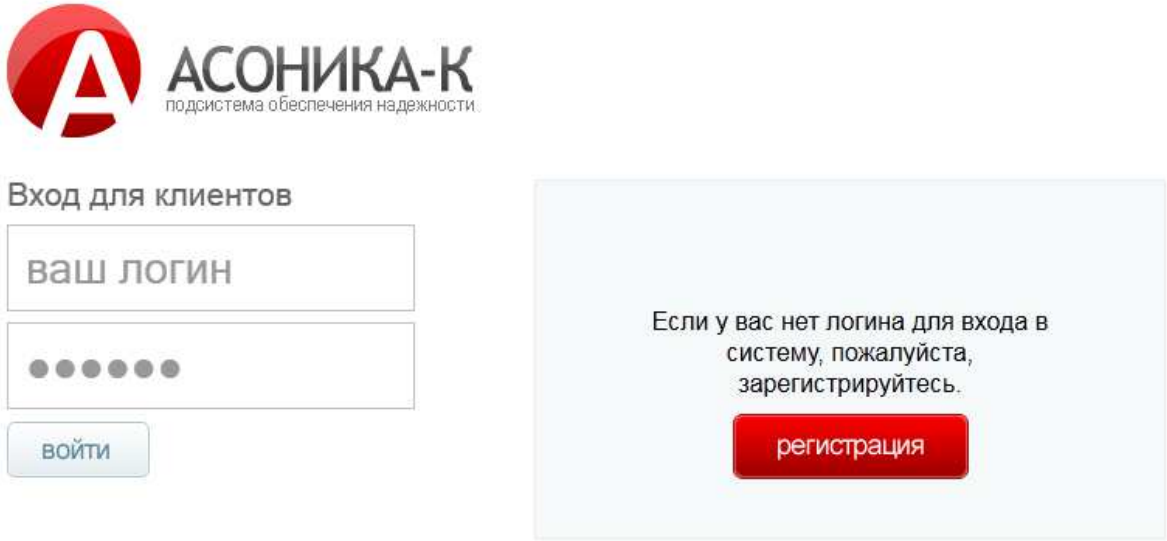


Рис. 3. Начальная страница WEB-портала

Все таблицы связаны уникальным идентификатором пользователя (A_ID), который присваивается автоматически при регистрации.

Укрупненная Блок-схема алгоритма функционирования сервисной системы приведена на рис. 2.

Ниже приведено краткое описание блоков алгоритма.

Блок 1. Начало. Запуск браузера и переход на начальную страницу WEB-портала (см. рис. 3).

Блок. 2. Решение. Выбор способа входа в систему: ввод персональных данных или регистрация.

Блок. 3. Регистрация. Ввод данных, необходимых для регистрации нового пользователя (см. рис. 4).

В случае успешной регистрации в таблице ASSK_USERS базы данных появится новая запись (см. рис. 5).

После успешной регистрации переход на Блок 5. В противном случае переход на Блок 2.

Блок 5. Авторизация. Ввод персональных данных пользователя. В случае успешной авторизации переход на Блок 5. В противном случае переход на Блок 2.

Блок 6. Вывод интерфейса сервисной системы (см. рис. 6).

Блок 7. Решение. Выбор функции сервисной системы.

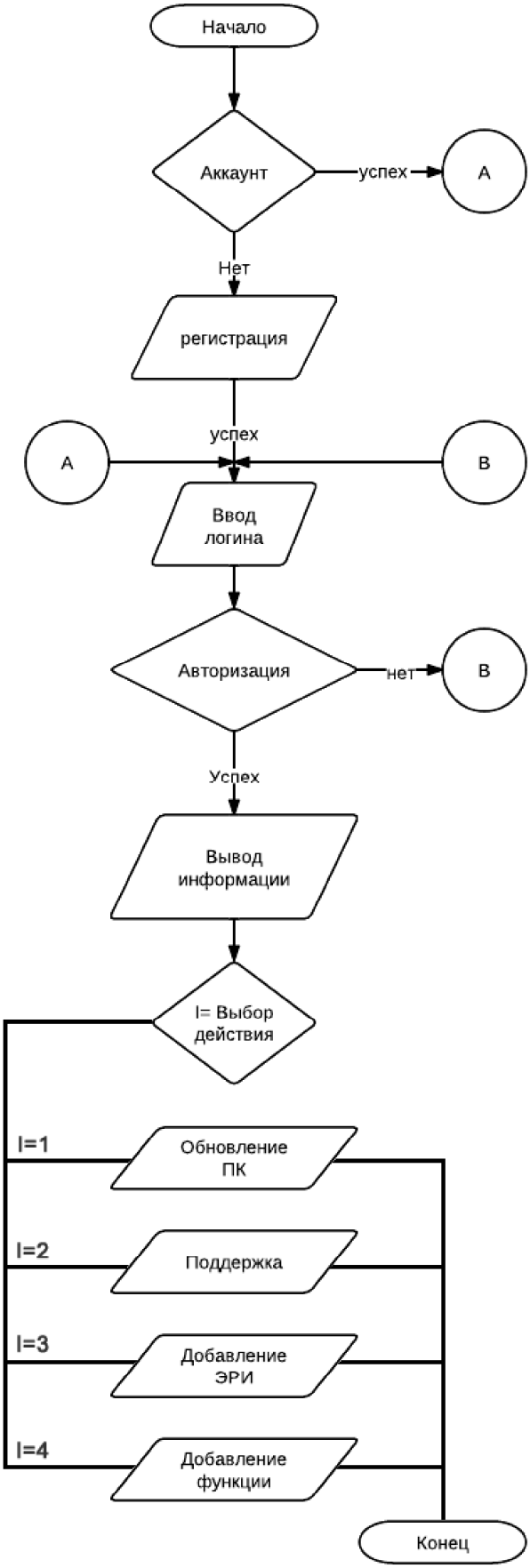


Рис. 2. Алгоритм функционирования сервисной системы

АСОНИКА-К
подсистема обеспечения надежности

Регистрация

Логин*
Пароль*
E-mail*
Фамилия*
Имя*
Отчество*

Организация
Адрес места работы
Почтовый адрес
Должность
Телефон
Факс
Откуда вы о нас узнали
Примечание

Я согласен с условиями публичной оферты*

Регистрироваться | Вернуться назад

Поля отмеченные * обязательны к заполнению.

Рис. 4. Страница регистрации

←T→	a_id	a_mail	a_login	a_pass	a_name	a_family	a_secondname
	1	sergionikol@gmail.com	snikolaev	qer@#$%jkb!4nkjk	Сергей	Николаев	Сергеевич

Рис. 5. Фрагмент таблицы ASSK_USERS

АСОНИКА-К
подсистема обеспечения надежности

Информация
Пополнить баланс
Обновление ПС
Поддержка
Добавление функции
Добавление ЭРИ
Выход

Отправить вопрос

Рис. 6. Интерфейс сервисной системы

Блоки 8-11. Вывод страницы соответствующей функции.

Поддержка (Блок 9). После отправления вопроса в таблице ASSK_QA базы данных появится новая запись (см. рис. 7).

qa_id	a_id	qa_status	qa_question	qa_answer	qa_time
1	1	0	Добрый день. Какая у вас температура за окном?		1338108768

Рис. 7. Фрагмент таблицы ASSK_QA

Как видно из рис. 7 вопрос добавился. Статус у вопроса 0, а это значит, что администратор системы еще не ответил на него.

Добавление ЭРИ (Блок 10). После отправления запроса на добавление ЭРИ в таблице ASSK_ERI базы данных появится новая запись (см. рис. 8).

Добавление функции (Блок 11). После отправления запроса на добавление функции в таблице ASSK_FUNC базы данных появится новая запись (см. рис. 9).

e_id	a_id	a_login	e_name	e_func_group	e_nomer_tu
1	1	snikolaev	К50-35	Конденсатор	939556220

Рис. 8. Фрагмент таблицы ASSK_ERI

f_id	a_id	a_login	f_text	f_status
1	1	snikolaev	Добрый день. У меня в ОС Windows нет калькулятора....	0

Рис. 9. Фрагмент таблицы ASSK_FUNC

Как видно из рис. 9 сообщение добавилось. Статус у заявки 0, а это значит, что администратор системы еще не просмотрел заявку.

Блок 12. Окончание. Завершение работы с системой.

Опытная эксплуатация сервисной системы подтвердила ее функциональные возможности и эффективность работы с интернет-пользователями системы АСОНИКА-К-СЧ.

Список литературы

1. Фингар П. Dot. Cloud: облачные вычисления - бизнес-платформа XXI века. - М.: Аквамариновая Книга, 2011. - 256 с.
2. Риз Д. Облачные вычисления. - СПб.: БХВ-Петербург, 2011. - 288 с.
3. Деревянко А.С., Солощук М.Н. Технологии и средства консолидации информации: Учебное пособие. - Харьков: НТУ «ХПИ», 2008. - 432 с.
4. Абрамешин А.Е., Жаднов В.В., Полесский С.Н. Информационная технология обеспечения надежности электронных средств наземно-космических систем: Научное издание. / Отв. ред. В.В. Жаднов. - Екатеринбург: Форт Диалог-Исеть, 2012. - 565 с.
5. Жаднов В.В. Перспективы создания программного обеспечения на технологиях «облачных вычислений» для расчетов надежности РЭА. / Инновации на основе информационных и коммуникационных технологий: материалы международной научно-технической конференции. // Под. ред. С.У. Увайсова, отв. за вып. И.А. Иванов. - М.: МИЭМ НИУ ВШЭ, 2012. - с. 288-292.

«Системы автоматизированного проектирования»

Моделирование термоэлемента Пельтье для процессов оптимизации систем обеспечения тепловых режимов радиоэлектронных средств

И.Ю.Громов, А.М. Кожевников
Каф. ИТАС, Московский государственный институт электроники и математики НИУ ВШЭ

Надежность работы и стабильность параметров радиоэлектронных устройств в значительной степени определяются их температурным состоянием [2], поэтому при решении задачи обеспечения теплового режима функционирования радиоэлектронных средств (РЭС) нужно исходить из того, что все электрорадиоизделия (ЭРИ), которые входят в состав РЭС, должны работать в нормальном тепловом режиме. В настоящее время основным показателем надежности ЭРИ с точки зрения их теплового режима является нахождение рабочей температуры ЭРИ в диапазоне, регламентируемом изготовителем, т.к. статистические показатели влияния их температуры на надежность чаще отсутствуют.

Для выполнения требования к тепловому режиму каждого ЭРИ из состава РЭС могут использоваться различные термоэлементы: локальные радиаторы, вентиляторы, теплоотводящие шины, термоэлектронные элементы Пельтье, тепловые трубы. Тепловые трубы ограничиваются компактными стандартными круглым и плоским типами. Применяются также различные типы радиаторов и всевозможные типы нагнетателей (вентиляторов) – как вдувающие, так и выдувающие.

При современной конструкторской разработке на этапе эскизного проектирования РЭС необходимо проанализировать целесообразность применения различных термоэлементов и их влияние на его тепловой режим. Для этого при расчете теплового режима составляется модель теплообмена, в которой учитывается конструкция РЭС, происходящие конвективный, кондуктивный и лучистый теплообмен и параметры средств теплообмена.

Для выполнения оценки влияния элементов системы на тепловое состояние РЭС и ЭРИ предлагается метод оптимизации систем обеспечения тепловых режимов РЭС, который в качестве модели теплообмена использует способ электротепловой аналогии. Концепция метода оптимизации построена на анализе температурного поля РЭС, вычисляемого по модели теплообмена в РЭС после пошагового изменения параметров средств обеспечения теплового режима. При определении направления движения к оптимуму значения каждого параметра определяются путем вычисления на очередном шаге оптимизации критерия оптимальности для блока РЭС в целом и для каждого средства обеспечения теплового режима, присутствующего в модели. Следует заметить, что важной особенностью предлагаемого метода оптимизации является возможность учета целесообразности применения того или иного средства обеспечения температурного режима для ЭРИ или для блока РЭС. Это достигается с помощью учета индивидуального параметра – эффективности, характеризующего затраты на применение средства обеспечения теплового режима.

В современных РЭС для выполнения задач поддержания температурного режима ЭРИ либо приборного контейнера все чаще применяется термоэлемент Пельтье, поэтому возникает задача моделирования его поведения в тепловой системе.

Элемент обладает множеством параметров и проявляет ряд комплексных тепловых и электрических эффектов.

Элемент Пельтье основан на эффекте выделения или поглощения тепла на границе двух разных проводников, при прохождении через них электрического тока (эффект Пельтье) (Рис.1).

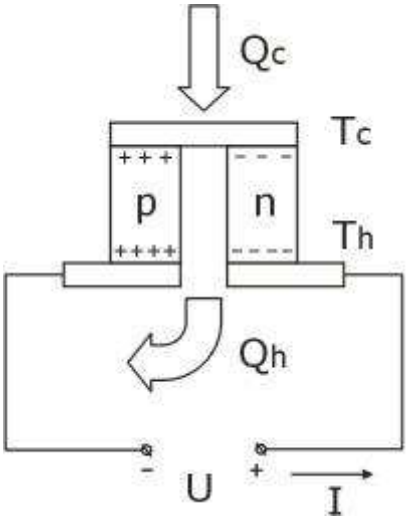


Рис. 1. Схема действия элемента Пельтье [5]

Конструктивно термоэлемент состоит из набора полупроводниковых стержней «дырочного» и «электронного» типа проводимости, соединенных металлическим мостиком. Термоэлемент может работать в режиме получения тепла, холода либо как электрогенератор. Как правило, в бортовой РЭС элементы Пельтье выполняют охлаждающую функцию приборного контейнера в целом и/или отдельных ЭРИ в частности. В режиме получения холода на термоэлементе проявляется ряд эффектов, которые необходимо учесть при его моделировании.

Рассмотрим основные параметры элемента Пельтье согласно [2]. Если выбрать направление тока I через элемент так, чтобы тепло поглощалось в охлаждающем спае при температуре T и выделялось в другом спае при температуре T_0, то тогда поглощаемую тепловую мощность в результате эффекта Пельтье можно выразить через:

$$Q = \alpha I T, \qquad (1)$$

а выделяемую мощность через:

$$Q_0 = \alpha_0 I T_0, \qquad (2)$$

где α – термоэдс обоих ветвей элемента.

Согласно эффекту Джоуля мощность выделяемой внутри элемента теплоты:

$$W = I^2 r, \qquad (3)$$

где: r – электрическое сопротивление обоих ветвей термоэлемента,

I – сила тока, протекающая через термоэлемент.

При этом половина мощности W переходит к теплой стороне модуля, половина – к холодной. Мощность, затрачиваемая на теплопроводность между теплой стороной модуля и холодной определяется через:

$$Q_T = k(T_0 - T), \qquad (4)$$

где k – коэффициент теплопроводности модуля.

Помимо этого существует затрата мощности током против разности потенциалов между теплыми и холодными спаями, возникающую согласно эффекту Зеебека в результате разности температур:

$$Q_{\Pi} = \alpha I(T_0 - T), \qquad (5)$$

Эффект Томпсона принято оставлять без внимания ввиду малого его влияния на модель по сравнению с другими эффектами [1,2].

Основными эксплуатационными параметрами элементов Пельтье являются *Imax* – ток, при котором достигается разность температур *ΔTmax*, *Qmax* – максимальная холодопроизводительность, *Umax* – напряжение, соответствующее *Imax*, *ΔTmax* – максимальная разность температур между сторонами модуля при определенной температуре горячей стороны.

В настоящее время разработан ряд моделей в соответствии с методом электротепловых аналогий, учитывающий большинство указанных эффектов [1]. Данные модели основаны либо на распределенных либо на сосредоточенных электрических параметрах, вычисляемых из основных эксплуатационных параметров, указываемых производителями в листах спецификаций на производимые модули. Вышеуказанные модели описывают поведение термоэлемента через эквивалентные электрические цепи, в которых термоэлектрические эффекты определяются через электрические токи (аналогии тепловых потоков) в ветвях [1]. Такой подход позволяет получить итоговые значения тепловых потоков на холодной и горячей стороне модуля и значения температур через вычисление всех токов в электрической цепи применяя законы Кирхгофа.

Т.к. данные модели основаны только на эксплуатационных параметрах термоэлемента и не учитывают конструктивные особенности конкретного модуля и материалы из которых выполняются термопары, т.е. не известен коэффициент термоэдс для полупроводников, следовательно, нельзя с высокой точностью рассчитать зависимости (1) и (2) и существует вероятность неточного моделирования поведения элемента.

Исходя из того, что производитель публикует подробные функциональные зависимости между различными параметрами термоэлемента Пельтье [6], можно предложить упрощенную и вместе с тем адекватную модель, использующую минимальный набор параметров элемента. На рис.1 представлена упрощенная эквивалентная модель термоэлемента, построенная в соответствии с методом электротепловых аналогий, которая может быть встроена в общую электротепловую модель РЭС:

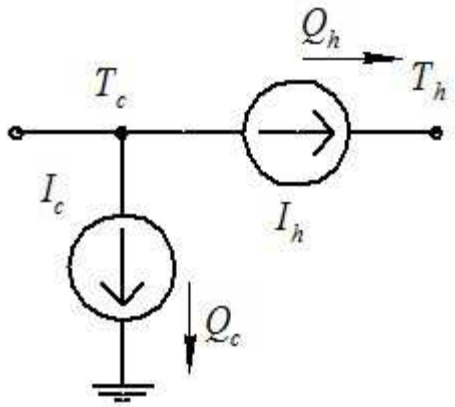


Рис. 2. Электротепловой аналог элемента Пельтье

В процессе решения задачи оптимизации необходимо обеспечить температуры на ЭРИ или в приборном контейнере в необходимом диапазоне нижней $T_н$ и верхней $T_в$ температур, т.е. температура T_c (К) на холодной стороне модуля должна

соответствовать диапазону $[T_н, T_в]$, а значение холодопроизводительности Q_c (Вт) соответствовать тепловыделению Q_e (Вт) ЭРИ/контейнера, на который устанавливается термоэлемент. В процессе охлаждения на горячей стороне выделяется тепловой поток Q_h в зависимости от заданного тока питания I (А). I_c *и* I_h – источники тока, электрические аналоги тепловых потоков Q_c и Q_h, численно равные им. T_a – температура окружающей среды. T_c и T_h – температура на холодной и горячей стороне модуля соответственно.

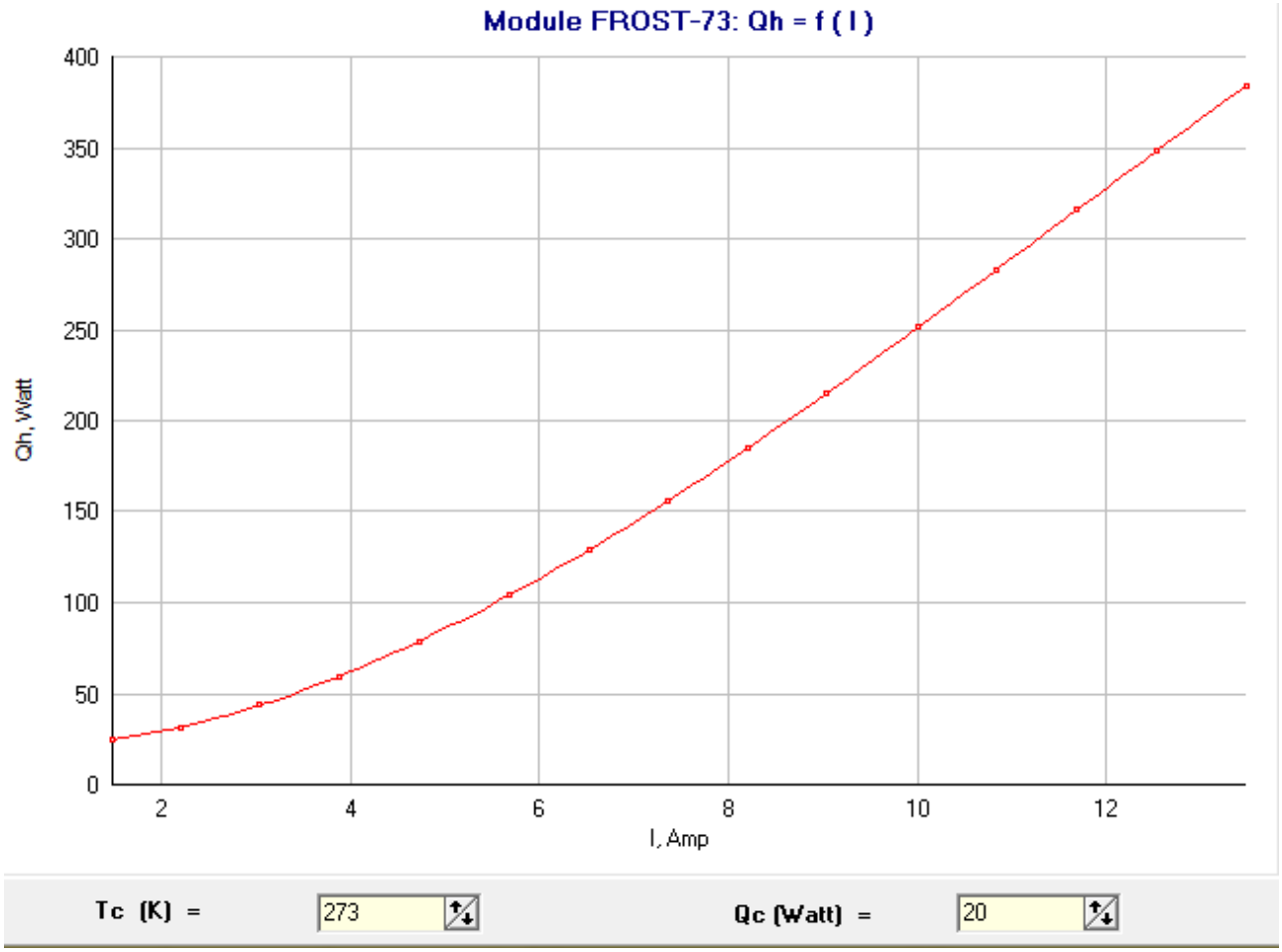


Рис.3. Зависимость Q_h=f(I) при задаваемых значениях T_h и Q_c [6]

Т.к. известны параметры тепловыделения на ЭРИ или приборном блоке и диапазон температур, необходимый для его надежной работы, то искомой зависимостью следует считать тепловой поток Q_h на горячей стороне модуля.

На рис. 2 представлен график зависимости Q_h=*f(I)* при задаваемых значениях T_c и Q_c по данным производителя термоэлемента. Следовательно, функциональная зависимость искомого теплового потока Q_h в общем виде представляется как:

$$Q_h = f(I, T_c, Q_c), \text{ при } T_c \in [T_н, T_в],\ Q_c \approx Q_e,\ I \in [0, I_{\max}].$$

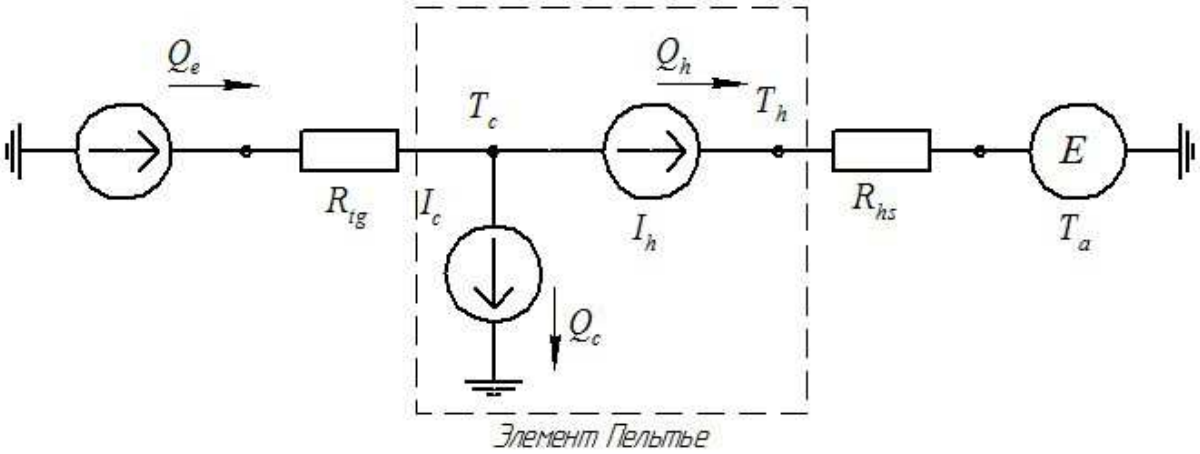
Конечный вид функции Q_h находится через решение интерполяционной задачи регрессионного анализа априорных данных, представленных в программном обеспечении с информацией о термоэлементах компании «Криотерм». Связь параметра Q_h с переменными факторами I, T_c и Q_c в таком случае можно представить в виде полинома второй степени, который выражается в виде уравнения:

$$Q_h = b_0 + b_1 I + b_2 T_c + b_3 Q_c + b_{11} I^2 + b_{22} T_c^2 + b_{33} Q_c^2 + b_{12} I T_c + b_{13} I Q_c + b_{23} T_c Q_c + b_{123} I T_c Q_c,$$

где b_0, b_1, … - коэффициенты регрессии.

Для определения коэффициентов регрессии используется план эксперимента второго порядка, согласно которому проводится 8 опытов полного факторного эксперимента, где каждый фактор варьируется не менее чем на трех уровнях. Значения функции отклика для каждого опыта определяются по зависимости $Q_h=f(I)$ ПО «Криотерм» конкретной модели термоэлекрического модуля, которая априори учитывает термоэлектрические эффекты, описанные уравнениями (1-5).

Поведение термоэлектрического модуля Пельтье моделируется целевой функцией Q_h исходя из исходных данных по тепловыделению компонента, на который он устанавливается и тепловым режимам, необходимым для его надежной работы. Значение температур *Tc* и *Th* могут быть найдены после встраивания электротепловой модели Пельтье в модель теплообмена РЭС. При этом нужно учесть, что элемент Пельтье, как правило, работает в паре с радиатором либо в сборке [эл.Пельтье + Радиатор + Вентилятор], чтобы обеспечить эффективный отвод тепла от горячей стороны модуля и снизить разность температур ΔT между горячей и холодной сторонами. На рис. 4 представлен пример модели элемента Пельтье в паре с радиатором на охлаждаемом ЭРИ/блоке.



R_{tg} – тепловое сопротивление, вносимое термопастой или термоклеем между термоэлементом и охлаждаемым компонентом; R_{hs} – тепловое сопротивление радиатора; Та – температура окружающей среды.

Рис. 4. Модель сборки элемента Пельтье и радиатора

Помимо этого, исходя из общей концепции оптимизации модель элемента Пельтье учитывает стоимостные параметры, позволяющие оценить параметры элемента с точки зрения затрат на их реализацию:

$$C_П = f(a, b, I, U),$$

где: $C_П$ – стоимость применения модуля Пельтье;

a, b – параметры радиатора и вентилятора для отвода тепловой мощности;

I, U – ток и напряжение, необходимые при работе модуля.

Конечный вид функции определяется в результате регрессионного анализа экспериментальных данных зависимостей параметров модуля.

Таким образом, оптимизация будет выполняться путем варьирования параметров для индивидуальных ЭРИ и блока РЭС различных элементов обеспечения тепловых режимов (как вышеописанного элемента Пельтье, так и других) и оценки целесообразности их применения через стоимость. В итоге планируется получить оптимальную конструкцию средств обеспечения теплового режима РЭС с учетом минимальных затрат на реализацию.

Литература

1. Chakib Alaoui. Peltier Thermoelectric Modules Modeling and Evaluation. International Journal of Engineering (IJE), Volume (5) : Issue (1) : 2011

2. Иоффе, А.Ф. Полупроводниковые термоэлементы / А.Ф. Иоффе ; АН СССР, Ин-т полупроводников. – М. ; Л. : Изд-во АН СССР, 1960. – 188 с.

3. Спиридонов А.А., Планирование эксперимента при исследовании технологических процессов. – М.: Машиностроение, 1981. – 184 с., ил.

4. Г.Н.Дульнев “Тепло- и массообмен в радиоэлектронной аппаратуре”. Москва, “Высшая школа”, 1984г.

5. http://peltier.narod.ru/ Элементы Пельтье. Дата обращения: январь 2013.

6. http://www.kryotherm.ru/ru/index.phtml?tid=84. Программа Криотерм. Дата обращения: январь 2013.

Метод оптимизации системы радиатор-кулер обеспечения тепловых режимов электрорадиоэлементов

А.М. Кожевников, И.Ю.Громов
Кафедра ИТАС, Московский государственный институт электроники и математики НИУ ВШЭ

При проектировании современных радиоэлектронных средств (РЭС), особенно бортовых, сталкиваются с серьезными проблемами, основные из которых:

- увеличение требований к надежности при ужесточении условий эксплуатации;
- снижение стоимостных параметров (массогабаритных, ценовых, энергопотребления и т.д.) при одновременном увеличении количества выполняемых функций;
- уменьшение сроков морального старения и, соответственно, необходимость сокращать сроки проектирования новых изделий.

Необходимость правильного выбора проектных решений на этапах разработки связана с тем, что выявление недостаточной надежности РЭС из-за тепловых воздействий на завершающих этапах проектирования (начиная с детального конструкторского проектирования) как путем математического моделирования, так и путем испытаний опытного образца приводит к длительным итерациям по отработке изделия, а значит и к резкому возрастанию материальных затрат и увеличению сроков проектирования.

Исходя из требований нормативно-технической документации по тепловым режимам, нужно на как можно более ранних этапах проектирования оптимально выбрать тип конструкции системы охлаждения обеспечивающие необходимую надежность РЭС..

Повышение качества автоматизированного проектирования РЭС должно состоять в нахождении оптимальных проектных решений, обеспечивающих заданную надежность с учетом всего комплекса влияющих на надежность эксплуатационных факторов при минимальных затратах на реализацию этих решений.

Снижение температурного режима электрорадиоизделий (ЭРИ) уменьшает интенсивность их отказов, но при этом увеличивается стоимость системы охлаждения РЭС. Для обеспечения необходимого температурного режима применяются различные виды охлаждения как РЭС в целом, так и отдельных ЭРИ. Тепловые режимы РЭС в значительной степени определяют надежность ее работы.

Все ЭРИ, которые входят в состав РЭС, должны работать в нормальном тепловом режиме. Тепловой режим ЭРИ считается нормальным, если выполняются два условия: 1) температура ЭРИ в условиях эксплуатации находится в пределах диапазона температур, допустимых для данного ЭРИ; 2) температура ЭРИ должна обеспечивать его работу с заданной надежностью. Обеспечение нормального теплового режима является одной из главных задач, решаемых при проектировании РЭС. Как правило, меры, применяемые для обеспечения нормального теплового режима ЭРИ и РЭС, приводят к увеличению габаритных размеров и массы конструкции, а также дополнительным затратам электроэнергии. Вследствие этого необходимо найти оптимальное решение, компромиссное между необходимостью обеспечить нормальный тепловой режим ЭРИ и недопустимостью значительного увеличения потребления энергии, массы, габаритов и т. д.

Обоснование применяемых мер может быть получено путем нахождения оптимального проектного решения по параметрам элементов системы обеспечения теплового режима.

В статье описан метод определения оптимальных параметров индивидуальных средств обеспечения теплового режима ЭРИ: радиаторов и вентиляторов (кулеров), которые минимизировали бы их стоимость при выполнении требований обеспечения заданного теплового режима ЭРИ. При этом под стоимостью понимается либо ценовой, либо абстрактно-экспертный показатель, включающий в себя требования к качеству изготовления элементов и их типам, к потреблению энергии, массо-габаритным показателям и т.д.

Для анализа тепловой модели использован метод электротепловой аналогии, при этом по исходным данным производится формирование уравнений состояния тепловой модели, которые записываются в матричной форме следующим образом:

$$[G]*[T] = [W],$$

где [G] - матрица тепловых проводимостей, [T] - вектор искомых потенциалов (температур), [W] - вектор воздействий (температура окружающей среды и тепловые потоки от ЭРИ).

Для системы воздушного охлаждения радиоэлектронных аппаратов и полупроводниковых силовых устройств наиболее широкое применение получили радиаторы [1], которые различаются по виду развитой площади поверхности, а именно: пластинчатые и игольчато-штыревые.

Расчет тепловой проводимости между пластинчатым или игольчато-штыревым радиатором и окружающим воздухом проводился по ниже следующему алгоритму.

Обозначим параметры радиатора: sr-шаг ребер, hr-высота ребер или штырьков, dl-длина ребер, delr-толщина ребра или диаметр штырька, delp-просвет между ребрами или штырьками, nr-количество ребер, lam-коэффициент теплопроводности материала радиатора, t-массив температур в узлах модели с номерами nuz…, nuz1 - узел модели нагретой поверхности, nuz2 - узел модели окружающей среды, np - количество промежутков между ребрами.

Для пластинчатого радиатора: sr=(2*hr*dl+delr*dl)*nr (площадь оребренной поверхности), sc=delp*dl*np (площадь просветов).

Для штырькового радиатора: sr=(3.14*(sqr(delr))/4+3.14*delr*hr)*nr; nrdl=round((dl)/(delr+delp))+1 (round-целая часть вещественного числа); nrsh:=nr/nrdl; ssh=delr*nrsh+delp*(nrsh-1); sc=dl*ssh-3.14*(sqr(delr))/4.

Для обоих типов радиаторов: s=sr+sc (общая площадь); sk1=t[nuz1]+t[nuz2]; sk2=t[nuz1]-t[nuz2]; opr=dl; s1=sqrt(sqrt(abs(sk2)/opr)); s2=26.2/(sk1/2+84)+1.13; α_k=s1*s2 (коэффициент конвективного теплообмена); b1=dl*sqrt(2*α_k /(lam*delr)); b2=(exp(b1)-exp(-b1))/(exp(b1)+exp(-b1)); a=b2/b1 (коэффициент эффективности оребрения);

pr=α_k*(a*sr+sc) (тепловая проводимость); r=1/pr {искомое тепловое сопротивление между узлами тепловой модели nuz1 и nuz2}.

При вынужденной конвекции (обдуве радиатора вентилятором) коэффициент конвективного теплообмена определяется по формуле [1]: $\alpha_k = \frac{N_u \lambda}{L}$, где N_u – критерий Нуссельда, $N_u \approx 0{,}032 Re^{0,8}$ при $Re \geq 5 \cdot 10^5$ (вихревое движение воздуха), $N_u \approx 0{,}8\sqrt{Re}$ при $10 \leq Re < 5 \cdot 10^5$ (ламинарное движение воздуха), Re – критерий

Рейнольдса, $Re = \frac{WL}{\nu}$, W – скорость воздуха, ν - коэффициент кинематической вязкости среды, L – определяющий размер (длина обтекания), $L = \sum_{i=1}^{n} L_i F_i \Big/ \sum_{i=1}^{n} F_i$, где L_i, F_i – длина обтекания и площадь теплоотдающей поверхности i-го элемента радиатора соответственно.

Разработаны метод и программа, реализующая алгоритм нахождения оптимального (максимально допустимого) значения теплового сопротивления радиатора, его параметров и производительности вентилятора. Для нахождения оптимальных конструктивных параметров радиатора и вентилятора использован метод оптимизации Хука-Дживса [2].

Задача нахождения минимума массы радиатора и производительности вентилятора решается в два этапа.

На первом этапе в качестве варьируемого параметра используется тепловое сопротивление радиатора (R_P) при конвективном теплообмене, для которого по тепловой модели РЭС находится максимальное значение, при котором температура охлаждаемого ЭРИ ($T_в$) будет равна предельно допустимому для него значению ($T_д$). Целевая функция при этом имеет следующий вид: $f(R_p)=\min(T_в-T_д)^2$.

На втором этапе в качестве целевой функции используется функция вида

$\min(\lambda_м c_м M+\lambda_p c_p P(W, F))$, где: M-масса радиатора охлаждения, при этом в качестве варьируемых параметров используются геометрические параметры радиатора, скорость воздуха W и площадь обдува, от которых зависит производительность вентилятора P(W,F); $\lambda_м$ – весовой коэффициент важности уменьшения массы; λ_p – весовой коэффициент важности уменьшения производительности вентилятора; $c_м$ и c_p – стоимости единицы массы радиатора и производительности вентилятора соответственно. Вводятся также необходимые ограничения на варьируемые параметры.

Литература

1. Дульнев Г.Н. Тепло- и массообмен в радиоэлектронной аппаратуре.- М.: Высш. шк., 1984.

2. Банди Б. Методы оптимизации. Вводный курс: пер. с англ. - М.: Радио и связь.

Математическое моделирование синхронного спирального излучателя

С.С. Захарова, к.т.н.
Кафедра «Радиоэлектроника телекоммуникации», МИЭМ НИУ ВШЭ
szaharova@hse.ru

Улучшение характеристик радиоэлектронных устройств невозможно без разработки новых классов антенн. Все более широкое применение получают излучатели, работающие в режиме обратной волны. Одним из таких излучателей является синхронная спираль, она может функционировать в режимах с нормальной и аномальной дисперсией, т.е. как с прямыми, так и с обратными волнами [1]. В настоящей работе рассматривается моделирование диаграмм направленности излучателя, имеющего форму синхронной спирали.

Синхронная спираль представляет собой систему, которая характеризуются тем, что фазовая скорость пространственных гармоник радиально распространяющейся волны постоянна по радиусу и по азимуту спирали. Дифференциальное уравнение для описания синхронной спирали имеет вид [2]:

$$(M^2-1)r'^2+2R_0M\ r'-r^2+R_0^2=0\,, \qquad (1)$$

где n — номер пространственной гармоники; $M=\dfrac{c}{v_n}$ — коэффициент замедления (ускорения) n-ой пространственной гармоники, v_n — её радиальная фазовая скорость, c — скорость волны вдоль витков, $R_0=nTv$, r — радиус, соответствующий текущей точке на спирали.

Важной особенностью синхронных спиралей является то, что каждой пространственной гармонике соответствует своя плотность навивки, определяемая предельным радиусом R_0 и замедлением M.

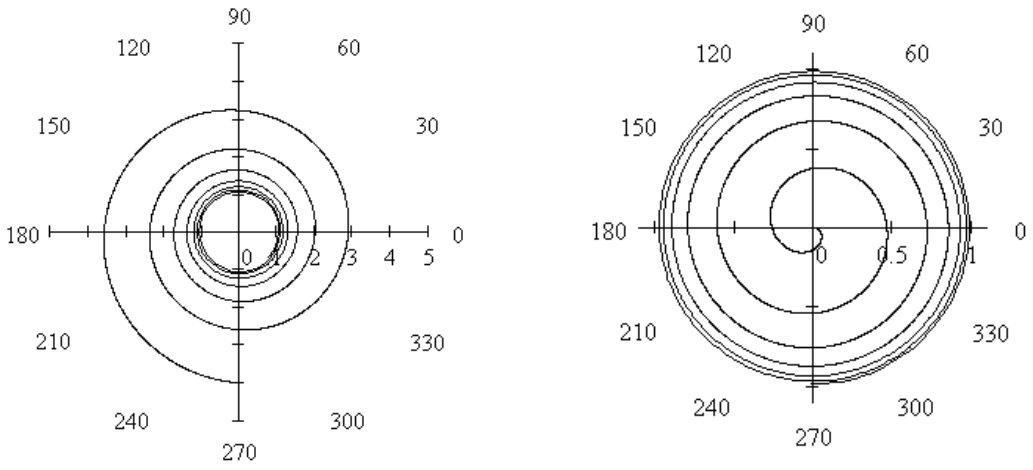
Решение дифференциального уравнения синхронной спирали, возбуждаемой замедленной волной, имеет вид:

$$\varphi=\pm\frac{1}{m}[\ln\left|-\rho+\sqrt{m^2+\rho^2}\right|\pm\sqrt{m^2+1}\ln\left|\frac{-\rho+\sqrt{m^2+\rho^2}\mp\sqrt{m^2+1}-1}{-\rho+\sqrt{m^2+\rho^2}\mp\sqrt{m^2+1}+1}\right|]_{\rho_0}^{\rho}\,, \qquad (2)$$

где $\rho=r/|R_0|$ — безразмерный радиус, ρ_0 — начальный радиус при $\varphi=0$, $m=\dfrac{1}{\sqrt{M^2-1}}$.

В зависимости от выбора знаков и значения радиуса могут быть получены разные виды синхронных спиралей. При выборе верхнего знака в уравнении (1) существует стационарный радиус, к которому спираль приближается снаружи, если $1<\rho<\infty$ или изнутри, если $0<\rho<1$. Рис. 1.

При нижнем знаке для $0<\rho<\infty$ спираль не имеет стационарного радиуса. Логарифмическая спираль является частным случаем синхронной спирали при $R_0\to 0$ т.е. $r_{стац}\to 0$. Направление навивки спирали будет определяться знаком произведения MR_0. Пример спиралей для замедленных волн приведен на рис. 1.



а) *б)*

Рис. 1. Синхронные спирали для случая замедленной волны со стационарным радиусом:
а) спираль навивается снаружи, M=10, $1 < \rho \le 4$;
б) спираль навивается изнутри, M=10, $0 \le \rho < 1$

Решение уравнения (1) для случая ускоренной волны имеет вид

$$\varphi = \mp \frac{1}{m}\left[2arctg\tau + \sqrt{m^2-1}\ln\left|\frac{\sqrt{m^2-1} \pm m + \tau}{\sqrt{m^2-1} \pm m - \tau}\right|\right]_{\rho_0}^{\rho}, \qquad (3)$$

где $m = \dfrac{1}{\sqrt{1-M^2}}$; $\tau = \dfrac{m}{\rho} + \sqrt{\left(\dfrac{m}{\rho}\right)^2 - 1}$.

В этом случае безразмерный радиус ρ может изменяться в пределах $0 < \rho < \infty$

Аналогично предыдущему, при выборе верхних знаков в уравнении (3) существует стационарный радиус. Если $1 < \rho < \infty$, то спираль навивается к этому радиусу снаружи, если $0 < \rho < 1$, — изнутри.

Расчет диаграммы направленности будем производить методом векторных потенциалов. Согласно этому методу, в случае монохроматических электромагнитных полей для определения поля по заданным источникам обычно вводят векторный электродинамические потенциал $\vec{A}$ [3]. Уравнение, связывающее напряженность электрического поля и векторный потенциал, имеет вид

$$\vec{E} = \frac{i}{\omega\varepsilon\mu}\,\mathrm{grad}\,div\vec{A} + i\omega\vec{A}. \qquad (4)$$

Уравнение (4) имеет решение

$$\vec{A} = \frac{\mu}{4\pi}\int_V \frac{\vec{j}e^{ikR}}{R}dV, \qquad (5)$$

где R — расстояние от элемента dV, в котором расположен элемент с плотностью тока $\vec{j}$, до точки наблюдения; $k = \omega\sqrt{\varepsilon\mu}$ — волновое число, ε — диэлектрическая проницаемость, μ — магнитная проницаемость, ω — угловая частота возбуждающего тока.

Для проведения моделирования поместим спираль в сферическую систему координат, как показано на рис. 2. Здесь P — точка наблюдения; Q — текущая точка на синхронной спирали.

Будем считать, что спираль выполнена из тонкого проводника и расположена в однородной изотропной среде. Пусть спираль возбуждается волной тока. В качестве

приближения будем считать, что амплитуда возбуждающего тока неизменна вдоль всей спирали.

При распределении тока $I(l)$ вдоль тонкого проводника с площадью поперечного сечения σ имеем $dV = \sigma dl$, плотность тока $\vec{j} = \frac{I(l)}{\sigma}\vec{l}_0$, где $\vec{l}_0$ — единичный вектор, направленный по касательной к проводнику, из которого намотана спираль.

Формула для векторного потенциала на основании соотношения (5) будет иметь вид:

$$\vec{A} = \frac{\mu}{4\pi}\int_l \frac{I(l)e^{ikR'}}{R'}d\vec{l}\ , \qquad (6)$$

где $d\vec{l} = \vec{l}_0 dl$.

В связи с тем, что точка наблюдения P находится в дальней зоне, и расстояние до точек, в которых определяется поле, велико по сравнению с радиусом витка, т.е. $R >> r$, то значение R' в знаменателе подынтегрального выражения (6) принимает значение, приближенно равное R, и тогда множитель $1/R$ можно вынести из-под знака интеграла.

В показателе экспоненты замена R' на R не производится, R' будем рассчитывать по формуле $\left|\vec{R}'\right| = \left|\vec{R}\right| - S$, где S — разность хода лучей, учитывающая запаздывание сферических волн, приходящих в точку P от элементарных источников, находящихся в начале системы координат, и в текущей точке интегрирования Q (рис. 2). Разность хода S может быть рассчитана по формуле $S = \left|\vec{R}\right| - \left|\vec{R} - \vec{r}\right| = \left|\vec{R}\right| - \left|\vec{R}'\right|$. В декартовой системе координат $\vec{r} = X\vec{x}_0 + Y\vec{y}_0 + Z\vec{z}_0$, где $\vec{x}_0$, $\vec{y}_0$, $\vec{z}_0$ — орты. Поскольку спираль лежит в плоскости XOY, то $\vec{r} = x\vec{x}_0 + y\vec{y}_0$, откуда имеем $\left|\vec{R} - \vec{r}\right| = \sqrt{(X-x)^2 + (Y-y)^2 + Z^2}$.

Для определения разности хода S используем сферическую систему координат. Для перехода к ней используем следующие формулы:

$$x = r\cos\varphi,\ \ y = r\sin\varphi,$$
$$X = R\sin\Theta\cos\Phi,\ Y = R\sin\Theta\sin\Phi,\ Z = R\cos\Theta. \qquad (7)$$

В сферической системе координат разность хода S определяется выражением

$$S = R - \sqrt{(R\sin\Theta\cos\Phi - r\cos\varphi)^2 + (R\sin\Theta\sin\Phi - r\sin\varphi)^2 + (R\cos\Theta)^2} =$$
$$= R\left(1 - \sqrt{1 - \frac{2r}{R}\sin\Theta\cos(\varphi - \Phi) + \frac{r}{R}}\right). \qquad (8)$$

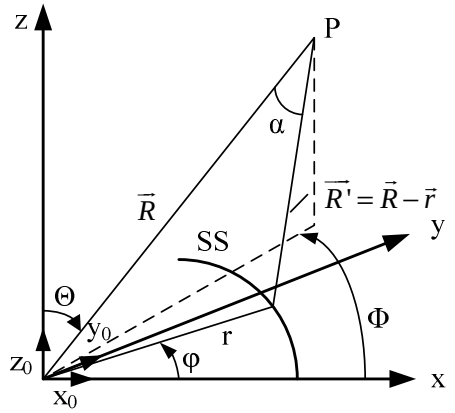


Рис. 2. К расчету разности хода волн S

В дальней зоне $\frac{r}{R} << 1$. Тогда, учитывая малость $\frac{r}{R}\sin\Theta$, формулу (8) с использованием соотношения $\left|\vec{R}'\right| = \left|\vec{R}\right| - S$ можно привести к виду

$$R' = R - r\sin\Theta\cos(\varphi - \Phi). \qquad (9)$$

Подставляя (9) в уравнение векторного потенциала (5), получим

$$\vec{A} = \frac{\mu}{4\pi}\frac{e^{ikR}}{R}\int_l I(l)e^{-ikr\sin\Theta\cos(\varphi-\Phi)}d\vec{l}\ . \qquad (10)$$

Волновое число возбуждающего тока, текущего в проводнике, образующего спираль, составляет $h_S = \omega / v_S$, где v_S — фазовая скорость волны тока вдоль проводника спирали. Ток, возбуждающий излучение электромагнитных волн, с учетом введенных приближений будет изменяться по закону $I(l) = I_0 e^{ih_S l}$. Подставляя это выражение в (10), получим

$$\vec{A} = \frac{\mu I_0}{4\pi}\frac{e^{ikR}}{R}\int_l e^{ih_S l - ikr\sin\Theta\cos(\varphi-\Phi)}d\vec{l}\ . \qquad (11)$$

Можно показать, что

$$d\vec{l} = \left(-r\sin\varphi + r'\cos\varphi\right)\vec{x}_0 d\varphi + (r\cos\varphi + r'\sin\varphi)\vec{y}_0 d\varphi. \qquad (12)$$

На основе приведенного разложения элементарных составляющих участков спирали можно получить разложение векторного потенциала по осям X и Y. С учетом того, что $\vec{A} = \vec{A}_x\vec{x}_0 + \vec{A}_y\vec{y}_0$, из уравнения (11) и выражения (12) следует, что

$$\begin{aligned} A_x &= \frac{\mu I_0}{4\pi}\frac{e^{ikR}}{R}\int_{\varphi_0}^{\varphi_1} e^{ih_S l(\varphi) - ikr\sin(\Theta)\cos(\varphi-\Phi)}(-r\sin\varphi + r'\cos\varphi)d\varphi, \\ A_y &= \frac{\mu I_0}{4\pi}\frac{e^{ikR}}{R}\int_{\varphi_0}^{\varphi_1} e^{ih_S l(\varphi) - ikr\sin(\Theta)\cos(\varphi-\Phi)}(r\cos\varphi + r'\sin\varphi)d\varphi. \end{aligned} \qquad (13)$$

Вводя обозначения

$$\begin{aligned} \Gamma_x &= \int_{\varphi_0}^{\varphi_1} e^{ih_S l(\varphi) - ikr\sin(\Theta)\cos(\varphi-\Phi)}(-r\sin\varphi + r'\cos\varphi)d\varphi, \\ \Gamma_y &= \int_{\varphi_0}^{\varphi_1} e^{ih_S l(\varphi) - ikr\sin(\Theta)\cos(\varphi-\Phi)}(r\cos\varphi + r'\sin\varphi)d\varphi, \end{aligned} \qquad (14)$$

представим уравнение для векторного потенциала в виде

$$\vec{A} = \frac{\mu I_0}{4\pi}\frac{e^{ikR}}{R}(\Gamma_x\vec{x}_0 + \Gamma_y\vec{y}_0). \qquad (15)$$

Переход к сферической системе координат для компонентов любого вектора может быть выполнен на основе следующих соотношений:

$$\begin{aligned} A_R &= A_x\sin\Theta\cos\Phi + A_y\sin\Theta\sin\Phi, \\ A_\Phi &= A_x\sin\Phi + A_y\cos\Phi, \\ A_\Theta &= A_x\cos\Theta\cos\Phi + A_y\sin\Theta\cos\Phi. \end{aligned} \qquad (16)$$

Векторный потенциал антенны на основе синхронной спирали в сферической системе координат определяется выражением

$$\vec{A} = \frac{\mu I_0}{4\pi}\frac{e^{ikR}}{R}(\Gamma_R\overrightarrow{R_0} + \Gamma_\Theta\overrightarrow{\Theta_0} + \Gamma_\Phi\overrightarrow{\Phi_0}). \qquad (17)$$

В (17) и далее $\overrightarrow{R_0}$ — единичный вектор, совпадающий по направлению с радиус-вектором $\vec{R}$, $\overrightarrow{\Phi_0}$ — единичный вектор, параллельный плоскости *XOY* и перпендикулярный вектору $\overrightarrow{R_0}$; $\overrightarrow{\Theta_0}$ — единичный вектор, перпендикулярный векторам $\overrightarrow{R_0}$ и $\overrightarrow{\Phi_0}$. Таким образом, орты $\overrightarrow{R_0}$, $\overrightarrow{\Theta_0}$ и $\overrightarrow{\Phi_0}$ образуют в любой точке пространства ортогональную систему.

Поскольку в дальней зоне распространяется сферическая волна [4], то электромагнитное поле не имеет продольных составляющих, и $\Gamma_R \equiv 0$ для всех точек дальней зоны. Отметим, что приведенные выше формулы справедливы для любой плоской кривой $r(\varphi)$, а не только для синхронной спирали [5].

Расчет компонентов электромагнитного поля в дальней зоне, формируемого синхронным спиральным излучателем, может быть выполнен с той же точностью, что и векторный потенциал, если составляющие векторов поля будут определены с точностью до членов, содержащих множитель 1/*R*.

В сферической системе координат $\vec{E} = E_\Theta \vec{\Theta}_0 + E_\Phi \vec{\Theta}_\Phi$, $\vec{H} = H_\Theta \vec{\Theta}_0 + H_\Phi \vec{\Theta}_\Phi$, откуда

$$\vec{E} = \frac{i\omega\mu I_0}{4\pi}\frac{e^{ikR}}{R}(\Gamma_\Theta \vec{\Theta}_0 + \Gamma_\Phi \vec{\Theta}_\Phi),$$
$$\vec{H} = \frac{ikI_0}{4\pi}\frac{e^{ikR}}{R}(\Gamma_\Theta \vec{\Theta}_0 - \Gamma_\Phi \vec{\Theta}_\Phi). \tag{18}$$

Учитывая, что

$$\Gamma_\Theta = \left(\Gamma_x \cos\Phi + \Gamma_y \sin\Phi\right)\cos\Phi,$$
$$\Gamma_\Phi = \Gamma_x \sin\Phi + \Gamma_y \cos\Phi, \tag{19}$$

из соотношений (14) получим следующие выражения для Γ_Θ и Γ_Φ:

$$\Gamma_\Theta = \cos\Theta \int_{\varphi_0}^{\varphi_1} e^{ih_s l(\varphi) - ikr(\varphi)\sin\Theta\cos(\varphi-\Phi)}[-r\sin(\varphi-\Phi) + r'\cos(\varphi-\Phi)]d\varphi,$$
$$\Gamma_\Phi = \int_{\varphi_0}^{\varphi_1} e^{ih_s l(\varphi) - ikr(\varphi)\sin\Theta\cos(\varphi-\Phi)}[r\cos(\varphi-\Phi) + r'\sin(\varphi-\Phi)]d\varphi, \tag{20}$$

где φ_0 и φ_1 — граничные углы навивки спиральной антенны.

Из условия синхронности волн в антенне при постоянстве фазовой скорости вдоль витков v_s имеем

$$\frac{\int_{\varphi_0}^{\varphi} r'(\varphi)d\varphi}{v_n} = \frac{\int_{\varphi_0}^{\varphi} \sqrt{r(\varphi)^2 + r'(\varphi)^2}\,d\varphi}{v_s} + \frac{nT_0}{2\pi}(\varphi - \varphi_0), \tag{21}$$

где $T_0 = \lambda_0 / c$ — период, соответствующий резонансной длине волны, для которой построена спираль. Скорость волны вдоль спирали принимается равной скорости света в свободном пространстве, т.е. $v_S = c$.

Интегрируя левую часть уравнения (21), получим

$$\frac{r(\varphi) - r_0}{v_n} = \frac{l(\varphi)}{v_s} + \frac{nT_0}{2\pi}(\varphi - \varphi_0), \tag{22}$$

где r_0 — радиус спирали, соответствующий начальному углу навивки спирали φ_0.

Умножая обе части (22) на частоту возбуждающего тока $\omega = kv_s = 2\pi v_s / \lambda$, с учетом того, что $h_S = \omega / v_S$ и $\omega = 2\pi v_s / \lambda$, получим

$$h_s l(\varphi) = \frac{\omega}{v_s} l(\varphi) = \frac{\omega}{v_n}(r(\varphi) - r_0) - n\frac{\lambda_0}{\lambda}(\varphi - \varphi_0), \qquad (23)$$

Подставляя (23) в (21) и переходя к интегрированию по r, получим

$$\begin{aligned} \Gamma_\Theta &= e^{-i\frac{\omega}{v_n}r_0 + in\frac{\lambda_0}{\lambda}\varphi_0} \cos\Theta \int_{r_0}^{r_1} e^{i[\frac{\omega}{v_n}r - n\frac{\lambda_0}{\lambda}\varphi(r)] - ikr\sin\Theta\cos(\varphi-\Phi)} [-\frac{r}{r'}\sin(\varphi-\Phi) + \cos(\varphi-\Phi)]dr, \\ \Gamma_\Phi &= e^{-i\frac{\omega}{v_n}r_0 - in\frac{\lambda_0}{\lambda}\varphi_0} \int_{r_0}^{r_1} e^{i[\frac{\omega}{v_n}r - n\frac{\lambda_0}{\lambda}\varphi(r)] - ikr\sin\Theta\cos(\varphi-\Phi)} [\frac{r}{r'}\cos(\varphi-\Phi) + \sin(\varphi-\Phi)]dr. \end{aligned} \qquad (24)$$

Выражения (24) для Γ_Θ и Γ_Φ являются наиболее общими для синхронных спиральных излучателей.

Анализ излучения синхронных спиральных излучателей для разных частот может быть выполнена на основе анализа соотношений (24). При заданной геометрии спирали, определяемой $\varphi(r)$, значения Γ_Θ и Γ_Φ зависят от углов Θ и Φ. Характер излучения синхронной спирали, т.е. режим её работы будет определяться соотношением текущего значения λ и резонансной длины волны λ_0, т.е. фактически соотношением λ и стационарного радиуса. При описании режимов работы будем использовать обозначение волн T_k, где число k приближенно показывает, какое количество волн укладывается вдоль окружности, соответствующей стационарному радиусу.

Для синхронной спирали с большим количеством витков могут быть рассмотрены четыре режима работы.

1. Низкочастотный режим соответствует $\lambda >> \lambda_0$, т.е. в этом режиме длина витка спирали и окружности стационарного радиуса существенно меньше длины волны возбуждающего тока. В спирали преобладает волна T_0, характеризующаяся изменением фазы тока в пределах 360^0 на протяжении нескольких витков спирали и фазовой скоростью порядка c. Поскольку соотношение длины волны и длины витков, например, при увеличении частоты возбуждающего сигнала будет изменяться в широких пределах, можно ожидать, что излучение будет носить ненаправленный характер с существенным изменением формы диаграммы направленности, которая будет определяться соотношением результирующих амплитуд в направлении оси спирали и в перпендикулярной плоскости. Диаграмма направленности для большинства длин волн имеет характер купола, поскольку взаимная компенсация излучения от разных участков спирали будет большей в её плоскости (рис. 3,а). Однако для некоторых частот интенсивность излучения в плоскости спирали будет выше, чем вдоль оси, и в этих случаях диаграмма направленности приобретает тороидальную форму, соответствующую минимуму излучения вдоль оси спирали.

2. Резонансный режим соответствует значениям $\lambda \approx \lambda_0$ и соответствует выполнению условия синхронности волн в витках спирали. При этом наблюдатся уменьшение отношения максимального значения диаграммы направленности к минимальному при таком же характере её формы, что связано с увеличением взаимокомпенсации составляющих излучения от разных точек спирали. Этот режим соответствует осевому излучению синхронной спирали, более равномерному, чем в первом случае. В синхронной спирали будет преобладать волна типа T_1, характеризующаяся сниженной фазовой скоростью, приближенно равной c. Волна T_1 будет интенсивно излучаться спиралью.

3. Режим $\lambda < \lambda_0$ характеризуется диаграммой направленности, имеющей несколько лепестков в плоскости, перпендикулярной плоскости расположения

спирали. В зависимости от соотношения λ и длины окружности стационарного радиуса в антенне будут наблюдаться волны T_2, T_3 и т.д. Для антенны, расположенной в свободном пространстве, будет наблюдаться четыре максимума излучения (рис. 3,б), что свойственно для многих спиральных антенн [6].

4. При существенном уменьшении длины волны относительно λ_0, т.е. в режиме $\lambda << \lambda_0$ характер диаграммы направленности меняется в направлении увеличения интенсивности излучения в плоскости спирали и снижения его интенсивности до очень малых значений в перпендикулярном плоскости спирали направлении. Это объясняется компенсацией излучения участков спирали вдоль её оси и невысоким уровнем таковой в других направлениях. Максимумы излучения станут выражены слабее и будут приближаться к плоскости, в которой расположена синхронная спираль (рис. 3,в).

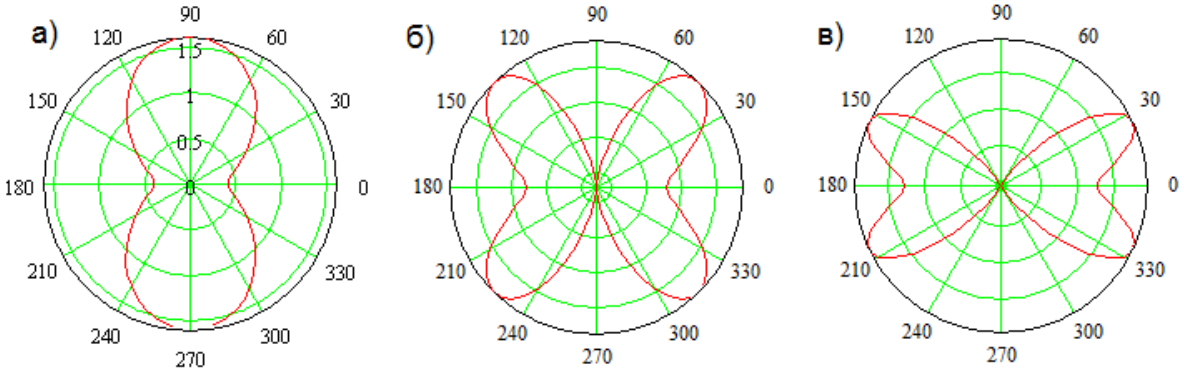


Рис 3 Диаграммы направленности синхронного спирального излучателя:
а) для $\lambda >> \lambda_0$, б) для $\lambda \approx \lambda_0$; в) для $\lambda << \lambda_0$

Таким образом, в представленной работе проведено математическое моделирование синхронного спирального излучателя и выявлены некоторые его свойства; рассчитаны компоненты электромагнитного излучения дальней зоны и рассмотрены основные режимы его работы.

Литература

1. Захарова С.С., Солнцев В.А. Анализ свойств антенн на основе синхронных спиралей как замедляющих структур. — Труды НИИР, №1, 2010. — с. 102-108.
2. Солнцев В.А. Плоские спиральные системы с постоянной радиальной фазовой скоростью. — Радиотехника и электроника, т.39, №4, 1994. — с 552-559.
3. Марков Г.Т., Сазонов Д.М. Антенны. Учебник для студентов радиотехнических специальных ВУЗов. — М.: Энергия, 1975. — 528 с.
4. Драбкин А.Л., Зузенко В.Л. Антенно-фидерные устройства. — М.: Советское радио, 1961. — 816 с.
5. Рамо С., Уиннери Дж. Поля и волны в современной радиотехнике. — М.: Гостехиздат, 1948. — 632 с.
6. Ротхаммель К. Антенны. — Серия «Массовая радиобиблиотека», вып.№637. — М.: «Энергия», 1967. — 272 с.

Моделирование диаграммы направленности излучателей, расположенных над металлическим экраном

С.С. Захарова, к.т.н.
Кафедра «Радиоэлектроника телекоммуникации», МИЭМ НИУ ВШЭ
szaharova@hse.ru

В настоящее время в антенной технике широко используются спиральные излучатели. Среди них наибольшее распространение получили арифметические и логарифмические спирали. В работе рассматривается построение математической модели для расчета диаграммы направленности синхронного спирального излучателя [1], расположенного на диэлектрической подложке над экраном.

На практике весьма часто возникает необходимость сконцентрировать излучение в части пространства. Для этого излучатель располагают на плоской диэлектрической подложке, с обратной стороны которой устанавливается экран с высокой проводимостью.

Рассмотрим синхронную спираль, расположенную на диэлектрической подложке над экраном, как показано на рис. 1, в целях получения соотношений для расчета диаграммы направленности такой системы. В настоящее время корректировку диаграммы направленности излучателя, расположенного на диэлектрической подложке над экраном, выполняют введением корректирующей функции [2]:

$$f_{корр}(\Theta) \approx \sin(kh\cos(\Theta)), \qquad (1)$$

где Θ — угол места, отсчитываемый от нормали к плоскости расположения антенны; h — толщина диэлектрической подложки; k — волновое число.

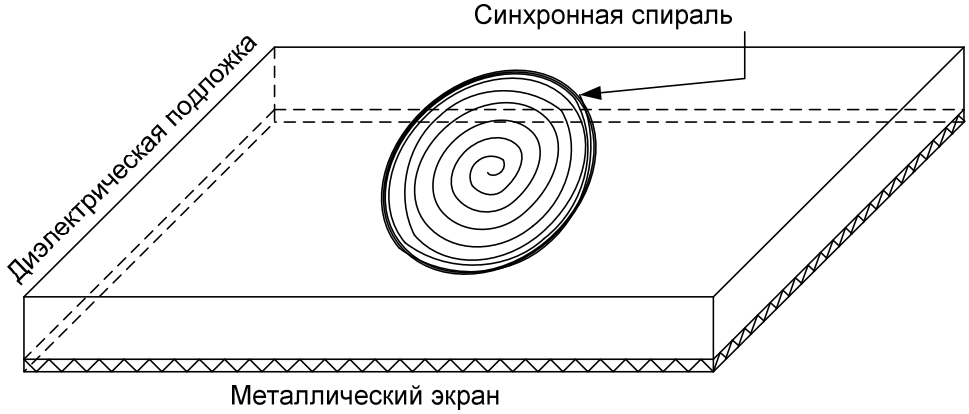


Рис. 1. Спиральная антенна на диэлектрической подложке над экраном

Уравнение (1) часто не может использоваться для практических целей, поскольку в нем не учтены многие существенные явления, которые влияют на форму диаграммы направленности. К таким явлениям следует отнести преломление волн электромагнитного излучения на границе раздела диэлектрика и свободного пространства, что приводит к изменению углового распределения интенсивности излучения в дальней зоне, и многократное отражение части электромагнитных волн от границы раздела диэлектрика и свободного пространства, что приводит к формированию поля в дальней зоне из совокупности парциальных диаграмм направленности.

Для учета этих явлений можно воспользоваться приближением геометрической оптики [3], которое является основой для вывода формул Френеля, описывающих преломление волн на границе раздела сред. Допустимость использования приближения геометрической оптики следует из радиального продвижения фазовых фронтов излучения каждой точки излучающей системы [4].

Согласно существующим представлениям о поле, создаваемом излучающей системой в дальней зоне, последнее является суперпозицией парциальных полей, формируемых элементами излучающей системы. В случае расположения спирали на диэлектрической подложке над отражающим экраном излучение, создаваемое в дальней зоне, в первом приближении будет складываться из двух составляющих. Первой из них является излучение собственно синхронной спирали (её будем называть материальной) в свободное от экрана полупространство. Второй составляющей будет излучение мнимой синхронной спирали, расположенной на поверхности диэлектрика. Это излучение будет обусловлено отражением электромагнитных волн от экрана. Такое представление о формировании поля в дальней зоне иллюстрирует рис. 2. Здесь показано направление положительного отсчета углов Θ и Φ, соответствующее использовавшемуся выше.

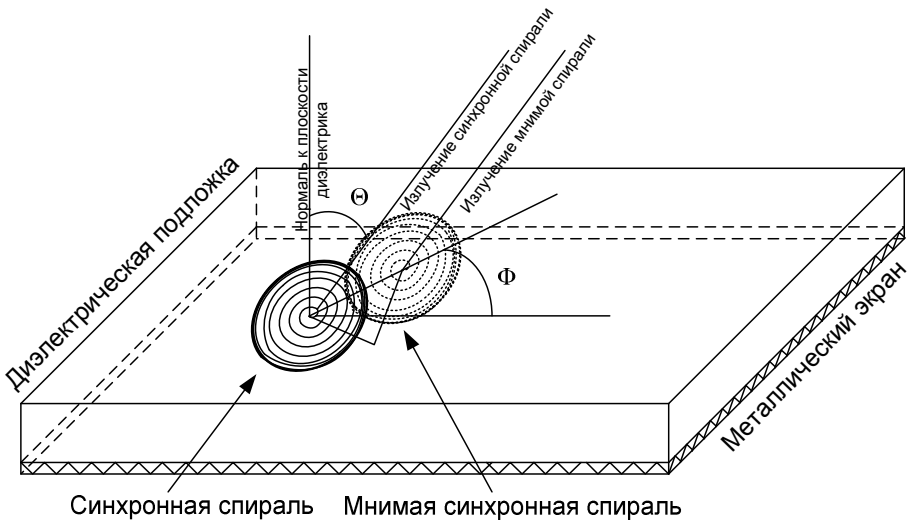


Рис. 2. Материальный и мнимый спиральные излучатели

Поскольку излучение мнимой спирали однозначно определяется излучением материальной спирали, то необходимо выяснить, как они связаны. Для того, чтобы установить такую взаимосвязь, необходимо доказать выполнение условий, которые позволяют использовать приближение геометрической оптики. Этим условием является наличие в излучающей системе точки, относительно которой фаза излучения будет постоянной для всех равноудаленных от неё точек в дальней зоне. Такая точка называется фазовым центром антенны. Наличие фазового центра у синхронного спирального излучателя, совпадающего с его геометрическим центром, было доказано в [5]. Наличие фазового центра в геометрическом центре синхронной спирали позволит ниже выполнить суммирование составляющих поля $\vec{E}_\Theta$, $\vec{E}_\Phi$ и $\vec{H}_\Theta$, $\vec{H}_\Phi$ в дальней зоне.

Для определения составляющих излучения мнимой синхронной спирали рассмотрим распространение излучения в диэлектрике в приближении геометрической оптики. Для произвольного угла Φ схема распространения лучей приведена на рис. 3;

ε_1 — относительная диэлектрическая проницаемость диэлектрика. При изотропности свойств диэлектрика направление распространения преломленной и отраженной электромагнитных волн определяется только углом наблюдения и свойствами сред.

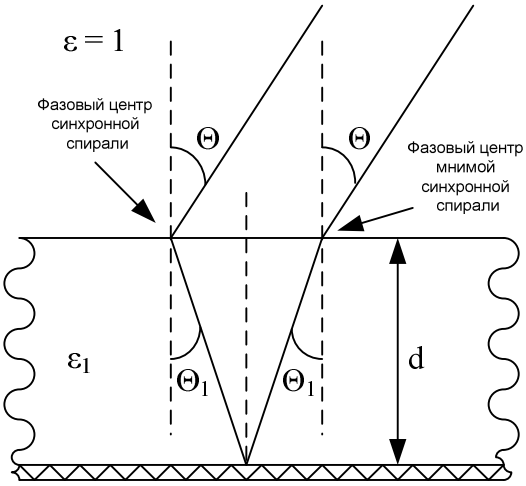


Рис. 3. Приближение геометрической оптики для расчета скорректированной диаграммы

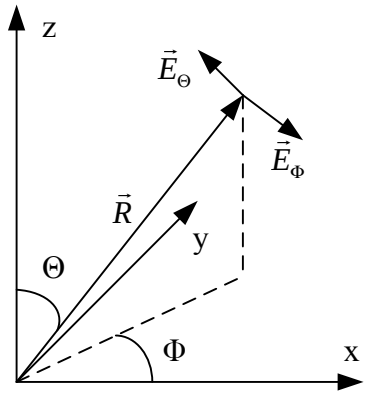


Рис. 4. Векторы $\vec{E}_\Theta$ и $\vec{E}_\Phi$ в дальней зоне направленности

Электромагнитные волны, формируемые синхронной спиралью, распространяются в диэлектрике, отражаясь от металлического экрана и преломляясь на границе раздела сред с отражением их части обратно в диэлектрик. Ход падающего и преломленного лучей описывается законом Снеллиуса:

$$\frac{\sin(\Theta_1)}{\sin(\Theta)} = \frac{n}{n_1} = \sqrt{\frac{\varepsilon}{\varepsilon_1}} = \frac{1}{\sqrt{\varepsilon_1}}, \qquad (2)$$

где n и n_1 — показатели преломления свободного пространства и диэлектрика. Из соотношения (2) следует, что излучение, формируемое в дальней зоне под углом Θ мнимой синхронной спиралью, обусловлено излучением физически существующей спирали, распространяющимся под углом Θ_1 относительно нормали к поверхности. Из выражения (2) $\Theta_1 = \arcsin\left(\sin(\Theta)/\sqrt{\varepsilon_1}\right)$. Поскольку электромагнитная волна в данном случае падает из оптически более плотной среды в оптически менее плотную, то для углов $\Theta_1 > \Theta_{1,KP} = \arcsin\left(1/\sqrt{\varepsilon_1}\right)$ будет наблюдаться полное отражение электромагнитных волн от границы раздела диэлектрика и свободного пространства.

Излучение, распространяющееся в диэлектрике из разных точек синхронной спирали под углом Θ_1 к нормали, приобретает фазу

$$\gamma = \frac{2\pi l}{\lambda_\varepsilon} = \frac{4\pi d}{\lambda_\varepsilon \cos(\Theta_1)} = \frac{4\pi f d\sqrt{\varepsilon_1}}{c\cos(\Theta_1)} = \frac{4\pi f d\varepsilon_1}{c\sqrt{\varepsilon_1 - \sin^2(\Theta)}}, \qquad (3)$$

где c — скорость света, f — частота возбуждающего излучение тока, λ_ε — длина волны излучения в диэлектрике. Исходя из этого, можно сделать качественный вывод о том, что увеличение излучения в дальней зоне в силу наличия металлического экрана будет наблюдаться для углов Θ, близких к нулю. С точки зрения формы диаграммы направленности это означает, что она будет «приподниматься» вверх.

Металлический экран обеспечивает практически полное отражение электромагнитного излучения. Рассмотрим некоторую точку, расположенную в дальней зоне, и составляющие вектора $\vec{E}$ в ней (рис. 4). Составляющая напряженности поля мнимой синхронной спирали $\vec{E}_{\Phi 1}$ параллельна плоскости

спирали, и формируется элементарными векторами напряженности, перпендикулярными плоскости падения (s-поляризация), изображенной на рис. 4, а составляющая $\vec{E}_{\Theta 1}$ — элементарными векторами напряженностями, лежащими в этой плоскости (p-поляризация).

Согласно существующим представлениям о формировании преломленных и отраженных лучей, коэффициенты отражения и преломления на границе двух сред описываются формулами Френеля. В нашем случае значения коэффициентов отражения $K_П$ и K_O для s- и p-поляризации могут быть записаны в виде:

— для s-поляризации:

$$K_{П,S} = \frac{1}{\varepsilon_1}\frac{2n_1\cos(\Theta_1)}{n_1\cos(\Theta_1)+n\cos(\Theta)} = \frac{1}{\sqrt{\varepsilon_1}}\frac{2\cos(\Theta_1)}{\sqrt{\varepsilon_1}\cos(\Theta_1)+\cos(\Theta)};$$
$$K_{O,S} = \frac{n_1\cos(\Theta_1)-n\cos(\Theta)}{n_1\cos(\Theta_1)+n\cos(\Theta)} = \left|\frac{\sqrt{\varepsilon_1}\cos(\Theta_1)-\cos(\Theta)}{\sqrt{\varepsilon_1}\cos(\Theta_1)+\cos(\Theta)}\right|; \quad (4)$$

— для p-поляризации:

$$K_{П,P} = \frac{1}{\varepsilon_1}\frac{2n_1\cos(\Theta_1)}{n\cos(\Theta_1)+n_1\cos(\Theta)} = \frac{1}{\sqrt{\varepsilon_1}}\frac{2\cos(\Theta_1)}{\cos(\Theta_1)+\sqrt{\varepsilon_1}\cos(\Theta)};$$
$$K_{O,P} = \frac{n\cos(\Theta_1)-n_1\cos(\Theta)}{n\cos(\Theta_1)+n_1\cos(\Theta)} = \left|\frac{\cos(\Theta_1)-\sqrt{\varepsilon_1}\cos(\Theta)}{\cos(\Theta_1)+\sqrt{\varepsilon_1}\cos(\Theta)}\right|. \quad (5)$$

В выражениях для коэффициентов преломления множитель $1/\varepsilon_1$ появляется ввиду того, что в диэлектрике напряженность поля в ε_1 раз меньше, чем в свободном пространстве. Из соотношений (4) и (5) видно, что коэффициенты отражения и прохождения являются функциями углов преломления и падения электромагнитных волн на границу раздела сред. Зависимость этих углов друг от друга определяется законом Снеллиуса, приведенным выше, и Θ_1 может быть легко выражен через полярный угол Θ. Исходя из этого, можно рассмотреть коэффициенты отражения и преломления как функции полярного угла Θ:

$$K_{П,S}(\Theta) = \frac{1}{\sqrt{\varepsilon_1}}\frac{2\cos\left(\arcsin\left(\sin(\Theta)/\sqrt{\varepsilon_1}\right)\right)}{\sqrt{\varepsilon_1}\cos\left(\arcsin\left(\sin(\Theta)/\sqrt{\varepsilon_1}\right)\right)+\cos(\Theta)}; \quad (6)$$

$$K_{O,S}(\Theta) = \left|\frac{\sqrt{\varepsilon_1}\cos\left(\arcsin\left(\sin(\Theta)/\sqrt{\varepsilon_1}\right)\right)-\cos(\Theta)}{\sqrt{\varepsilon_1}\cos\left(\arcsin\left(\sin(\Theta)/\sqrt{\varepsilon_1}\right)\right)+\cos(\Theta)}\right|; \quad (7)$$

$$K_{П,P}(\Theta) = \frac{1}{\sqrt{\varepsilon_1}}\frac{2\cos\left(\arcsin\left(\sin(\Theta)/\sqrt{\varepsilon_1}\right)\right)}{\cos\left(\arcsin\left(\sin(\Theta)/\sqrt{\varepsilon_1}\right)\right)+\sqrt{\varepsilon_1}\cos(\Theta)}; \quad (8)$$

$$K_{O,P}(\Theta) = \left|\frac{\cos\left(\arcsin\left(\sin(\Theta)/\sqrt{\varepsilon_1}\right)\right)-\sqrt{\varepsilon_1}\cos(\Theta)}{\cos\left(\arcsin\left(\sin(\Theta)/\sqrt{\varepsilon_1}\right)\right)+\sqrt{\varepsilon_1}\cos(\Theta)}\right|. \quad (9)$$

Согласно формулам Френеля и представлению о распространении электромагнитных волн, в произвольной точке дальней зоны основной вклад в напряженность электрического поля будет вноситься излучением в свободное от диэлектрика полупространство физически существующей спирали и её мнимого прототипа. Вместе с тем, теоретически можно рассмотреть большее число

переотражений от экрана и границы диэлектрика и свободного пространства, и вести, таким образом, речь об излучении многих мнимых спиралей, определяемых излучением физически существующей спирали.

Для такого случая амплитуды *n*-ых составляющих $\vec{E}_{\Phi n}$, $\vec{E}_{\Theta n}$ векторов $\vec{E}_{\Phi}$, $\vec{E}_{\Theta}$, формируемые излучением *n*-ой мнимой синхронной спирали, для фиксированного расстояния *R* от излучающей системы будут определяться выражениями

$$E_{\Phi n}(\Phi,\Theta) = K_{\Pi,S}(\Theta)\left(K_{O,S}(\Theta)\right)^{n-1} E_{\Phi}\left(\Phi, \arcsin\left(\sin(\Theta)/\sqrt{\varepsilon_1}\right)\right), \qquad (10)$$

$$E_{\Theta n}(\Phi,\Theta) = K_{\Pi,P}(\Theta)\left(K_{O,P}(\Theta)\right)^{n-1} E_{\Theta}\left(\Phi, \arcsin\left(\sin(\Theta)/\sqrt{\varepsilon_1}\right)\right). \qquad (11)$$

С учетом доказанного выше наличия фазового центра каждая из составляющих $\vec{E}_{\Phi n}$, $\vec{E}_{\Theta n}$ векторов $\vec{E}_{\Phi}$, $\vec{E}_{\Theta}$ будет иметь сдвиг фазы относительно амплитуд напряженностей $\vec{E}_{\Phi 0}$, $\vec{E}_{\Theta 0}$, формируемых в дальней зоне непосредственно синхронной спирали и имеющих амплитудные значения $E_{\Phi}(\Phi,\Theta)$ и $E_{\Theta}(\Phi,\Theta)$. Данный сдвиг фазы для любого типа поляризации составляет

$$\psi_n = n\gamma = n\frac{4\pi f d \varepsilon_1}{c\sqrt{\varepsilon_1 - \sin^2(\Theta)}}. \qquad (12)$$

Рассмотрим временную диаграмму амплитудных векторов какой-либо составляющей в некоторой точке дальней зоны (рис. 5). На рис. 5 отмечено направление вращения векторов с течением времени. За период векторы $\vec{E}_0, \vec{E}_1, ..., \vec{E}_n$ делают полный оборот. Поскольку на прохождение электромагнитного излучения через диэлектрик требуется определенное время, то результирующая фаза Ψ сдвига суммарного вектора $\vec{E}_{\Sigma}$ относительно нулевой составляющей будет отрицательной, за исключением случаев, когда значение γ достаточно велико.

Из рис. 5 и уравнений векторной алгебры [6] следует, что амплитуда результирующего вектора и результирующая фаза относительно $\vec{E}_0$ составят

$$E_{\Sigma} = \sqrt{\left(E_0 + \sum_{n=1}^{N} E_n \cos\left(n\frac{4\pi f d \varepsilon_1}{c\sqrt{\varepsilon_1 - \sin^2(\Theta)}}\right)\right)^2 + \left(\sum_{n=1}^{N} E_n \sin\left(n\frac{4\pi f d \varepsilon_1}{c\sqrt{\varepsilon_1 - \sin^2(\Theta)}}\right)\right)^2}, \qquad (13)$$

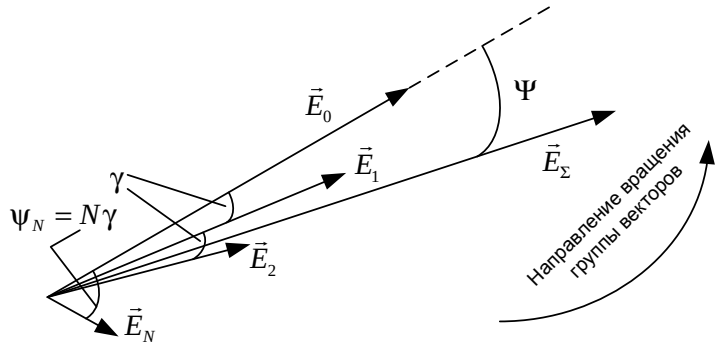


Рис. 5. Векторная диаграмма составляющих электрического поля в произвольной точке дальней зоны

В практике проектирования антенных устройств при построении диаграмм направленности допускается погрешность до 5…10 %. Расчеты с использованием формул (13) показали, что для обеспечения данной погрешности можно ограничиться значением
N = 5…6.

Необходимо отметить, что составляющие магнитного поля в дальней зоне будут преобразовываться по аналогичным формулам в силу их однозначной связи с напряженностью электрического поля через волновое сопротивление свободного пространства.

Рассмотрим, каким образом преобразуется диаграмма направленности антенны на основе синхронной спирали при размещении её над экраном. Как показано в работе [7], диаграмма направленности для антенны из витков, близких к стационарному радиусу, для большого их числа не зависит от азимутального угла Ф, и описывается выражением $ДН(\Theta) = \dfrac{|\Gamma_\Theta(\Theta)|^2 + |\Gamma_\Phi(\Theta)|^2}{\max\left(|\Gamma_\Theta(\Theta)|^2 + |\Gamma_\Phi(\Theta)|^2\right)}$, где

$$|\Gamma_\Theta| = |\cos\Theta| \int_0^{2\pi} e^{-ib\sin\Theta\cos\psi + ib\psi} \sin\psi d\psi,$$

$$|\Gamma_\Phi| = \int_0^{2\pi} e^{-ib\sin\Theta\cos\psi + ib\psi} \cos\psi d\psi.$$

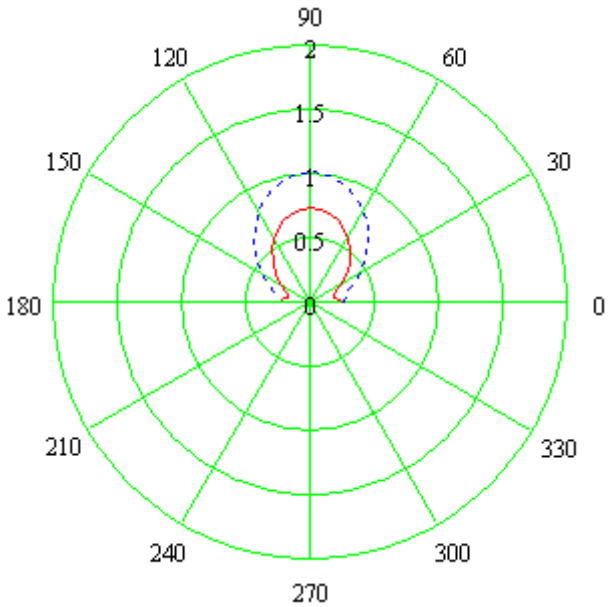


Рис. 6. Диаграммы направленности антенн на основе синхронной спирали без экрана в свободном пространстве (пунктир) и с экраном (сплошная линия)

Данные формулы описывают диаграмму направленности синхронной спирали для витков, близких к стационарному радиусу, в режимах осевого и двулучевого излучения. Здесь $b = nk/k_0 = n\lambda_0/\lambda$, $\lambda_0 = 2\pi R_n/n$, λ_0 — длина волны, для которой строится спираль. На рис. 6 в относительном масштабе показаны диаграммы направленности антенны на основе синхронной спирали, расположенной в свободном пространстве (пунктир) и над металлическим экраном для разных толщин диэлектрика. Частота возбуждающего воздействия составляет 1 ГГц, диэлектрическая проницаемость диэлектрика ε_1 = 6. Толщина диэлектрика составляет d = 0,025 м. Для исследования выбрана спираль со значением стационарного радиуса R = 0,01 м (λ_0 = 6,28 см). Для графиков на рис.6 спираль лежит в плоскости, расположенной перпендикулярно максимуму мощности излучения.

Точность результатов, получаемых с использованием предложенной методики, определяется использованными здесь приближениями. В частности, она не учитывает

диссипацию электромагнитной энергии при её прохождении через границу диэлектрика и свободного пространства.

Таким образом, в настоящей работе предложена методика определения диаграмм направленности для излучателей, в частности, имеющих геометрию синхронной спирали, расположенных над экраном и учитывающая особенности распространения волн в диэлектрике и их преломления для разных типов поляризации, что обуславливает её новизну и преимущества перед другими известными методиками.

Литература

1. Захарова С.С. Солнцев В.А. Расчет электромагнитного поля синхронной спирали, возбуждаемой волной тока. — Труды НИИР, №4, 2008. — с.51-54.
2. Юрцев О.А., Рунов А.В., Казарин А.Н. Спиральные антенны. — М.: Советское радио, 1974. — 224 с.
3. Григорьев А.Д. Электродинамика и микроволновая техника. Учебное пособие. — СП.-Б.: Лань, 2007. — 704 с.
4. Семенов А.А. Теория электромагнитных волн. Лекционный курс для радиофизиков. — М.: Издательство московского университета, 1962. — 256 с.
5. Захарова С.С. Методика расчета скорректированной диаграммы направленности при расположении синхронной спиральной антенны на диэлектрике над металлическим экраном. — Электромагнитная совместимость и проектирование электронных средств», сборник научных трудов. / Под ред. Кечиева Л.Н. — М.: МИЭМ, 2009. — с. 94 -105.
6. Ильин В.А., Позняк Э.Г. Аналитическая геометрия. — М.: Наука, 1971. - 232 с.
7. Захарова С.С. Расчет характеристик антенн на основе синхронных спиралей. — «Электромагнитная совместимость и проектирование электронных средств», сборник научных трудов. / Под ред. Кечиева Л.Н. — М.: МИЭМ, 2009. — с. 85-93.

Эффективная поверхность рассеяния, применимость и возможность ее определения для сложных целей

Т.А. Ясова
МИЭМ НИУ ВШЭ, Факультет Информационных Технологий
cuarto64@gmail.com

Эффективная площадь рассеяния (ЭПР; англ. Radar Cross-Section, RCS; в некоторых источниках — эффективная поверхность рассеяния,эффективный поперечник рассеяния, эффективная отражающая площадь, ЭОП) в радиолокации — площадь некоторой фиктивной идеальной сферы, являющейся идеальным и изотропным переизлучателем, которая, будучи помещена в точку расположения цели, создаёт у антенны радиолокационной станции ту же плотность потока мощности, что и реальная цель.

Грубо говоря, ЭПР – это некая количественная мера отражённой в направлении приёмника энергии электромагнитного излучения, падающего на цель.

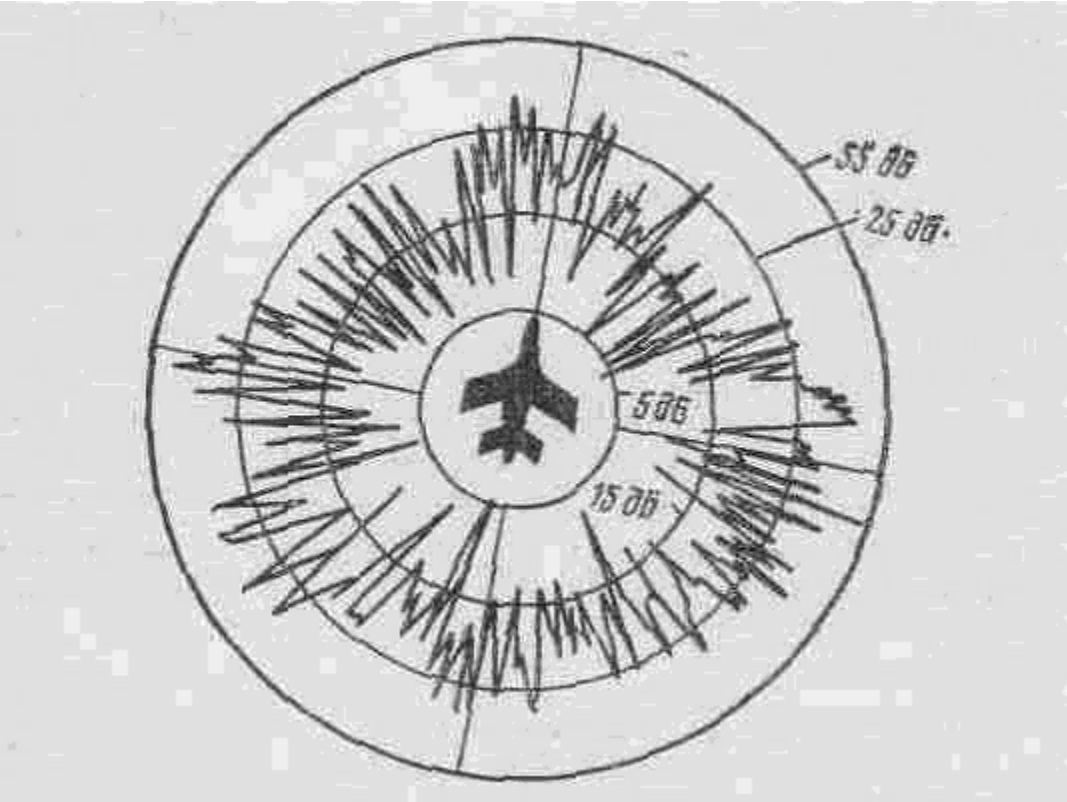


Рис. 1. Примерно так выглядит график ЭПР обычного самолета в зависимости от ориентации его к радару.

В первую очередь, конечно же, ЭПР определяет незаметность или, наоборот, видимость целей радарами. Чем меньшее ЭПР имеет цель, тем сложнее ее засечь радарами. На этом и основаны разработки технологии «Стелс». Но даже самолеты, спроектированные по этой технологии, не всегда «невидимы» для радаров. Чаще всего, когда самолет ориентирован к радару носом либо хвостовой частью, электромагнитное излучение отражается от него большей частью вниз, или в какую-нибудь другую сторону от радара, и тогда на радиолокационную станцию возвращается малозаметный сигнал. Но так происходит не всегда и не со всеми радарами. Во-первых, «Стелс» технология не рассчитана на поглощение энергии

излучения. То есть мы имеет дело не с «абсолютно черным телом» (определения этого термина для электродинамики взято из книги П. Я. Уфимцева «Теория дифракционных краевых волн в электродинамике»), а всего лишь телом, перенаправляющим эту энергию в отличное от направления ее получения сторону. Потому сразу видятся два способа обнаружения «Стелс» - по «тени» от объекта, который не пропускает электромагнитные импульсы в определённом направлении, и отслеживая отраженный сигнал. С обеими задачами способна справиться связная сеть приемников и передатчиков, более того, такие системы уже не только разработаны, но и существуют, и даже не представляют из себя что-то сложное и особо дорогое (по меркам военной промышленности, по заявленным данным стоимость подобной сети примерно в 7 раз меньше, чем стоимость одного единственного истребителя последнего поколения). Единственный минус – точность подобной сети, конечно же, будет меньше, чем у обычных радиолокационных станций. Но порог практической применимости так же легко преодолим, легко достижима точность в несколько метров.

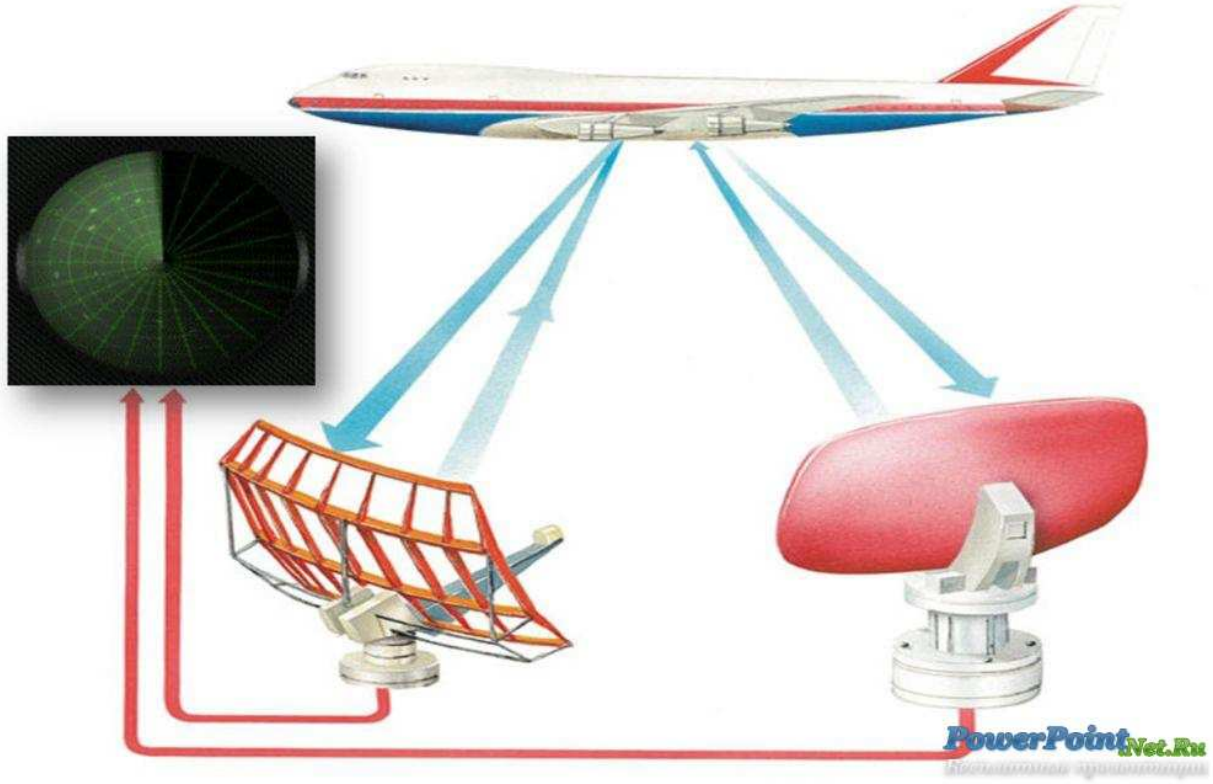


Рис. 2. Пример системы радаров

Кроме того, возможны и такие сценарии: радиолокационной сетью могут служить и сами истребители, если создать систему для согласованного действия их радаров. Тогда облучать «Стелс» будет один истребитель, а получать отраженный сигнал – второй, находящийся в другой точке пространства. И по его точным данным уже может осуществляться наводка и атака. Либо, цель возможно облучать с земли, чтобы истребители так же ловили уже отраженный в их сторону сигнал.

Все эти технологии требуют обмена данными между истребителями и/или наземными радиолокационными станциями, но в виду относительно близкого расстояния между ними возможно использование узконаправленных волн, либо оптического канала во избежание увеличения их заметности для пассивных РЛС.

Но кроме рассмотренного случая со «Стелс» технологиями, эффективная площадь рассеяния используется и в других областях. К примеру, важную роль играет определение ЭПР для ракет и противоракет, и здесь точность и быстрота определения целей имеет ключевую роль для эффективности и тех, и других.

Итак, эффективная площадь рассеяния – крайне важная характеристика как для проектирования оборудования/аппаратов, так и для разработки стратегий радиолокации. Одним из способов моделирования является метод цифровое моделирование входных сигналов систем радиолоквции от сложных радиолокационных схем, разработанный А.Б.Борзовым, А.В.Соколовым и В.Б.Сучковым. Они разработали метод расчёта ЭПР для ближней радиолокации, что применимо и для дальней, поскольку для этого нужно лишь считать фронт падающей волны плоским. Предельное расстояние для такой аппроксимации определяете по формуле:

$$R_{\min} \geq \frac{4L^2}{\lambda},$$

где λ - длина волны падающего излучения, $R_{\min}$ - предельное расстояние между СБРЛ и объектом.

В качестве энергетической характеристики рассеяния объекта в дальней зоне локации как раз используется эффективная площадь рассеяния (ЭПР), которая определяется для сосредоточенного объекта при облучении его плоской электромагнитной волной:

$$\sigma = \lim_{R\to\infty} 4\pi R^2 \frac{\left|\mathrm{E}_{scat}\mathrm{E}_{s0}\right|^2}{\left|\dot{E}_i\right|^2}$$

где R – расстояние от фазового центра антенны до объекта, $\dot{E}_i$ - величина напряженности поля, падающего на объект, E_{s0} - вектор поляризации приемной антенны в дальней зоне локации, E_{scat} - вектор напряженности электрического поля рассеяния объекта в точке приема.

В связи с этим, возник вопрос о том, что считать ЭПР объекта в ближней зоне локации, когда соотношение не выполнимо. Для решения указанных проблем потребовалось создание единого методического подхода, позволяющего наиболее адекватно описывать процесс электромагнитного взаимодействия СБРЛ и объекта как в ближней, так и в дальней зоне локации.

Одним из приоритетных направлений совершенствования принципов и методов радиолокационного обнаружения объектов на фоне подстилающих поверхностей и при наличии помех в СБРЛ является разработка новых более совершенных методов математического и цифрового моделирования рассеяния электромагнитных волн в ближней зоне локации.

Это осуществимо благодаря возможности представить модель исследуемого объекта в виде полигональной сетки с такими фрагментами, каждый из которых удовлетворял бы критерий дальней зоны:

$$L_m \leq \frac{\sqrt{R_m \lambda}}{2},$$

где L_m - характерный линейный размер m-го элемента полигональной модели, R_m - расстояние между фазовым центром антенны СБРЛ и m-м элементом. Пример подобной полигональной модели можно увидеть на рис. 3, рис. 4.

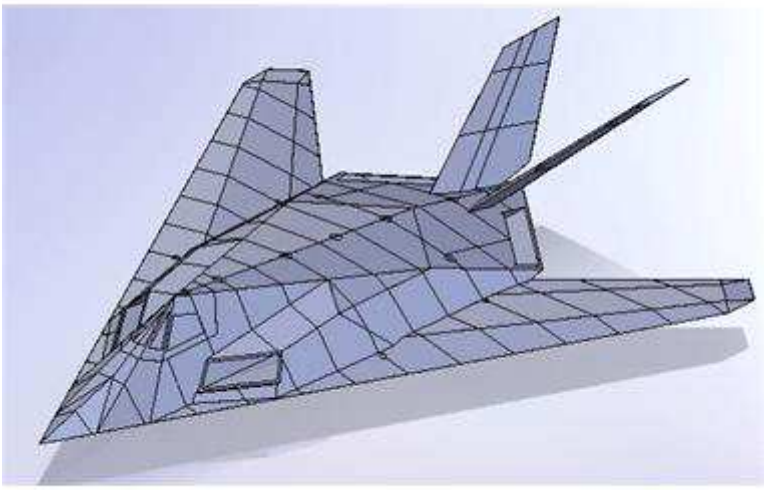


Рис 3. Полигональная модель истребителя F-117

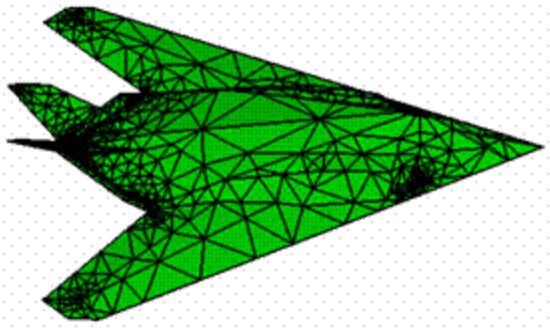


Рис. 4. Полигональная модель истребителя F-117, описанная с помощью треугольных полигонов

При представлении модели подобным образом, возможен расчёт величины результирующей напряженности электрического поля рассеяния, принятого в апертуре приемной антенны СБРЛ как когерентная сумма напряженности для каждого отдельного сегмента полигональной модели:

$$\dot{E}_{scat\Sigma} = \sum_{m=0}^{N_\Sigma - 1} \dot{E}_{sm} = \sqrt{\frac{P_i D_i Z_0}{2\pi}} \sum_{m=0}^{N_\Sigma - 1} F_{im} \dot{\rho}_{sm} \frac{\exp(i(kR_{im} + \omega t + \Phi))}{R_{im} R_{sm}} .$$

где R_{im} - расстояние между фазовым центром антенны и началом системы координат ЭО (элементарного отражателя), Pi – мощность сигнала, излучаемого передающей системой, Di – КНД передающей антенны в направлении максимума излучения, Z0=120πОм – волновое сопротивление воздуха, $\omega = 2\pi f$ - круговая частота электромагнитных колебаний, $\overline{F_{im}}$ - векторная характеристика направленности передающей антенны в направлении на ЭО, Φ - начальная фаза передающей антенны, R_{sm} - расстояние между ЭО и фазовым центром приемной антенны, $\dot{E}_{sm}$ -

величина напряженности поля рассеяния m-м ЭО. Индекс i соответствует облучающей волне, а s – вектору поляризации приемной антенны.

$\dot{\rho}_{sm}$ рассчитывают по формуле:

$$\dot{\rho}_{sm} = (\dot{\rho}^1_{1m}\dot{A}^i_{1m} + \dot{\rho}^1_{2m}A^i_{2m})\dot{A}^s_{1m} + (\dot{\rho}^2_{1m}\dot{A}^i_{1m} + \dot{\rho}^2_{2m}A^i_{2m})A^s_{2m} .$$

где фигурируют элементы матрицы рассеяния и амплитуда волны.

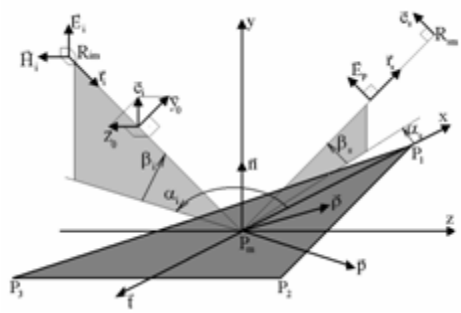


Рис. 5. Схема радиолокации плоского треугольного элемента полигональной модели.

Полный вывод формул приводится в работе [3]

Список литературы

1. Справочник по радиолокации в 4-х томах. Перевод с английского под общей редакцией К.Н. Трофимова, Москва, «Советское радио», 1976 г.
2. Теория дифракционных краевых волн в электродинамике. Уфимцев П. Я. Москва, Бином. Лаборатория знаний, 2012 г.
3. А.Б.Борзов, А.В.Соколов, В.Б.Сучков. Цифровое моделирование входных сигналов систем ближней радиолокации от сложных радиолокационных сцен // Журнал радиоэлектроники, N 4, 2004.
4. Математическая модель рассеяния электромагнитных волн на объектах сложной формы. Борзов А.Б., Соколов А.В. Электромагнитные волны & электронные системы. 1998 г.
5. Эффективная площадь отражения сложных радиолокационных целей. Юсеф Н.Н. ТИИЭР. 1989 г.

Неопределенность результатов виртуальной сертификации РЭС по эмиссии излучаемых радиопомех

Н.В. Лемешко, к.т.н.
Кафедра «Радиоэлектроника телекоммуникации», МИЭМ НИУ ВШЭ
nlemeshko@hse.ru, м.т. 8-915-231-17-47

Введение

С повышением пространственной насыщенности радиоэлектронными средствами (РЭС) качественное и своевременное решение задачи ЭМС приобретает все большее значение. Исследования, проводимые в части индустриального радиошума, показали возможность лишь статистического прогнозирования электромагнитного фона в выбранной местности [1, 2]; таким образом, в настоящее время отсутствует возможность точного прогнозирования создаваемой совокупностью РЭС электромагнитной обстановки.

Согласно современной практике, формируемое радиоэлектронным средством электромагнитное поле считается приемлемым, если при испытаниях, проводимых в условиях, соответствующих [3], измеренный уровень его напряженности не превысит норм, установленных продуктовыми стандартами, например, [4]. Такие испытания выполняются аккредитованными лабораториями [5] в обязательном порядке для всех РЭС, выпускаемых серийно, а также для специальных технических средств.

Если в процессе проведения испытаний окажется, что помехоэмиссия превышает установленные нормы, то РЭС отправляют на доработку. На момент изготовления первого опытного образца разработчик не имеет гарантии успешного прохождения сертификационных испытаний, которые вместе с доработкой могут приобрести цикличность, грозящую финансовыми потерями и срывом сроков вывода продукции на рынок.

Анализ этого противоречия, а также порядка проведения и технической стороны сертификационных испытаний позволяют прийти к выводу, что последние могут быть переведены в виртуальную плоскость, что позволит оценивать результаты испытаний непосредственно на стадии проектирования.

Сущность виртуальной сертификации

Виртуальная сертификация предполагает моделирование ряда физических процессов, соответствующих ходу лабораторных испытаний. Ими являются:

— протекание токов в проводниках РЭС, вызывающих излучение;

— распространение электромагнитных волн от проводников к точке наблюдения при их взаимодействии с элементами конструкции РЭС;

— формирование результирующего поля в точке наблюдения как векторной функции времени;

— формирование показаний измерительного приемника при поступлении на вход напряжения, соответствующего воспринимаемой измерительной антенной напряженности электромагнитного поля.

Методики построения математической модели, выявленные их особенности и ограничения подробно изложены в монографии [6], поэтому они не требуют конкретизации и раскрытия содержания. Важно отметить, что как для лабораторных, так и для виртуальных сертификационных испытаний существует множество факторов, определяющих погрешность получаемых результатов. Если рассматривать

моделирование сертификационных испытаний в укрупненном формате, то для каждого типа используемых моделей будут характерны свои погрешности. Исследования электрической модели измерительного приемника, электрической и электродинамической модели РЭС, а также общего модельного представления формирования результирующего поля показали, что для погрешностей, соответствующих каждому влияющему фактору, могут быть оценены максимальные значения. Для виртуальной сертификации следует рассмотреть проблему неопределенности результатов моделирования и привести количественную ее оценку по аналогии с лабораторными испытаниями.

Источники неопределенности измерений в области ЭМС и ее численная оценка

В настоящее время неопределенность измерений в области ЭМС оценивается в соответствии с методикой, изложенной в стандарте [7], который распространяется на измерения кондуктивных помех на портах электропитания, мощности помех и напряженности электрической составляющей электромагнитного поля излучаемых радиопомех при измерениях на открытой испытательной площадке. В последнем случае предписано учитывать следующие потенциальные источники неопределенности:

— погрешность показаний измерительного приемника (ИП);
— затухание в измерительном кабеле, включенном между ИП и антенной;
— коэффициент калибровки антенны;
— точность измерения синусоидального напряжения с помощью ИП;
— амплитудное соотношение и импульсную характеристику ИП;
— минимальный уровень собственных шумов ИП;
— рассогласование между входом ИП и измерительной антенной;
— зависимость коэффициента калибровки от высоты подъема антенны;
— пространственную избирательность антенны и положение ее фазового центра;
— восприимчивость антенны к полям с ортогональной поляризацией;
— симметричность измерительной антенны;
— затухание измерительной площадки;
— расстояние между испытуемым оборудованием и измерительной антенной;
— высоту стола, на котором размещено испытуемое оборудование.

В качестве основного показателя используют расширенную неопределенность, рассчитываемую по формуле $\Delta = 2\Delta_C$, где Δ_C — суммарная неопределенность измерений. Для определения необходимости корректировки результатов измерений значение Δ сравнивают с базовым, принятым равным Δ_S = 5,2 дБ, причем корректировку проводят в случае, когда $\Delta > \Delta_S$, увеличивая расчетные показания измерительного приемника на разность этих величин.

Суммарную стандартную неопределенность измерений определяют по формуле

$$\Delta_C = \sqrt{\sum_{i=1}^{N} c_i^2 \Delta_i^2}\ , \qquad (1)$$

где Δ_i и c_i — стандартная неопределенность, дБ, и коэффициент влияния для i-ой входной величины. В стандарте [7] под входными величинами понимаются любые численные параметры, погрешности которых требуют учета при вычислении значения Δ_C.

Стандартные неопределенности Δ_i рассчитывают на основе характерных значений относительных погрешностей δ_i, присущих данной входной величине и измеряемых в децибелах. При этом учитывают статистический тип распределения значений Δ_i характеризуемый коэффициентом k_i. Значения δ_i рассчитывают по формуле

$$\Delta_i = \delta_i / k_i . \qquad (2)$$

Для испытаний на открытой измерительной площадке рассматривают входные величины, соответствующие приведенному выше перечислению источников неопределенности.

Неопределенность результатов моделирования сертификационных испытаний

Совокупность потенциальных источников неопределенности задается используемыми принципами моделирования. Поскольку оно обладает свойством идеальности, то многие из источников неопределенности теряют смысл либо не уже не выступают в качестве таковых. Например, в разработанном методе моделирования [6] измерительная антенна рассматривается как устройство, выполняющее преобразование напряженности поля во входное напряжение ИП и не имеющее собственных погрешностей.

Однако при моделировании сертификационных испытаний следует учитывать ряд других факторов, от которых зависит расширенная неопределенность результатов моделирования Δ_M. При реализации алгоритма моделирования, аналогично проведению лабораторных исследований, это значение сравнивают с базовым, равным 5,2 дБ, и в случае, если $\Delta_M > \Delta_S$, показания ИП по модели перед сравнением с нормами увеличивают на разность этих величин. Значение Δ_M рассчитывается с использованием формул, аналогичных приведенным выше.

Основными источниками неопределенности результатов моделирования сертификационных испытаний являются:

1) расчет токов, возбуждающих излучение, связанная с несовершенством моделей компонентов и численных методов средств моделирования;

2) использование частотно-ограничивающих фильтров;

3) приближение, согласно которому элементы декомпозиции рассматриваются как участки с током, одинаковым по всей их длине;

4) отбор части проводников РЭС при расчете помехоэмиссии;

5) определение положения точки наблюдения;

6) свойства измерительной площадки;

7) свойства модели измерительного приемника;

8) взаимное расположение элементов конструкции РЭС.

Приведенная здесь нумерация источников неопределенности используется ниже, в т.ч. в таблице 1. Важно отметить, что этот перечень может быть дополнен и другими источниками неопределенности, но они будут оказывать значительно меньшее влияние на результат расчета Δ_M (например, погрешность диэлектрической проницаемости корпуса РЭС).

Погрешность расчета токов при схемном моделировании обычно не превосходит 10…15% [8], что соответствует δ_1 = 0,8…1,2 дБ. Для особо сложных схем значение собственной погрешности моделирования может достигать больших значений. Для этого источника неопределенности примем значение k_1 = 1.

Использование частотно-ограничивающих фильтров при расчете токов в излучающих элементах приводит к возникновению погрешности, оцененной в [6] на уровне 0,1…1 %, что соответствует значениям $\delta_2 < 0{,}1$ дБ. Данная погрешность должна рассматриваться как родственная предыдущей, и, следовательно, ей целесообразно сопоставить значение $k_2 = k_1 = 1$.

Погрешность, вызванная неравномерностью распределения тока в элементах декомпозиции, определяется текущей частотой анализа f_{max} и выбранной максимальной длиной элемента продольной декомпозиции l_d. На практике следует использовать значения сопутствующей приближению коротких антенн погрешности $\tau = 0{,}001..0{,}01$, определяющие компромисс между потенциальным снижением точности и количеством элементов декомпозиции, что соответствует погрешности $\delta_3 <$ 0,1 дБ. Примем $k_3 = 1$.

Следующий источник неопределенности может быть учтен через коэффициент, используемый при отборе проводников для анализа помехоэмиссии. Естественно, что вклад неучтенных проводников в формирование помехового излучения будет зависеть от форм спектров текущих в них токов. Следовательно, распределение проводников по интенсивности излучения может быть определено точно только для каждого конкретного РЭС. При этом возможны случаи, когда, согласно результатам формального анализа, значительная часть проводников требует учета и тогда указанная функция распределения будет похожа на относящуюся к нормальному распределению. В других случаях она может быть соотнесена с другими типами статистических распределений [9]. Если принять распределение проводников по интенсивности излучений (по коэффициентам, полученным на основе формального анализа) равномерным, то в случае, если из N проводников для анализа отобраны N_0 при коэффициенте отбора k_o, то рассматриваемая относительная погрешность составит $\delta_4 = \dfrac{(N - N_0)k_o}{N_o(1 - k_o)}$. Из статистических соображений имеем $(N - N_0)/N \approx k_0$. Отсюда имеем $\delta_4 \approx k_o^{\,2}/(1 - k_o)$.

Экспериментально показано, что рекомендуемое значение $k_o = 0{,}001….0{,}01$. Это соответствует значениям относительной погрешности $10^{-6}...10^{-4}$. Поскольку функция распределения зависит от схемных решений РЭС, частоты анализа и характеристик используемых сигналов, то при ориентировочных расчетах неопределенности результатов моделирования следует использовать большее значение, поэтому примем $\delta_4 = 0{,}3$ дБ и установим $k_4 = 1$.

Положение точки наблюдения определяется с использованием методики, согласно которой рассчитывают приближенное распределение показаний измерительного приемника с выбранным типом детектора при варьировании положения измерительной антенны с некоторым шагом по углу и по высоте подъема. Соответствующие им линейные перемещения точек наблюдения приближенно равны. Поэтому максимальная погрешность будет соответствовать случаю, когда за точку наблюдения принимается точка в распределении, расположенная диагонально. Таким образом, максимальная абсолютная погрешность определения положения точки наблюдения составит $\sqrt{2}\Delta z_1$. Максимальное расстояние, в пределах которого может меняться по диагонали положение точки наблюдения на развертке, составляет $\sqrt{2}\Delta h$, где Δh — интервал варьирования высоты измерительной антенны при измерениях. Следовательно, максимальная относительная погрешность определения положения точки наблюдения составит $\Delta z_1 / \Delta h$. Минимальное значение Δh следует принять

равным 3 м, что соответствует измерительным расстояниям 3 и 10 м. Оценочное значение погрешности соответствует максимальному значению шага, который на втором, уточняющем этапе алгоритма принят равным $\Delta z_1 = 0{,}04$ м. Отсюда имеем погрешность, равную 1,3%, что соответствует 0,1 дБ. Поскольку методика определения положения точки наблюдения является приближенной, целесообразно это значение несколько увеличить. Примем $\delta_5 = 0{,}3$ дБ и установим $k_5 = 1$.

При оценке погрешности, вызванной измерительной площадкой как источником неопределенности, следует обратиться к стандарту [7], согласно таблицам А.4 — А.7 которого $\delta_6 = 4$ дБ и $k_6 = 3$. Максимальная погрешность модели ИП, согласно результатам экспериментальных исследований, составляет $\delta_7 = 2{,}3$ дБ, значение коэффициента, учитывающего вид функции распределения, принимается равным $k_7 = 2$ [7].

Погрешность, учитывающая взаимное расположение элементов конструкции, требует отдельного анализа, поскольку даже в пределах партии РЭС будут наблюдаться различия, например, в положении проводов и шлейфов внутри приборов. С этой точки зрения печатные платы обладают значительно меньшим расхождением в конструкции, в особенности при автоматизированном производстве и монтаже компонентов. Поскольку информация о влиянии взаимного расположения элементов на результаты виртуальной сертификации на текущий момент отсутствует, то следует задать его приближенное значение. В данном случае примем $\delta_8 = 0{,}5$ дБ и $k_8 = 1$.

Согласно [7], значения коэффициентов влияния c_i для всех источников неопределенности при измерениях на открытой площадки приняты равными единице. Это же значение следует использовать и для источников неопределенности результатов моделирования сертификационных испытаний. Значения характерных для источников неопределенности параметров сведены в таблицу 1.

Таблица 1. Значения характеристик для источников неопределенности, свойственных моделированию сертификационных испытаний РЭС по эмиссии излучаемых радиопомех

Источник неопределенности	δ_i, дБ	k_i	Δ_i , дБ	c_i
1. Расчет токов в проводниках РЭС	0,8…1,2	1	0,8…1,2	1
2. Использование частотно-ограничивающих фильтров	0,1	1	0,1	
3. Неравномерное распределение тока в элементах декомпозиции	0,1	1	0,1	
4. Учет части проводников при расчете помехоэмиссии	0,3	1	0,3	
5. Определение положения точки наблюдения	0,3	1	0,3	
6. Свойства измерительной площадки	4,0	3	1,3	
7. Свойства модели измерительного приемника	2,3	2	1,2	
8. Взаимное расположение элементов конструкции РЭС	0,5	1	0,5	

На основе данных таблицы 1 в соответствии с формулами (1) и (2) можно рассчитать суммарную стандартную неопределенность моделирования сертификационных испытаний. Она составляет 2,05…2,25 дБ. Соответственно, расширенная неопределенность будет иметь ориентировочное значение 4,1…4,5 дБ.

Расчетный интервал значений расширенной неопределенности меньше, чем 5,2 дБ, поэтому в большинстве случае результаты моделирования перед сравнением с нормами помехоэмиссии не будут требовать корректировки.

Заключение

Таким образом, в рамках проделанной работы выполнена оценка характерной неопределенности результатов моделирования сертификационных испытаний РЭС по эмиссии излучаемых радиопомех. В случае каких-либо дополнительных особенностей процесса моделирования, а также при наличии уточняющей информации расширенная неопределенность должна рассчитываться непосредственно для конкретной реализации алгоритма моделирования сертификационных испытаний.

Важно отметить, что совокупность факторов, влияющих на неопределенность результатов измерений, допускает исключение либо уменьшение значений некоторых составляющих погрешности при совершенствовании методологии моделирования сертификационных испытаний. В частности, использование теории дифракции и распространения радиоволн может позволить снизить погрешность расчета результирующего поля в точке наблюдения за счет уточнения модели измерительной площадки. Такие усовершенствования следует отнести к последующему развитию методологии виртуальной сертификации.

Литература

1. Уильямс Т. ЭМС для разработчиков продукции. — Пер. с англ. Кармашева В.С., Кечиева Л.Н. — М.: Издательский дом «Технологии», 2003. — 540 с.

2. Рекомендация МСЭ-R P.372-9 Радиошум. — Женева, 2004.

3. ГОСТ 51320-99 «Совместимость технических средств электромагнитная. Радиопомехи индустриальные. Методы испытаний технических средств — источников индустриальных помех» — М.: Издательство Стандартов, 2000. — 39 с.

4. ГОСТ Р 51318.22-2006 «Совместимость технических средств электромагнитная. Оборудование информационных технологий. Радиопомехи индустриальные. Нормы и методы измерений» — М.: Стандартинформ, 2007. — 60 с.

5. ГОСТ 29037-91 «Совместимость технических средств электромагнитная. Сертификационные испытания. Общие положения» — М.: Издательство Стандартов, 1991. — 7 с.

6. Лемешко Н.В. Теоретические основы моделирования сертификационных испытаний радиоэлектронных средств по эмиссии излучаемых радиопомех. Монография. — М.: МИЭМ, 2012. — 196 с.

7. ГОСТ Р 51318.16.4.2-2006 «Совместимость технических средств электромагнитная. Неопределенность измерений в области электромагнитной совместимости». — М.: Стандартинформ, 2007. — 16 с.

8. Алексеев О.В., Головков А.А., Пивоваров И.Ю. и др. Автоматизация проектирования радиоэлектронных средств. Учебное пособие для вузов. Под ред. Алексеева О.В. — М,: Высшая школа, 2000. — 400 с.

9. Гмурман В.Е. Теория вероятностей и математическая статистика. — М.: Высшая школа, 2003. — 479 с.

Структура виртуального проекта в системе автоматизации проектирования радиоэлектронных средств с функцией виртуальной сертификации по эмиссии излучаемых радиопомех

Н.В. Лемешко, к.т.н.
Кафедра «Радиоэлектроника телекоммуникации», МИЭМ НИУ ВШЭ
nlemeshko@hse.ru, м.т. 8-915-231-17-47

Введение

Одним из перспективных направлений теории автоматизации проектирования является виртуальная сертификация (ВС) по эмиссии излучаемых радиопомех. Понятие виртуальной сертификации было впервые введено в работах [1, 2]. Ее суть заключается в том, что обязательные сертификационные испытания радиоэлектронных средств (РЭС) переводятся в виртуальную плоскость. Это позволяет оценивать уровень помехоэмиссии непосредственно на этапе проектирования, что дает возможность снизить вероятность выявления недопустимой помехоэмиссии при лабораторных испытаниях.

Оформление методологии виртуальной сертификации как целостности дает теоретическую базу для построения соответствующей системы автоматизации проектирования с функцией виртуальной сертификации (САПР ВС). В настоящее время теория ВС вполне развита и наиболее полно изложена в монографии [3]. Необходимость построения САПР ВС обусловлена тем, что моделирование сертификационных испытаний предполагает выполнение очень большого объема расчетов. В частности, моделированию подлежат электрические процессы в РЭС для определения временных функций токов, возбуждающих излучение на частоте анализа, электродинамические процессы излучения электромагнитных волн, их перенос к точке наблюдения с учетом взаимодействия с элементами конструкции РЭС и формирование показаний измерительного приемника с нормированными характеристиками. Несмотря на разнотипность этих процессов, их объединяет информационное единство, поэтому моделирование следует выполнять в среде, характеризуемой внутренней однородностью и согласованностью.

Основной целью выполнения исследований любого объекта на основе математической модели является получение новой информации о его существенных свойствах. В случае моделирования сертификационных испытаний таким объектом является проект РЭС, заменяющий физический образец. Комплексирование информации о нем в совокупности с другими необходимыми данными является проблемой, от решения которой зависит идеология построения САПР ВС в целом. В первую очередь, оно должно быть логичным и оптимизированным.

Структура виртуального проекта РЭС в САПР ВС

Под виртуальным проектом в САПР ВС следует понимать совокупность технической информации о РЭС, взаимосвязанной с виртуальным макетом изделия [4] и достаточной для построения его электрической и электродинамической модели на основе методов, изложенных в [3]. Под виртуальным макетом РЭС, в свою очередь, понимаются структурированные некоторым образом модели разнородных процессов, подлежащих моделированию в САПР ВС, а также его результаты для текущей версии проекта.

Важно отметить, что связь между технической информацией и виртуальным макетом радиоэлектронного средства является неотъемлемым их свойством, именно поэтому выше речь шла о виртуальном проекте в целом. Такая взаимосвязь позволяет при любой корректировке технической информации поддерживать виртуальный проект в актуальном состоянии.

Техническая информация, являясь основой для построения виртуального макета РЭС путем алгоритмических операций, не является первоисточником, характеризующим конструкцию и схемотехнику РЭС. С позиции разработчика конечным продуктом проектирования является не формальное представление об объекте, используемое в САПР, а конструкторская документация, необходимая для его производства. Поэтому техническая информация о РЭС в САПР ВС рассматривается как промежуточное звено, предназначенное для формирования виртуального проекта и всецело зависящее от электронной технической документации, с которой, собственно, и работает инженер.

Техническая информация, используемая для построения моделей в составе виртуального макета, включает в себя дополнительно сведения о стандартных материалах и конструкциях, используемых в РЭС, а также об их физических свойствах. Эти данные содержатся в специальной базе, непосредственно связанной с электронной технической документацией. Ее использование обусловлено тем, что информации о схемотехнике и конструкции РЭС, как правило, не достаточно для построения электродинамических моделей; эти недостающие сведения, в первую очередь касающиеся электрофизических свойств материалов, и содержатся в базе данных, причем выбор их должен осуществляться в интерактивном режиме с участием пользователя.

Основной особенностью виртуального макета следует считать то, что с ним можно проводить те же основные действия, что и с физическим образцом РЭС [5]. Для виртуальной сертификации наиболее важными операциями являются схемное и электродинамическое моделирование РЭС, расчет показаний измерительного приемника по модели. Виртуальный макет в составе с информацией о его текущих свойствах упрощает принятие решений о внесении изменений в конструкцию и схемотехнику проектируемого РЭС.

Важно отметить, что в САПР ВС в качестве исходной для построения виртуального макета используется информация о схемотехнике, тепловых режимах (в общем случае), конструкции РЭС, свойствах используемых материалов и т.д. Рассматривая САПР ВС как подсистему более сложного программного комплекса, можно утверждать, что сформированный в соответствии с приведенным описанием виртуальный проект может быть использован и в других подсистемах. Альтернативным подходом является использование виртуального проекта единой структуры во всех подсистемах такого программного комплекса. Виртуальный проект в САПР ВС заменяет опытный образец, а математическое моделирование, выполняемое в соответствии с заложенной теорией, — лабораторные исследования помехоэмиссии.

Рассмотрим структуру и взаимосвязи виртуального проекта, предлагаемого для использования в САПР ВС. Она изображена на рис. 1. Виртуальный проект формируется, как отмечалось выше, на основе технической информации о РЭС, представляемой в форме, необходимой для функционирования системы. В состав виртуального макета включена модель тепловых процессов в РЭС, которая обеспечивает уточненный расчет токов в проводниках. Отмеченное на рисунке

взаимодействие электрической и тепловой моделей обусловлено тем, что выделяемая элементами РЭС тепловая энергия влияет на режимы их работы, что приводит к изменению токов в проводниках [4, 6].

Наличие в составе виртуального макета как тепловой, так и электрической модели РЭС позволяет сделать САПР ВС функционально независимой от других подсистем в составе системы более высокого уровня. Однако если в последнюю входят САПР, имеющие функцию комплексного моделирования, то использование аналогичных моделей в составе САПР ВС приведет к избыточности. Устранить её можно путем интеграции САПР ВС с системой, реализующей комплексное моделирование. Таким образом, рассматриваемая структура виртуального проекта имеет определенный потенциал к упрощению. Вместе с тем, при разработке структуры САПР ВС следует считать её автономным средством автоматизации.

Модель электрических процессов в составе виртуального макета должна строиться не только с учетом схемотехники РЭС, но и вносимых его конструкцией эффектов. Кроме того, важной чертой САПР ВС является ориентация на разные классы оборудования, обеспечивающая универсальность и широту применения системы. При схемном моделировании должны обязательно учитываться свойства интегральных компонентов, что является одним из самых сложных вопросов в теории и практике автоматизированного проектирования. Значительную роль в этом смысле могут сыграть IBIS-модели [7], предназначенные для использования при построении схем замещения цифровых устройств.

Рис. 1. Структура и взаимосвязи виртуального проекта в САПР ВС

Номенклатура микросхем и других активных элементов, используемых в устройствах каждого из классов, обновляется сравнительно редко. Для пассивных компонентов это обновление происходит еще реже. Поэтому информацию о моделях компонентов представляется целесообразным объединить в отдельный банк, данные из которого будут использоваться на основе содержания технической информации о проекте. Поскольку виртуальной сертификации планируется в основном подвергать новые РЭС широкого потребления, то банк моделей, несмотря на отмеченную статичность, необходимо периодически обновлять, что может выполняться в автоматическом режиме при наличии подключения к сети Интернет. Кроме того, САПР ВС должна предусматривать возможность подключения стандартизованных библиотек производителей электронных компонентов. В сравнении с элементной базой РЭС номенклатура конструкций и материалов изменяется еще медленнее.

Установлено, что в САПР ВС рационально использовать комплексные компонентные модели, включающие информацию о схеме замещения, конструкции корпуса и его тепловых свойствах. В текущем варианте схемы на рис. 1 предусмотрен банк только электрических моделей, поскольку комплексные компонентные модели используются только в некоторых САПР, например, [8]. Информация о топологии корпусов компонентов, необходимая для построения электрической и электродинамических моделей, передается из базы данных стандартных материалов, конструкций и их физических свойств, вливаясь в общий поток технической информации в составе виртуального проекта.

Условия проведения сертификационных испытаний оказывают непосредственное влияние на их результат. Они включают в себя требуемую конфигурацию измерительного оборудования, перечень частот анализа и другие данные, выбираемые пользователем. К информации об условиях проведения измерений относят измерительное расстояние, поэтому её используют и для построения электродинамических моделей. Выбирая условия выполнения сертификационных испытаний, пользователь должен руководствоваться принятыми стандартами.

Входящие в состав виртуального макета модель электродинамических процессов и электрическая модель средств измерений строятся на заложенной в [3] методологической базе. Взаимодействие моделей электрических, электродинамических процессов в РЭС, а также модели средств измерений, заключается в том, что результаты моделирования для предыдущей модели в этом перечислении служат исходными данными для последующей. САПР ВС предполагает использование упрощенной электродинамической модели РЭС, служащей для поиска точки наблюдения и, при необходимости, приближенных расчетов, и уточненной, на основе которой в выбранной точке наблюдения рассчитывается временная функция напряженности поля.

При разработке САПР ВС особое значение имеет методическое обеспечение. Согласно схеме на рис. 1, методики моделирования наряду с технической документацией являются отправной точкой формирования виртуального проекта. На их основе разрабатываются алгоритмы построения моделей, входящих в состав виртуального проекта, а также вычислительное ядро САПР ВС. Именно поэтому методологический аспект играет очень важную роль. Алгоритмы формирования моделей являются формализацией разработанных методик, позволяющей строить модели для проектов конкретных РЭС в автоматическом режиме.

Функцией вычислительного ядра САПР ВС является расчетное сопровождение моделирования, реализуемое на основе моделей, входящих в состав виртуального макета. В качестве основных результатов моделирования следует рассматривать оценки помехоэмиссии, выполненные для заданных условий в некоторой полосе частот, однако в САПР ВС следует предусмотреть возможность вывода и другой информации, получаемой в результате моделирования. Это позволит упростить анализ результатов моделирования сертификационных испытаний и сделать более обоснованным принятие решений о доработке проекта РЭС в том или ином направлении.

Литература

1. Кечиев Л.Н., Лемешко Н.В. Виртуальная сертификация радиоэлектронных средств по уровню помехоэмиссии как средство подготовки к лабораторным испытаниям по электромагнитной совместимости. — Труды НИИР, сборник научных статей / Под ред. Бутенко В.В. — М.: НИИР, 2010, №1. — с.57-70.
2. Кечиев Л.Н., Лемешко Н.В. Виртуальная сертификация радиоэлектронных средств по уровню помехоэмиссии. Постановка проблемы. — Технологии ЭМС, №2 (33) — М.: ООО Издательский дом «Технология», 2010. — с.3-15.
3. Лемешко Н.В. Теоретические основы моделирования сертификационных испытаний радиоэлектронных средств по эмиссии излучаемых радиопомех. Монография. — М.: МИЭМ, 2012. — 196 с.
4. Автоматизированная система АСОНИКА для проектирования высоконадежных радиоэлектронных средств на принципах CALS-технологий. Т.1. — Под ред. Кофанова Ю.Н., Малютина Н.В., Шалумова А.С. — М.: Энергоатомиздат, 2007 г. — 368 с.
5. Кофанов Ю.Н., Новиков Е.С., Шалумов А.С. Информационная технология моделирования механических процессов в конструкциях радиоэлектронных средств. — М.: Радио и связь, 2000. — 160 с.
6. Жаднов В.В., Сарафанов А.В. Управление качеством при проектировании теплонагруженных радиоэлектронных средств. — М.: СОЛОН-Пресс, 2004. — 464 с.
7. Кечиев Л.Н., Лемешко Н.В. Моделирование цифровых устройств с использованием IBIS-описания интегральных схем. — М.: МИЭМ, 2006. — 243 с.
8. Сабунин А.Е. Altium Designer: новое поколение в проектировании электронных устройств. — М.: Солон-Пресс, 2009. — 432 с.

Разработка структуры программных средств системы автоматизации проектирования с функцией виртуальной сертификации радиоэлектронных устройств по эмиссии излучаемых радиопомех

Н.В. Лемешко, к.т.н., С.С. Захарова, к.т.н.
Кафедра «Радиоэлектроника телекоммуникации», МИЭМ НИУ ВШЭ
nlemeshko@hse.ru, szaharova@hse.ru

Одним из основных вопросов в теории виртуальной сертификации, сущность которой раскрыта в других публикациях в настоящем сборнике, состоит в проработке структуры программных средств реализующей ее системы автоматизации. Поскольку моделирование сертификационных испытаний радиоэлектронных средств по эмиссии излучаемых радиопомех является задачей, требующей большого объема вычислений, то оно должно проводиться в единой системе автоматизации, характеризующейся универсальностью по отношению к исследуемым объектам в пределах ограничений, установленных в [1].

Основные принципы построения системы автоматизации проектирования с функцией виртуальной сертификации (САПР ВС), являющиеся отправной точкой в ее разработке, состоят в следующем [2].

1. Требования к САПР, как правило, определяются её функциями в составе более сложной системы, включающей в себя САПР как подсистему. Если говорить о РЭС, то такой системой может быть САПР сквозного проектирования, используемая на предприятии. Для САПР ВС должен действовать принцип включения, обеспечивающий такое вхождение.

2. САПР ВС должна отвечать принципу системного единства, что обеспечивается наличием информационных и управляющих связей между ее программными модулями.

3. САПР ВС должна соответствовать принципу комплексности, который требует связности каждого элемента и всего РЭС в целом на этапах создания, редактирования, моделирования и анализа его результатов при проведении виртуальной сертификации.

4. Для САПР ВС должен выполняться принцип информационного единства, согласно которому отдельные подсистемы и компоненты должны быть информационно согласованы. Это означает, что в модулях САПР должны использоваться единые термины, символы, условные обозначения и т.п. Этот принцип должен проявляться и в части хранения наиболее часто используемой информации, необходимой для функционирования САПР, которую рекомендуется структурировать.

5. Принцип совместимости также должен быть реализован в САПР ВС. Он состоит в том, что языки, коды, информационные и технические характеристики структурных связей подсистем и компонентов системы должны быть согласованы так, чтобы обеспечивалось их совместное функционирование и сохранялась открытая структура системы в целом. Это позволит создавать дополнительные надстройки для САПР ВС, как технические, так и программные, без изменения модулей, уже введенных в эксплуатацию. Этот, казалось бы, второстепенный принцип позволяет без полной модернизации вводить в эксплуатацию модули, расширяющие функциональность САПР, подключать дополнительные базы данных и т.д.

6. САПР ВС должна отвечать принципу инвариантности, что означает универсальность применения компонентов и подсистем к объектам разработки. В [1] были выделены принципиальные ограничения, которым должно соответствовать подвергаемое виртуальной сертификации РЭС. Эти ограничения естественным образом вытекают из сущности методов, составляющих ее теоретический базис, поэтому инвариантность САПР ВС может быть обеспечена только в пределах существующих методических ограничений.

7. Рассмотренный выше принцип совместимости должен быть дополнен реализацией в САПР ВС принципа развития, который предусматривает возможность наращивания и совершенствования компонентов и подсистем, а также связей между ними. Для САПР ВС использование принципа развития не только повышает универсальность системы, но и дает возможность создавать специализированные комплексы, предназначенные для моделирования сертификационных испытаний аппаратуры отдельных типов, проводимых в т.ч. в нестандартных условиях и с использованием специальных измерительных средств.

Изложенные требования к САПР ВС и принципы её построения реализуются выработкой соответствующей структуры разрабатываемой системы.

Как следует из публикаций [3 — 5], структурная схема для одной и той же САПР может разрабатываться на разных уровнях детализации и в различных формах, что в первую очередь определяется текущей проработкой системы. Например, в [4] для программного комплекса ТРиАНА приводится подробная структурная схема программных средств с обозначением отдельных модулей в составе подсистем и подробным описанием их взаимодействия. С другой стороны, в [3] приводятся многочисленные схемы САПР электротехнических устройств без столь подробного описания.

В настоящее время САПР ВС находится на уровне концептуально-эскизной проработки при наличии вполне сформированного методического обеспечения. Естественно, что приведенным выше требованиям могут соответствовать САПР, построенные по разным схемам, поэтому для САПР ВС ниже рассматривается лишь один из вариантов структуры программных средств системы, который по мере развития будет дополнен требуемыми подробностями. Структура программных средств САПР ВС изображена на рис. 1.

Взаимодействие пользователя с САПР ВС осуществляется через модуль универсальных пользовательских интерфейсов, которые определяются типом используемого системного программного обеспечения и характеристиками аппаратной платформы. Этот модуль выполняет визуализацию информации, выводимой САПР ВС, а также ввод текстовой и графической информации при просмотре и редактировании проекта РЭС и баз данных.

САПР ВС характеризуется большим количеством информационных потоков. Согласованность функционирования программных модулей обеспечивается управляющей программой САПР ВС, взаимодействующей с каждым из них. Таким образом, основной функцией управляющей программы САПР ВС являются коммутация информационных потоков, формирование управляющей информации для программных модулей и получение контрольных данных, а также информирование пользователя о текущем состоянии и действиях САПР ВС.

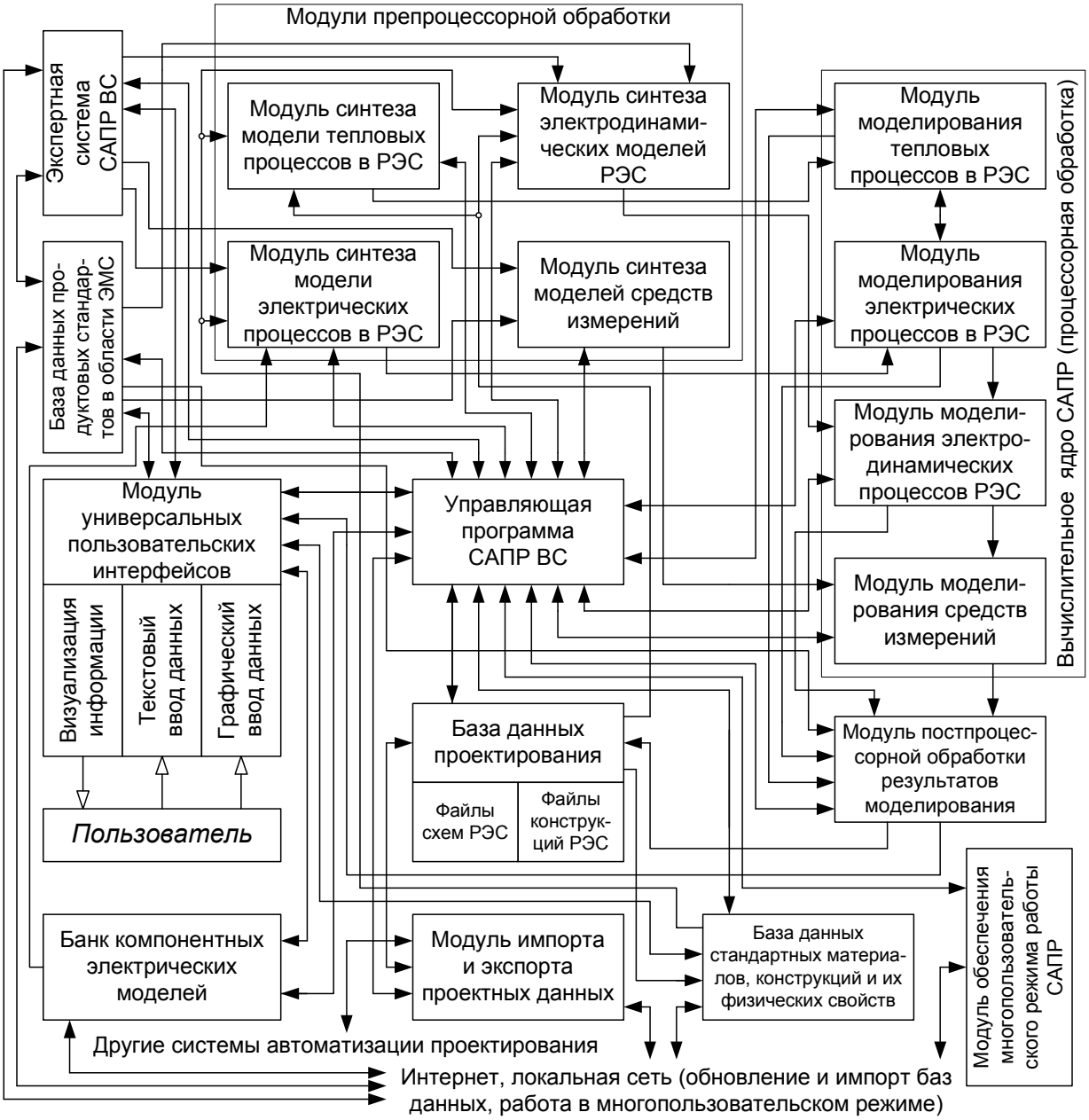


Рис. 1. Структурная схема программных средств САПР ВС

Модели электрических, тепловых и электродинамических процессов строятся при помощи модулей синтеза, осуществляющих препроцессорную обработку информации и приводящих разнородные данные о проекте к пригодному для использования виду. При построении моделей соответствующих типов использует не только информация базы данных проектирования, но и дополнительны сведения об используемых стандартных материалах, конструкциях и их физических свойствах. Построение электрических моделей выполняется с использованием банка компонентных электрических моделей.

Практика использования средств автоматизации показала, что процесс построения моделей разнородных процессов в сложных объектах, какими бы совершенными алгоритмами он не реализовывался, не может проходить в полностью автоматизированном режиме [6, 7]. Во всех случаях, когда в САПР возможно появление неоднозначных решений, должен использоваться интерактивный режим.

При построении электродинамических моделей РЭС и моделей средств измерений используется информация из базы данных продуктовых стандартов по

ЭМС, определяющих условия проведения сертификационных испытаний. Эти стандарты являлись основой для разработки методологии моделирования сертификационных испытаний. Важно отметить, что содержание этой базы данных является рекомендательным, т.е. при необходимости пользователь может сам задавать основные параметры построения моделей и выполнения моделирования.

Использование в составе САПР ВС экспертной системы может значительно упростить отбор проводников для выполнения анализа помехоэмиссии, а также выбор моделируемых средств измерений. В зависимости от того, какие базы знаний включены в состав экспертной системы, она может выполнять многие дополнительные функции, поэтому на текущем уровне проработки перечень дополнительных баз знаний в целом следует оставить открытым, отнеся к нему проблемы выбора тепловых режимов для радиоэлементов при отсутствии информации, достаточной для теплового моделирования РЭС, а также оптимизации настроек модулей препроцессорной обработки и вычислительного ядра САПР ВС для конкретных типов сертифицируемых РЭС.

Сформированные в ходе препроцессорной обработки модели используются для моделирования свойств РЭС в вычислительном ядре САПР, которое имеет соответствующие программные модули. Взаимосвязь последних, отмеченная на рис. 1, заключается в том, что результаты, полученные в предыдущем модуле, служат в качестве исходных данных для последующего. Это положение может быть нарушено только для взаимодействия тепловой и электрической моделей РЭС, что отмечено двунаправленной стрелкой. При совместной организации моделирования электрических и тепловых процессов [8] обычно возникает итерационный цикл, обусловленный расчетом мощности тепловыделения в радиоэлементах и их температур с последующим попеременным уточнением их значений до достижения сходимости.

После выполнения моделирования полученная информация поступает в программный модуль постпроцессорной обработки. Как отмечалось выше, основным результатом моделирования сертификационных испытаний являются оценки показаний средств измерений, полученные в точках наблюдения, соответствующих их максимуму. Поэтому предполагается сохранять и выводить второстепенную информацию только по запросу пользователя при проведении углубленного анализа. В модуль постпроцессорной обработки результатов моделирования поступают данные из базы данных продуктовых стандартов, необходимые для оценки соответствия РЭС требованиям стандартов и для расчета неопределенности результатов моделирования.

Постпроцессорная обработка включает в себя представление данных, необходимое для информативного отображения полученной путем моделирования информации об объекте разработки. Эти данные в последствии визуализируются. Вместе с тем, они сохраняются в базе данных проектирования, которая должна, как это сейчас принято во многих САПР, содержать архив проектов. Как следует из предложенной структуры САПР ВС, решение о внесении изменений в проект РЭС принимается пользователем, однако и здесь возможно использование экспертной системы, особенно при выявлении путей снижения помехоэмиссии.

Многопользовательский режим работы САПР ВС предполагает использование соответствующего модуля при работе системы в локальной либо всемирной компьютерной сети. Взаимодействие с остальными модулями САПР ВС осуществляется при этом через управляющую программу, которая в данном случае не

только коммутирует потоки данных, но и обеспечивает ограничение доступа к базам данных системы и их изменение. Все базы данных САПР ВС, кроме базы данных проектирования, а также экспертная система должны иметь возможность обновляться через Интернет либо через локальную сеть.

В настоящее время САПР обычно позволяют импортировать данные из других систем проектирования, что делает их более универсальными. В САПР ВС это обеспечивается функциональностью модуля импорта и экспорта проектных данных. Важно отметить, что конвертирование форматов имеет и производственное значение, поскольку современное оборудование позволяет изготавливать конструкционные элементы РЭС на основе файлов специальных форматов.

При использовании САПР ВС ввиду сложности решаемой ей задачи и наличия ряда особенностей функционирования может возникнуть потребность в ознакомлении разработчиков РЭС с содержанием баз данных и с правилами баз знаний экспертной системы. Очевидно, что это не будет связано с использованием препроцессорной, постпроцессорной обработки либо вычислительного ядра. Поэтому в структурной схеме имеются связи, допускающие просмотр информации из указанных модулей, однако её изменение и редакция возможны только при наличии соответствующих прав.

Заканчивая описание предложенной структурной схемы, отметим, что непосредственная редакция проекта осуществляется в модуле универсальных пользовательских интерфейсов, а результаты сохраняются в базе данных проектирования.

Литература

1. Лемешко Н.В. Теоретические основы моделирования сертификационных испытаний радиоэлектронных средств по эмиссии излучаемых радиопомех. Монография. — М.: МИЭМ, 2012. — 196 с.
2. Алексеев О.В., Головков А.А., Пивоваров И.Ю. и др. Автоматизация проектирования радиоэлектронных средств. Учебное пособие для вузов. Под ред. Алексеева О.В. — М,: Высшая школа, 2000. — 400 с.
3. Кулон Ж.-Л., Сабоннадьер Ж.-К. САПР в электронике. — Пер. с франц. под ред. Стрельбицкого Э.К. — М.: Мир, 1988. — 208 с.
4. Жаднов В.В., Сарафанов А.В. Управление качеством при проектировании теплонагруженных радиоэлектронных средств. — М.: СОЛОН-Пресс, 2004. — 464 с.
5. Автоматизированная система АСОНИКА для проектирования высоконадежных радиоэлектронных средств на принципах CALS-технологий. Т.1. — Под ред. Кофанова Ю.Н., Малютина Н.В., Шалумова А.С. — М.: Энергоатомиздат, 2007 г. — 368 с.
6. Разевиг В.Д. Система схемотехнического моделирования Micro-CAP V. — М.: Солон, 1997. — 273 с.
7. Разевиг В.Д. Система схемотехнического моделирования и проектирования печатных плат Design Center (PSpice). — М.: СК Пресс, 1996. — 272 с.
8. Сарафанов А.В. Структурная организация подсистемы моделирования тепловых характеристик РЭС. — Вестник Красноярского государственного технического университета. Сборник научных трудов. Выпуск 4. — Красноярск, КГТУ, 1996. — с.37 – 42.

«Разработка программного обеспечения»

Система человеко-машинного взаимодействия на базе акселерометра

Аксёнов К.В
МИЭМ НИУ ВШЭ, каф. ИТАС

Введение

Безусловно, каждый человек, работавший с компьютером, осознает важность наличия компьютерной мыши. Это устройство проверено годами, тем не менее, все более развивающийся функционал компьютерного обеспечения требует изменения самих манипуляторов. К примеру, как альтернативный вид манипулятора давно зарекомендовал себя графический планшет, позволяющий художникам разговаривать с компьютером на более понятном им языке движения пера, а так же появились и имеют большой успех сенсорные экраны, для которых манипулятор – наши руки. Что может быть естественнее этого?

В связи с этим разделением манипуляторов на отдельные группы появляется так же уйма проблем, связанных с необходимостью привыкать к новому виду манипулятора, если перед человеком внезапно встает такая необходимость. Художник, замечательно управляющийся с обычным пером, ни за что не нарисует красивой картины пером планшета. Он умеет обращаться с пером, но на бумаге, он умеет обращаться с компьютером, но с помощью компьютерной мыши. И проблема перехода, переучивания, останавливает многих на пути освоения всех возможностей современной компьютерной техники.

Это лишь один пример, заставляющий задуматься над решением проблемы усовершенствования компьютерных манипуляторов. Ведь объем информации, получаемой нами через компьютер и объем работ, проводимых с помощью него, растет непомерными темпами, и в дополнение к вопросу универсальности увеличивается значение эргономичности всех окружающих нас устройств. Активно продвигаемые сейчас сенсорные экраны этому требованию не вполне отвечают. Они прекрасно подходят для короткого взаимодействия, когда человеку нужно срочно найти какую-либо информацию в сети, к примеру, адрес, название чего-либо, или включить фильм для просмотра. Но если представить себе домашний или рабочий компьютер, взаимодействовать через который можно только с помощью сенсорного экрана? Во-первых, монитор такого компьютера не может быть очень большим, чтобы вся его область была доступна. Во-вторых, человек не сможет долго с ним работать – ведь для этого руку придется практически все время держать на весу, или вовсе поднятой! Даже если рассматривать экраны, расположенные горизонтально или почти горизонтально.

Так что за бумом сенсорных экранов абсолютно точно не последует исчезновение манипуляторов. Вопрос – в каком виде они останутся? В данной статье представленна беспроводная система взаимодействия с компьютером на базе акселерометров, которая не будет нагружать руку человека излишним весом, облегчая работу с ней, и может работать на любой поверхности.

Структура системы

Цель этого проекта – спроектировать более эргономичный компьютерный манипулятор и добиться его подстраивания под разные сферы работ. Манипулятор должен по максимуму использовать функционал человеческой руки, и потому на ней не должно быть ничего лишнего.

Проектируемый манипулятор будет состоять из следующих компонентов:

- Две насадки-клипсы на кончики указательного и среднего пальцев, содержащие в себе датчики ускорения;
- Браслет, содержащий в себе блок питания;
- Чип, обрабатывающий данные с датчиков;
- Bluetooth передатчик, обеспечивающий связь с компьютером;
- Провода, соединяющие датчики на кончиках пальцев с браслетом, крепящимся на запястье.

Так же компьютер, с которым предполагается работать, должен иметь встроенное или подключаемое Bluetooth устройство.

Ускорение, фиксируемое датчиками, будет соответствовать движению мыши по столу, позволяя нам определить вектор перемещения, который легко можно конвертировать в движение указателя на мониторе. Постукивание указательным или средним пальцем по столу равнозначно клику левой или правой кнопки мыши.

Подобная конфигурация позволяет достичь гибкости не только в смене рабочей области, но и замечательно подходит для использования устройства как правшой, так и левшой (которому для работы левой рукой необходимо только поменять клипсы на пальцах местами).

Анализ предметной области

Несмотря на всю точность новейших приборов и датчиков, любой из них на выходе дает вовсе не чистый сигнал, а шум в некотором диапазоне, так как любой измерительный прибор обладает некоторой погрешностью, и на него может оказывать влияние большое количество внешних и внутренних воздействий. Это и приводит к тому, что информация с него оказывается зашумленной. Чем сильнее зашумлены данные, тем сложнее обрабатывать такую информацию. Пример сигнала датчика представлен на рис. 1.

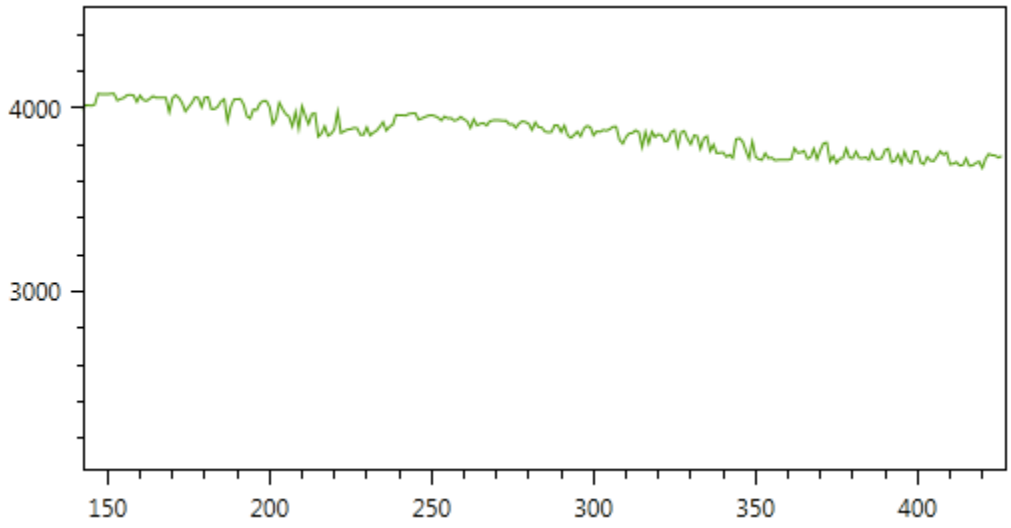


Рис.1. Пример сигнала, получаемого с датчика

Для анализа этого шума существует большое количество алгоритмов, но мною выбран фильтр Калмана.

Фильтр Калмана — это, наверное, самый популярный алгоритм фильтрации, используемый во многих областях науки и техники. Благодаря своей простоте и эффективности, его можно встретить в GPS-приемниках, обработчиках показаний датчиков, при реализации систем управления и т.д.

Слово «фильтр» подразумевает под собой алгоритм обработки данных, который убирает шумы и лишнюю информацию. В фильтре Калмана есть возможность задать априорную информацию о характере системе и связи переменных, и на основании этого строить более точную оценку. Но даже в простейшем случае он дает отличные результаты.

Результат применения фильтра Калмана представлен на рис.2.

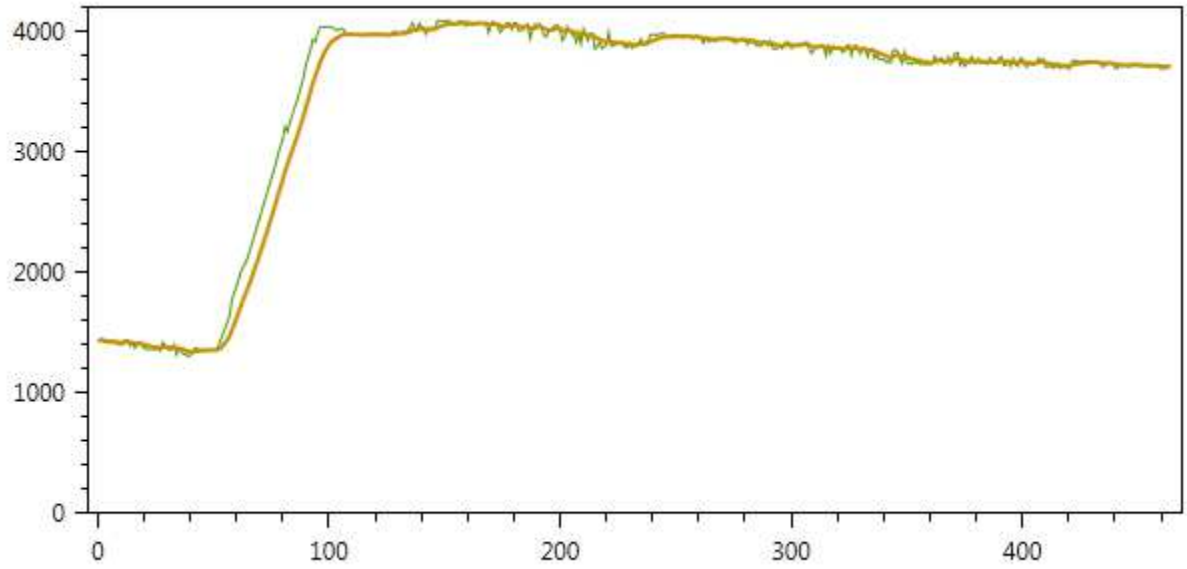


Рис.2. Фильтрация сигнала с использованием фильтра Калмана

Фильтр Калмана использует динамическую модель системы (например, физический закон движения), известные управляющие воздействия и множество последовательных измерений для формирования оптимальной оценки состояния. Алгоритм состоит из двух повторяющихся фаз: предсказание и корректировка (рис.3.).

На первом рассчитывается предсказание состояния в следующий момент времени (с учетом неточности их измерения). На втором новая информация с датчика корректирует предсказанное значение (также с учетом неточности и зашумленности этой информации).

Как видно из представленной UML-схемы (рис.4.), в рассматриваемой системе четыре основных элемента:

- Пользователь;
- Акселерометр;
- Платформа Arduino;
- Компьютер.

- Связь между пользователем и акселерометром осуществляется путем закрепления датчиков на кончиках пальцев, движение которых отслеживается. Данные с датчиков по проводам передаются обрабатывающей платформе Arduino, которая с помощью передатчика Bluetooth передает обработанные

данные на соединенный с ней компьютер. Далее данные поступают в разработанную программу для обработки.

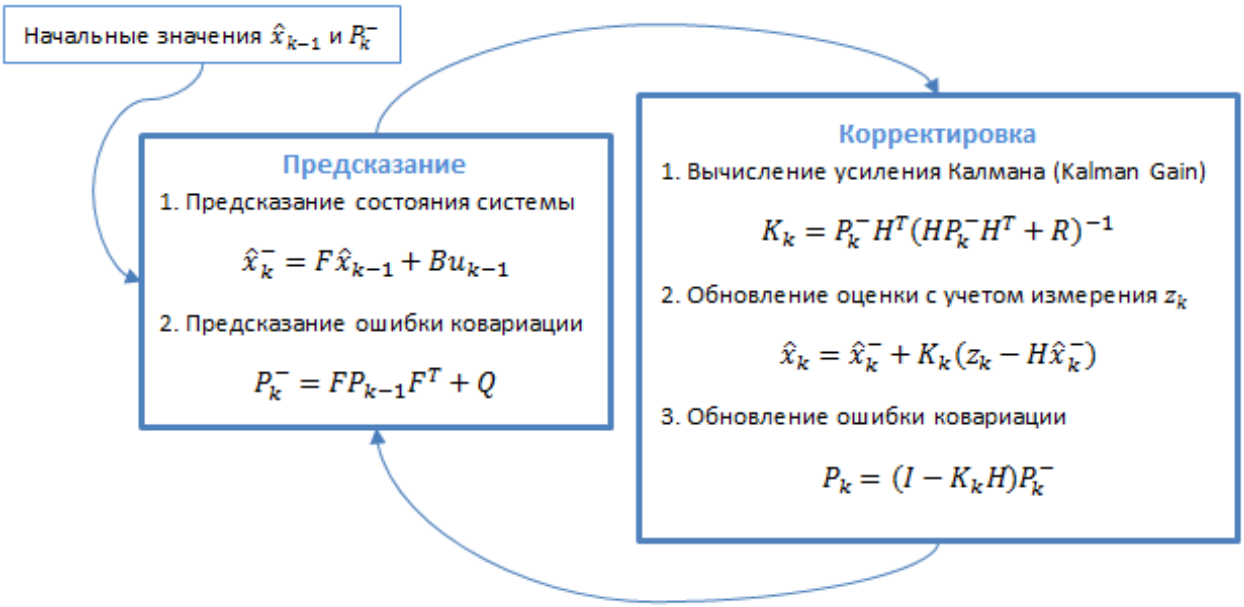


Рис.3. Алгоритм оценки сигнала

Структура системы

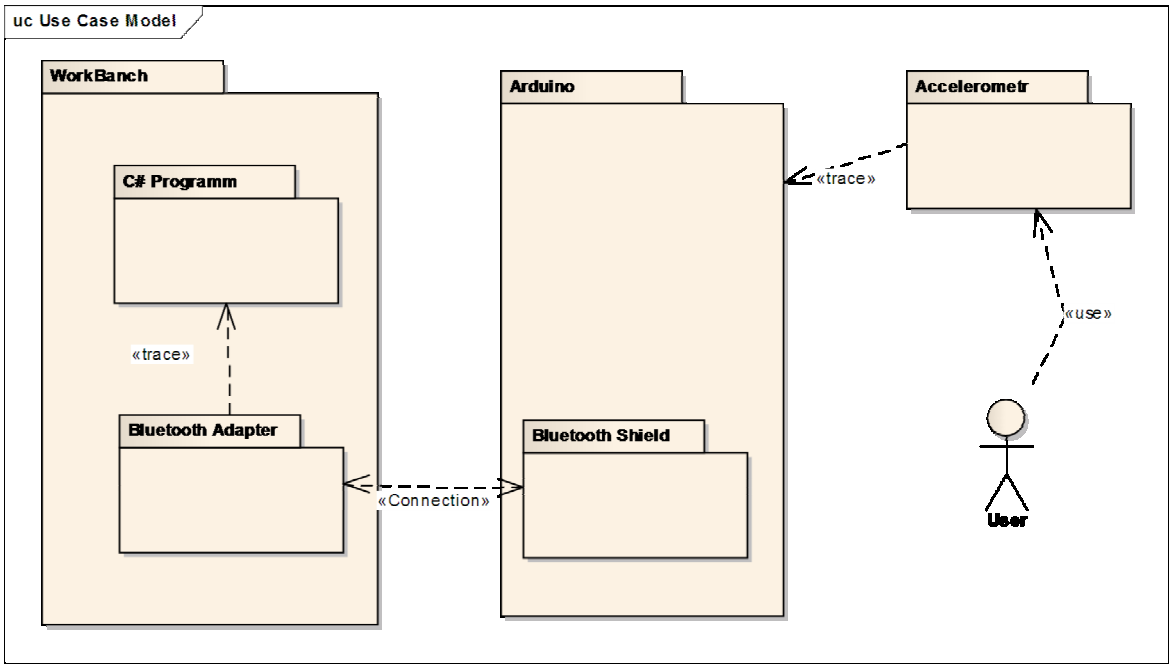


Рис.4. UML-схема устройства

Заключение

В данной статье была описана система человеко машинного взаимодействия, лишенная многих недостатков сегодняшних аналогов – узкопрофильности, низкой эргономичности, неудобства в использования на многих поверхностях. В ходе разработки были решены проблемы шума, которого невозможно избежать в виду вибрации рук человека, а также учтена возможность использования разработанной системы различными типами пользователей, по максимуму учтены их потребности и условия работы.

Особенности построения графического интерфейса для ИПС Серелекс на платформе Windows 8

А. Романов, П. Романов
МГТУ имени Н.Э. Баумана

В современном мире все чаще начинают использоваться новые, инновационные, способы работы с информацией. На смену клавиатуре и мыши приходит сенсорный экран и взаимодействие с объектами на нем с помощью жестов и прикосновения пальца (или стилуса), и данную особенность нельзя не учитывать при проектировании новых, современных интерфейсов взаимодействия человека-оператора с информационной системой.

Рассмотрим пример такого взаимодействия на примере приложения «Серелекс», предназначенного для визуализации семантических отношений между некоторыми словами (подробнее -- http://serelex.it-claim.ru/page/about)

После запуска приложения пользователь видит окно для ввода запроса, причем это окно имеет такой размер, что бы человек мог легко попасть в него пальцем, и кнопку «Поиск», также соответствующего размера. После нажатия на неё происходит поиск слов, семантически связанных с данным (синонимов, гиперонимов, ко-гипонимов и ассоциаций). Результаты выводятся на экран в формате, указанном на Рис 1.

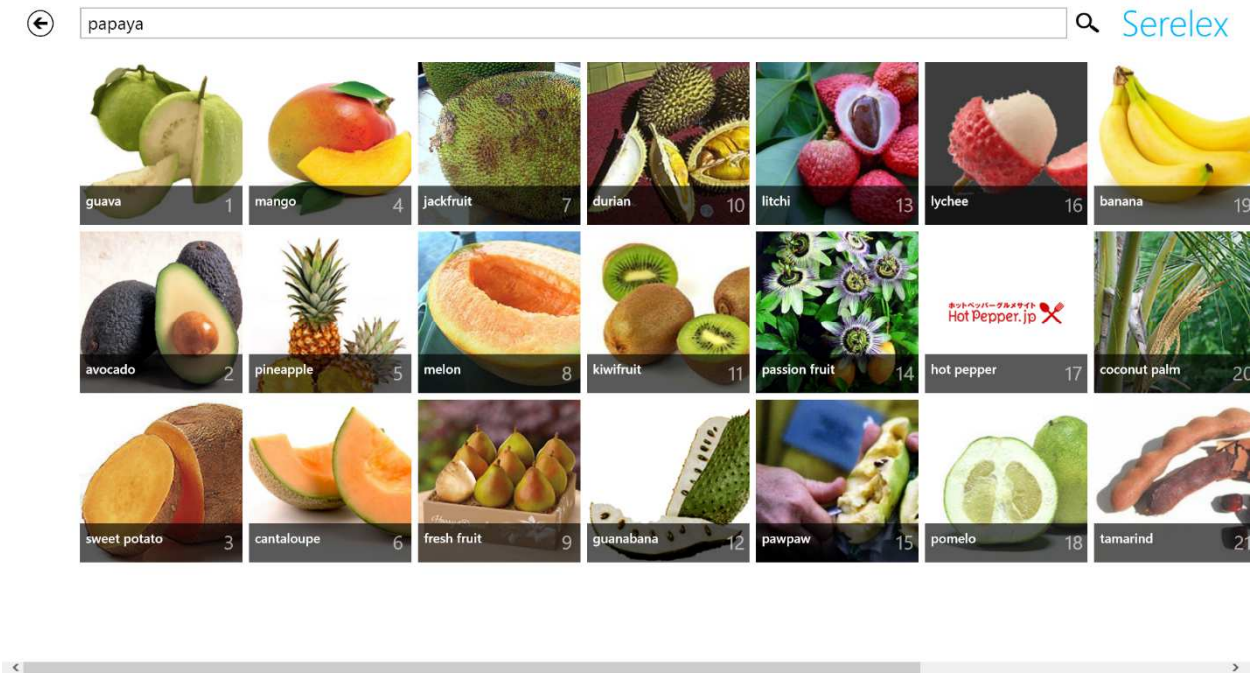


Рис 1. Результат поиска по запросу «papaya»

Как видно из Рис. 1, результаты поиска выводятся в специальном виде, а не просто простым списком, как принято в классических системах. Каждое слово из результата символизируется изображением 250x250 пикселей, что позволяет легко попасть пальцем или стилусом именно по этому результату для осуществления поиска по данному слову. Кроме того, в нижней части отображается само слово, а также его порядковое место в результатах поиска.

Если же пользователю недостаточно показанных на экране результатов, то он может осуществить жест «прикосновение и ведение пальца справа налево» для того, что бы прокрутить отображаемые на экране элементы и поставить на их место новые, пока невидимые результаты.

Кроме того, следует обратить внимание на сам стиль данного приложения. Оно выполнено в стиле современных приложений для Windows 8, который характеризуется правильным использованием шрифтов, приоритетом контента и скрытием лишних, отвлекающих деталей, динамичным и плавным пользовательским интерфейсом.

Все это приводит к тому, что данным приложением, запущенным на устройстве с сенсорным экраном, удобно пользоваться, время информационного поиска – уменьшилось, а интенсивность воспринимаемой информации – увеличилась.

Построение такого приложения происходит не с нуля, а с заранее созданного программистами Microsoft шаблона, так как в официальной поставке Visual Studio 2012 есть шаблон проекта с похожим внешним видом и функциональностью. Программисту остается только реализовать структуру классов и источников данных, написать шаблон (DataTemplate) для отображения элементов коллекции, в данном случае – результатов поиска. Например, он может быть таким:

```
    <DataTemplate x:Key="SearchResultGridItemTemplate">
      <Grid HorizontalAlignment="Left" Width="250" Height="250">

<Border Background="{StaticResource ListViewItemPlaceholderBackgroundThem
eBrush}">

<Image Source="{Binding Image}" Stretch="UniformToFill" AutomationPropert
ies.Name="{Binding Word}"/>
        </Border>
        <StackPanel VerticalAlignment="Bottom"

Background="{StaticResource ListViewItemOverlayBackgroundThemeBrush}">

<TextBlock Text="{Binding Word}" Height="60" Margin="15,0,15,0"

Foreground="{StaticResource ListViewItemOverlayForegroundThemeBrush}"
                     Style="{StaticResource TitleTextStyle}"/>

<TextBlock Text="{Binding Value}" TextWrapping="NoWrap" Margin="15,0,15,1
0"

Foreground="{StaticResource ListViewItemOverlaySecondaryForegroundThemeBr
ush}"
                     Style="{StaticResource CaptionTextStyle}"/>
        </StackPanel>
      </Grid>
    </DataTemplate>
```

Источник данных в данном случае – RESTfull-сервис, который возвращает данные в JSON-формате. Получение данных обеспечивает стандартный класс HttpClient, котрый имеет асинхронные методы доступа к HTTP-ресурсам, благодаря чему не происходит «замораживания» пользовательского интерфейса, как было бы в

случае, если использовались бы синхронные методы без создания отдельного потока. Полученные текстовые данные преобразуются в объекты благодаря библиотеке Newtonsoft.JSON (http://json.codeplex.com/) и заносятся в список, который и отображается на экране пользовательского устройства.

Список литературы

1. Принципы дизайна приложений для Windows 8 // MSDN Blog. URL: http://blogs.msdn.com/b/kichinsky/archive/2012/09/23/windows-8-design-principles-introduction.aspx (дата обращения: 02.12.2012)
2. Panchenko A., Romanov P., Morozova O., Naets H., Philippovich A., Romanov A., Fairon C. Serelex: Search and Visualization of Semantically Related Words. In Proceedings of the 35th European Conference on Information Retrieval (ECIR 2013). Lecture Notes in Computer Science (Springler), vol.7814, Moscow, Russia
3. С. Пугачев, А. Шериев, К. Кичинский. Разработка приложений для Windows 8 на языке C#. БХВ-Петербург, 416 стр.

Автоматизированное создание тестового регрессионного набора на основе документации процесса тестирования программного обеспечения

Мартюков А.С.
Московский институт электроники и математики Научно-исследовательского университета «Высшая школа экономики»,
кафедра "Информационные технологии и автоматизированные системы"

Аннотация. В данной статье рассматривается автоматизированная система, позволяющая формировать наборы тестов для регрессионного тестирования на основе результатов автоматизированного тестирования программного обеспечения и входной документации, а так же создавать отчетную документацию по этому автоматизированному процессу тестирования.

Введение

Тестирование программного обеспечения является сейчас сложной задачей, требующей освоения множества дисциплин. Это связано с большим количеством видов тестирования, которые необходимо проводить: функциональное тестирование, нагрузочное тестирование(тестирование производительности, тестирование стабильности), тестирование удобства использования, тестирование интерфейса пользователя, тестирование локализации, тестирование совместимости (в данной классификации не рассматривается тестирование безопасности, так как обычно его выделяют в отдельный класс). Основным видом можно назвать функциональное тестирование. Это обусловлено тем, что функциональность является самой важной характеристикой качества программного обеспечения. Функциональность определяется, как способность программы выполнять все функции, которые обозначенные в ее спецификации и удовлетворяющее заданным потребностям пользователей. Поэтому функциональному тестированию уделяют наибольшее внимание.

При функциональном тестировании могут тестироваться новые элементы программы или элементы, которые уже тестировались ранее, но в них были внесены изменения (исправление ошибок, добавление новых функций и т.д.). Тестирование измененной версии программного обеспечения называется регрессионным тестированием. Регрессионное тестирование — вид тестирования программного обеспечения, который выполняется для проверки корректности работы функционала, который уже тестировался ранее. Это наиболее часто применяемый вид функционального тестирования, потому что:

- выпускаются новые версии продукта и их обновления;
- во многих компаниях применяется интеграционная разработка программного обеспечения и на каждой итерации необходимо проверить не только новые элементы и функции, но и уже протестированные ранее элементы на предмет не внесения в них ошибок.

Формирование регрессионного тестового набора

В регрессионном тестировании используются тесты, созданные при проведении предыдущего тестирования. Из этих тестов формируется новое множество тестов, которое направлено на тестирование только измененных или добавленных элементов

программы. Это формирование происходит на основе результатов предыдущего тестирования и информации об изменениях, не связанных с исправлением ошибок. Вся эта информация хранится в документации процесса тестирования.

Но анализ этих данных и создание документации для следующего тестирования отнимает много времени у тестировщика. Для решения этой проблемы предлагается автоматизировать процесс сбора необходимой информации по процессу тестирования, процесс формирования нового регрессионного тестового набора, а так же процесс создания отчетной документации по процессу тестирования.

Для автоматизации формирования нового множества тестов, которое будет использоваться в регрессионном тестировании, предлагается безопасный метод, основанный на входящей документации процесса тестирования.

Документирование процесса тестирования

При создании документации по процессу тестирования предлагается использовать структуру из стандарта IEEE 829. По этому стандарту она состоит из:

- test Plan;
- test Design Specification;
- test Case Specification;
- test Procedure Specification;
- test Incident Report;
- test log;
- test Summary Report.

Первые четыре документа представляют собой входную документацию. Из них берется вся необходимая информация для проведения тестирования и формирования отчетной документации. В отчетную документацию входят: test Incident Report, test log, test Summary Report. Ее формирование происходит на основе входных данных и данных, полученных в результате тестирования (рис.1):

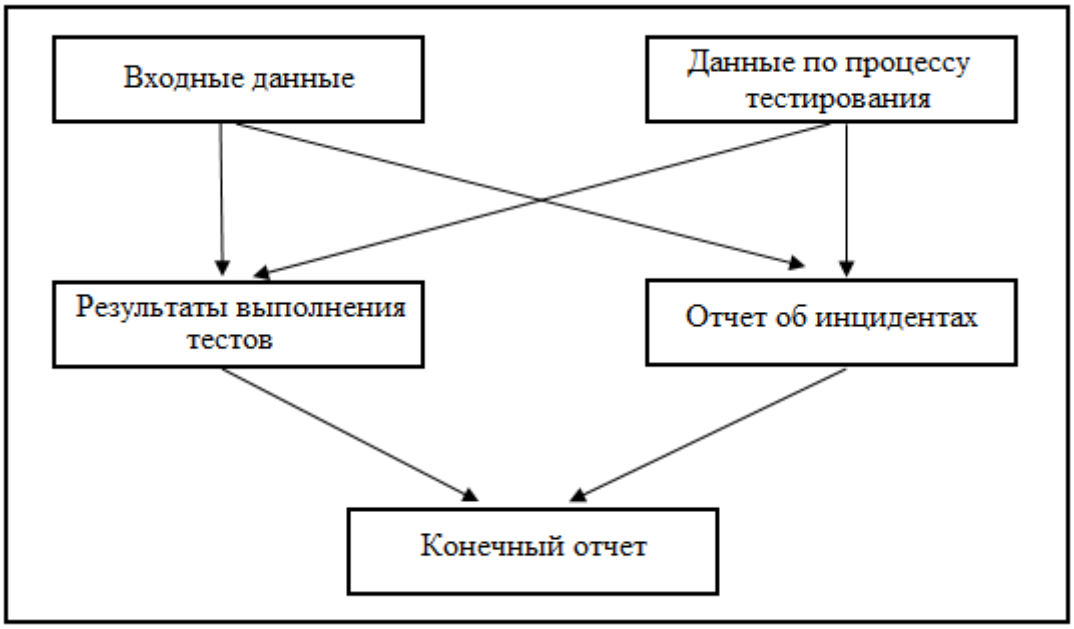


Рис.1 Структура создания отчетной документации

За сбор необходимой информации в процессе тестирования отвечает модуль логирования. В нем реализованы две основные функции:

1. Screen_Shoter – эта функция позволяет сделать снимок экрана при возникновении ситуации, которая не описана в выполняемом тесте, и сохранить его для отчета;
2. Saver_log – эта функция собирает и сохраняет все данные (в нужном формате) в ходе выполнения тестов.

После завершения процесса тестирования все данные, собранные модулем логирования передаются в модуль создания документации. Там данные анализируются и создаются конечные документы по процессу тестирования: результаты выполнения тестов, отчет по инцидентам, конечный отчет.

Когда создана выходная документация, она передается в модуль формирования нового регрессионного тестового набора.

Формирование регрессионного тестового набора

Для формирования регрессионного тестового набора используются данные из входной документации, которые добавленные специально в стандартные документы. Работа по этому методу происходит следующим образом:

- вносятся изменения в документ Test Design Specification;
- после получения отчетной документации проводится ее анализ;
- на основе данных анализа создается файл, в котором находится множество тестов, которые необходимо выполнить для регрессионного тестирования;
- полученный файл передается вместе с входной документацией при следующем тестировании продукта.

Связи элементов программы в документе Test Design Specification указываются следующим образом:

- в документ добавляется еще один столбец с данными;
- в этом столбце указываются номера тестов, которые необходимых для выполнения при изменении элемента программы.

Возможна ситуация, когда один и тот же тест добавляется в регрессионное множество несколько раз (когда изменены несколько элементов для проверки которых необходим этот тест). Это влияет на его приоритет: чем больше раз встречается тест – тем выше его приоритет.

При использовании рассматриваемого метода может возникнуть проблема, связанная с тем, что при составлении документа Test Design Specification в нем могут быть указаны не все связи элементов. Для ее решения при составлении документа анализируются:

- код программы;
- технические риски тестирования, в которых описываются:
    - связанные сущности;
    - определенные проблемы при подобных изменениях;
    - выявленные ранее проблемы, связанные с внесением изменений в данный функционал.

Анализ этих данных позволяет решить проблему формирования документов со связями элементов программы, а так же:

- исключает неполноту информации по программе;
- позволяет автоматизировать часть действий:
    - отбор тестов для проверки элементов, в которых были найдены ошибки;
    - отбор связанных тестов;

- отбор тестов, необходимых для проверки функций, измененных не из-за ошибки.

Структура автоматизированной системы

Для сокращения времени процесса тестирования и улучшения его качества предлагается автоматизировать часть описанных выше действий. Автоматизированная система, включающая в себя модули логирования, документирования и создания тестового набора имеет структуру, представленную на рис. 2.

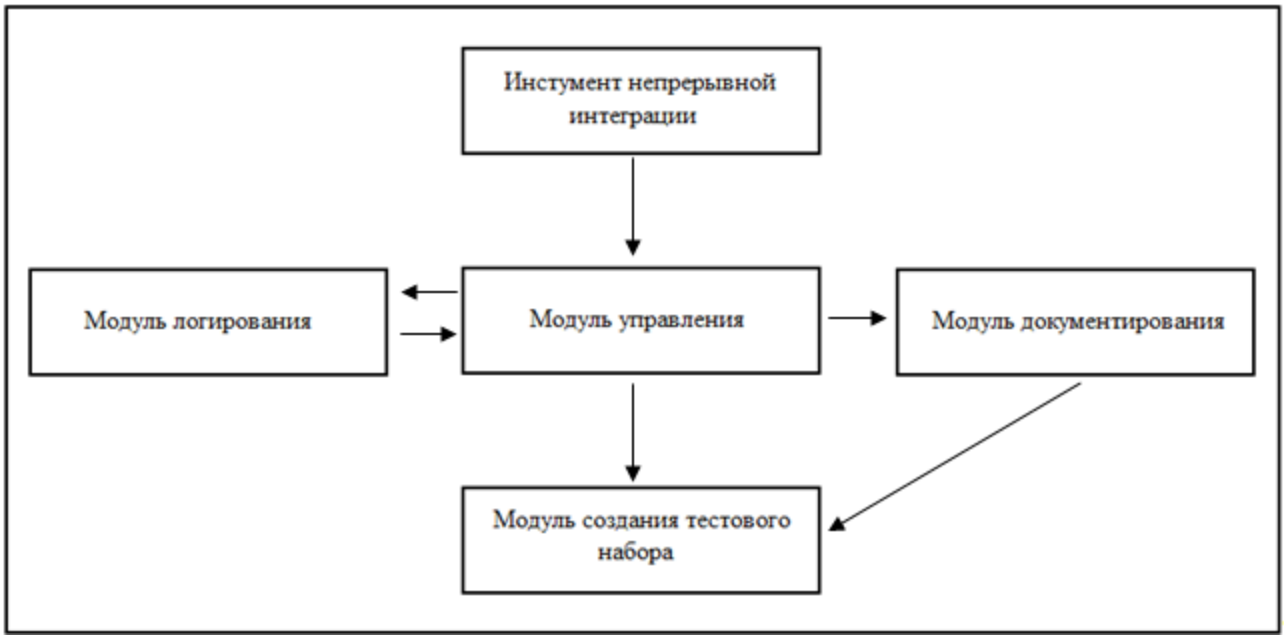


Рис. 2 Структура автоматизированной системы

Запуск процесса тестирования происходит в системе непрерывной интеграции вручную или по расписанию. После запуска система собирает новую сборку с последней версией входных данных (тесты, документация). Затем модуль управления начинает выполнять нужные тесты на сервере и параллельно запускает логгер. После выполнения всех тестов и записи данных по их выполнению запускается модуль документирования, который создает документы выходной документации (test Incident Report, test log, test Summary Report). Эта документация передается в модуль формирования тестового набора, где на основе ее и входной документации формируется множество тестов для регрессионного тестирования и записывается в специальный файл.

Заключение

Применение рассматриваемого метода формирования тестового набора повысит качество процесса регрессионного тестирования, так как этот метод безопасный и точный. Он так же является универсальным, то есть его можно использовать для большого множества программ с различными вносимыми в них изменениями, а автоматизированная система позволит сократить время проведения процесса тестирования и сделать отчетность по нему максимально наглядной.

Автоматизированное тестирование программных комплексов оптического моделирования

Е.Ю. Денисов
Институт прикладной математики им. М.В.Келдыша РАН, Москва
e-mail: eed@spp.keldysh.ru

Аннотация. В работе рассматриваются особенности тестирования современных программных комплексов оптического моделирования, и предлагается система автоматизированного тестирования таких комплексов.

Ключевые слова: программный комплекс оптического моделирования, компьютерная графика, автоматизация тестирования программ, автоматизация программирования.

Введение

Сложность современных программных комплексов оптического моделирования и синтеза фотореалистичных изображений достигла уровня, на котором классическое тестирование этих комплексов становится неэффективным, а часто и невозможным по ряду причин. Можно выделить следующие особенности тестирования таких комплексов:

- Много разнородных комплексов имеют в своей основе общие программные компоненты;
- Частые выпуски новых версий;
- Ограниченные ресурсы и время.

В частности, предлагаемые решения применяются для тестирования набора из десяти программных комплексов; частота выпуска новых версий – 50-70 версий в год; количество программных компонент – более 500; коллектив разработчиков состоит из 15 человек.

В таких условиях абсолютно необходимым является комплексный подход к автоматизации тестирования. Предложенные методы позволяют добиться приемлемого качества тестирования, а значит и приемлемого качества выпускаемых программных комплексов.

Автоматизированному тестированию подвергаются как отдельные программные компоненты (низкоуровневое тестирование), так и весь программный продукт в целом (высокоуровневое тестирование). В основном используется так называемое регрессионное тестирование, обеспечивающее совпадение поведения новой реализации системы и её предыдущей реализации.

Покомпонентное тестирование

После каждого изменения программной компоненты кроме непосредственного содержательного тестирования вновь добавленной функциональности проводится также её регрессионное автоматическое тестирование. Результат вычисления сравнивается с ожидаемым, и выдается заключение "тест пройден - не пройден". В последнем случае разработчиком исследуется ожидаемый и реальный вывод и производится исправление выявленных ошибок.

Для обеспечения автоматизации тестирования потребовалось создать несколько систем для сравнения результатов. Поскольку результатами работы различных программных компонент являются как текстовые, так и двоичные данные, в том

числе изображения, потребовалось создать подсистемы сравнения и анализа для каждого из этих типов данных. На рис. 1 приведён пример автоматического регрессионного тестирования программных компонент для работы с памятью и ввода-вывода, на рис. 2 – компоненты аппаратно-независимой реализации математических библиотек; в процессе тестирования проверяются как точность реализации математических функций, так и скорость работы данной компоненты.

```
--- Run base\test\bin\test_base.exe
Used virtual memory: 60 MB
Virtual memory limit: 4096 MB
Free available physycal memory: 4036 MB
Total available physycal memory: 4036 MB
Memory allocation result:
    Cur size     Max size  Class name
-----------------------------------
         0MB          0MB  UNNAMED
    SUCCESS!
--- Run base\plugin\test\bin\test_plugin.exe
    SUCCESS!
--- Run base\fields\testf\bin\testf.exe
    SUCCESS!
    Check Testf.iof...
    Check Copyf.iof...
--- Run base\fields\test_units\bin\units.exe
    SUCCESS!
--- Run base\iof\testh\bin\testh.exe
    SUCCESS!
    Check Test100.iof...
    Check Copy100.iof...
```

Рис. 1. Тестирование подсистем памяти и ввода-вывода

```
--- Run math\test\bin\math_test.exe
***Testing SSE version of Cos on [0,PI/2]***
The precision is 1.35103e-007
The acceleration is 14.401666
***Testing SSE version of Cos on [0, PI]***
The precision is 5.36442e-007
The acceleration is 6.695166
***Testing SSE version of Sin on [0, PI]***
The precision is 1.02073e-006
The acceleration is 7.321569
***Testing SSE version of Arcsin***
The precision is 6.76808e-005
The acceleration is 17.923254
***Testing SSE version of Arccos***
The precision is 6.77109e-005
The acceleration is 16.912662
***Testing the Binary search...***
The acceleration is 2.564422
    SUCCESS!
--- Run math\perf\test\bin\math_perf.exe
   Number of iterations 262144000
```

Рис. 2. Тестирование математических библиотек

Подсистема сравнения изображений обеспечивает качественный и количественный анализ различий в сравниваемых изображениях. Ниже приведен пример работы этой подсистемы: сравниваемое изображение (рис. 3), разница в графическом представлении для последующей оценки разработчиком (рис. 4), количественные данные: абсолютные и относительные ошибки в различных цветовых пространствах, гистограмма ошибочных пикселей (рис. 5).

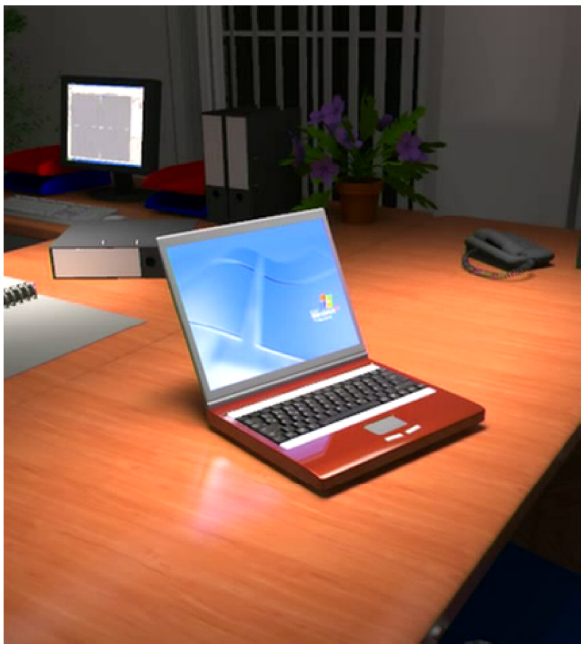


Рис. 3. Исходное изображение

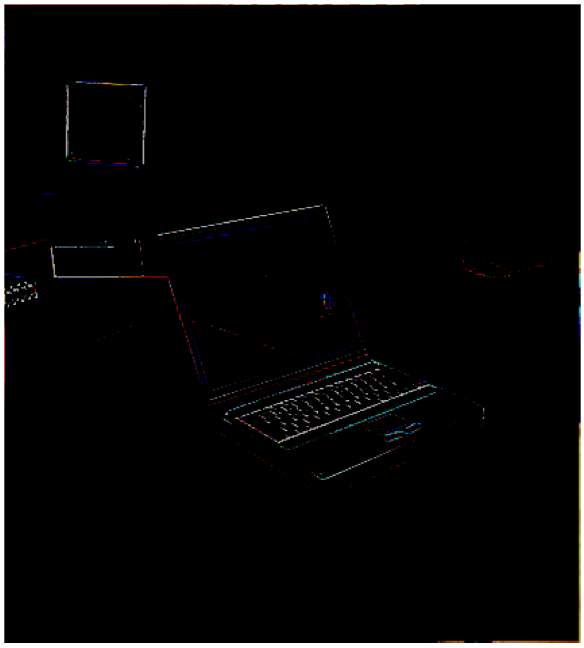


Рис. 4. Найденные отличия

```
type apr95.cmp|more

                Mean values of intensities

          Pattern         Tested      |Tested - Pattern|

RED       84.5914         84.5968          0.0054
GREEN     71.1868         71.2045          0.0177
BLUE      64.0259         64.0457          0.0198


          Abs. error (L2)   Rel. error (L2)

RED          0.49          0.0047
GREEN        0.48          0.0052
BLUE         0.47          0.0056
Total absolute RGB-error (L2):     1.44
Average relative RGB-error (L2):   0.0052
Average error in Lab space (CIE 1976) =   3.6450
Average error in Lab space (CIE 1995) =   1.3240


                        Histogram of differences

DIFFER   RGB % ( pixs)    RED % ( pixs)  GREEN % ( pixs)   BLUE % ( pixs)

  0       73.0 (54022)     80.2 (59383)     81.4 (60223)     81.9 (60599)
  1       25.2 (18645)     18.5 (13696)     17.3 (12834)     16.8 (12473)
  2        1.7 ( 1272)      1.2 (  898)      1.2 (  906)      1.2 (  888)
  3        0.1 (   85)      0.1 (   47)      0.1 (   61)      0.1 (   64)
  4        0.0 (    1)      0.0 (    1)      0.0 (    1)      0.0 (    1)

Maximum difference:              4
Allowed difference:              3
Image resolution:                225 x 329
Pixels below allowed difference: 100.0 (74024)
Pixels above allowed difference:   0.0 (    1)
Images of differences generated: YES
```

Рис. 5. Количественное представление отличий

Важной особенностью является то, что сравнение изображений происходит в автоматическом пакетном режиме; допустимый порог погрешности (разницы в пикселях) задается предварительно, и при превышении этого порога тестируемая подсистема считается ошибочной и подлежащей дальнейшему анализу и исправлению ошибок. Весь процесс тестирования автоматически записывается в журнал, содержащий список протестированных подсистем и результаты тестирования каждой из них.

Тестирование готового программного продукта

Помимо покомпонентного низкоуровневого тестирования готовый программный продукт проходит также высокоуровневое тестирование, обеспечивающее проверку качества совместной работы компонент и их правильного использования разработчиком. Такое тестирование проводится также в пакетном режиме с использованием специальных модулей пакетного выполнения, являющихся частью программного комплекса, либо с использованием скриптов (в тех комплексах, которые поддерживают их выполнение).

На рис. 6 представлен пример тестирования одновременно нескольких функциональностей программного продукта за один шаг. В данной специально созданной тестовой сцене проверяется правильность функционирования различных алгоритмов наложения текстур на различные типы и формы поверхностей. Для анализа правильности результата вновь используется подсистема сравнения изображений, описанная выше.

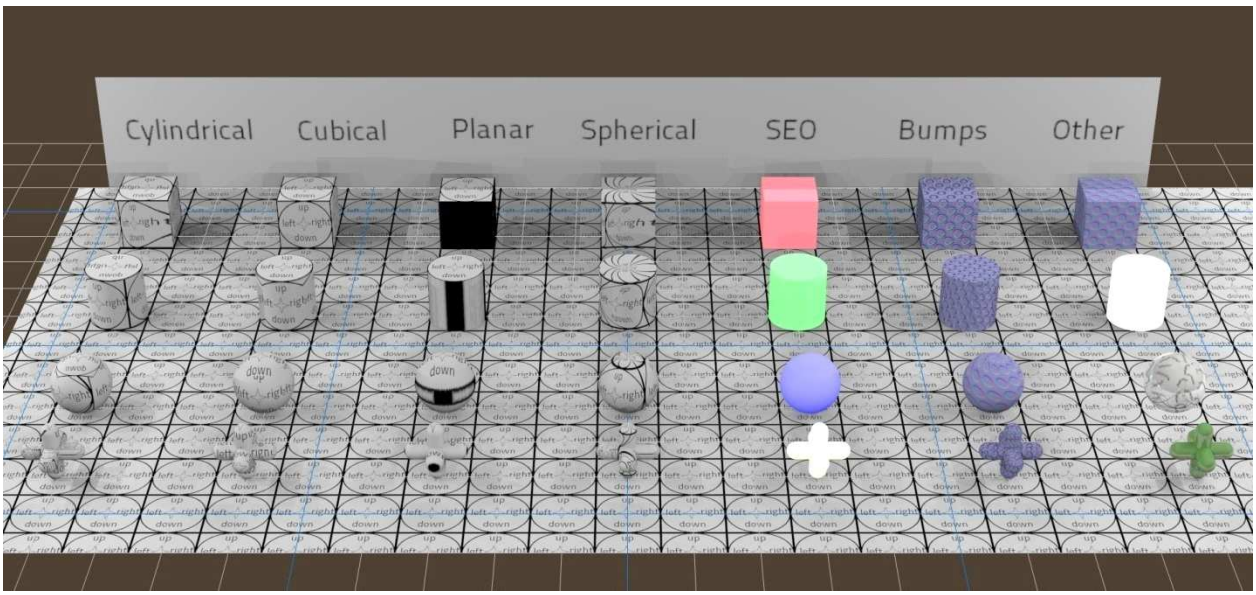


Рис. 6. Тестовая сцена для алгоритмов наложения текстур

Для тестирования интерактивных компонент, т.е. элементов интерфейса, требующих взаимодействия с пользователем, система тестирования предоставляет разработчику возможность проверить функционирование компоненты в интерактивном режиме. На рис. 7 и 8 показаны соответственно подсистемы отображения прогресса вычисления и отображения графиков, рис. 9 содержит тестовую анимацию подсистемы трехмерного отображения объектов.

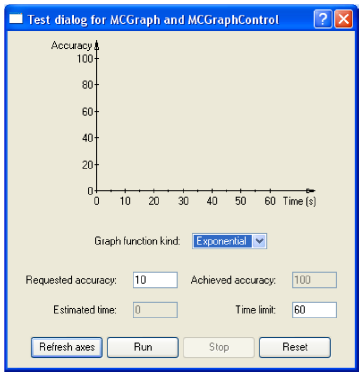


Рис. 7. Тест диалога прогресса вычислений

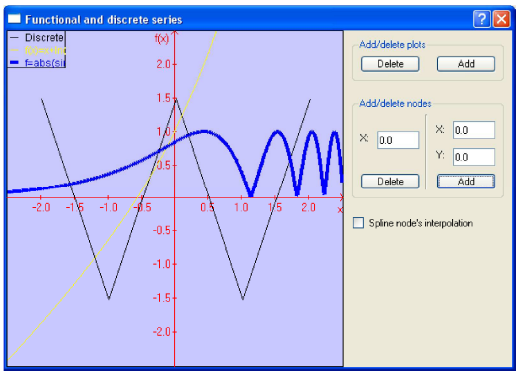


Рис. 8. Тест подсистемы отображения графиков

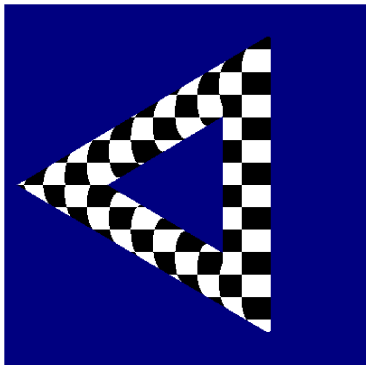


Рис. 9. Тест подсистемы визуализации

Параллельное тестирование

Предложенный подход помимо полной автоматизации тестирования позволяет производить тестирование на нескольких компьютерах параллельно, тем самым позволяя еще больше увеличить производительность тестирования и уменьшить время, затрачиваемое на него. Используется как так называемое "натуральное распараллеливание", когда разные компоненты тестируются на разных компьютерах, так и параллельное вычисление в самих программных комплексах, которое дополнительно тестирует и саму функциональность параллельных вычислений.

Заключение

Предлагаемая система тестирования была использована в реализации нескольких систем синтеза реалистичных изображений и оптического моделирования, в частности, для систем, представленных в [1], [2], [3], [4], [5]. Использование описанного подхода позволило ускорить разработку и реализацию этих систем и одновременно снизить трудозатраты на их создание, тестирование и сопровождение.

Список литературы

1. Б.Х. Барладян, А.Г. Волобой, В.А. Галактионов, Л.З. Шапиро. Интерактивная визуализация результатов моделирования освещенности // "Информационные технологии в проектировании и производстве", № 4, 2011, с.58-61.
2. Д.Д.Жданов, И.С.Потемин, В.А.Галактионов, Б.Х.Барладян, К.А.Востряков, Л.З.Шапиро. Спектральная трассировка лучей в задачах построения фотореалистичных изображений // "Программирование", № 5, 2011, с. 13-26.
3. А.Г.Волобой, С.В.Ершов, С.Г.Поздняков. Интерактивное моделирование автомобильных красок // Труды 22-й Международной Конференции по Компьютерной Графике и Зрению, Московский Государственный Университет им. М.В.Ломоносова, 01-05 октября 2012 года, с.242-247.
4. Ignatenko A., Barladian B., Dmitriev K., Ershov S., Galaktionov V., Valiev I., Voloboy A. A Real-Time 3D Rendering System with BRDF Materials and Natural Lighting // Proc. 14th International Conference on Computer Graphics and Vision GraphiCon-2004, Russia, Moscow, September 6 -10, 2004, p. 159-162.
5. Барладян Б.Х., Волобой А.Г., Вьюкова Н.И., Галактионов В.А., Дерябин Н.Б. Моделирование освещенности и синтез фотореалистичных изображений с использованием Интернет технологий // "Программирование", № 5, 2005, с.66-80.

Автоматизированное тестирование интерактивных программных комплексов

Е.Ю. Денисов, И.А. Калугина
Институт прикладной математики им. М.В.Келдыша РАН, Москва
eed@spp.keldysh.ru, qik@gin.keldysh.ru

Аннотация. В работе рассматривается система записи и воспроизведения действий пользователя и применение её для регрессионного тестирования современных программных комплексов компьютерной графики и оптического моделирования.

Ключевые слова: программный комплекс оптического моделирования, компьютерная графика, автоматизация тестирования программ.

Введение

Сложность современных программных систем требует автоматизации их разработки и сопровождения на всех этапах жизненного цикла программного продукта. Одним из важнейших этапов является тестирование. Описываемая ниже система записи и воспроизведения интерактивных действий пользователя названа TIPE – Testing Interactive Program Execution. Её функциональность можно сравнить с системами записи и воспроизведения макро команд, доступными в некоторых редакторах, однако возможности, реализуемые TIPE, гораздо шире.

Анализ существующих решений

Авторами были протестированы десятки программных пакетов, позволяющих в том или ином виде записать и воспроизвести действия пользователя в интерактивной программе. Среди них: AutoIt [1], Test Automation FX [2], TestComplete [3] и другие. К сожалению, обнаруженные недостатки не позволили полноценно использовать для тестирования создаваемых программных комплексов ни один из этих пакетов. Среди этих недостатков: запись действий пользователя в абсолютных координатах (не позволяет воспроизводить записанную сессию на другом компьютере с другим разрешением экрана); неспособность распознать UI элементы пакета Qt; невозможность записи сессии (сценарий должен быть предварительно написан на специальном языке); отсутствие поддержки новейших версий Windows, а также ошибки в работе (иногда приводящие к сбою всей OS). Такая ситуация потребовала создания собственного средства записи и воспроизведения действий пользователя.

Применения и условия

Все действия, производимые пользователем в программном комплексе, перехватываются и записываются модулем TIPE и сохраняются в файле. Позднее этот файл может быть воспроизведен, в том числе на другом компьютере, отличном по характеристикам от компьютера, использованного при записи. Выполнение записанных действий в программном комплексе произойдет в том же порядке, в каком оно было записано.

Основные применения системы TIPE:

- Демонстрация возможностей приложения, не требующая знаний о них;
- Использование в качестве интерактивного дополнения к инструкции по эксплуатации приложения;

- Средство для регрессионного тестирования (после внесения изменений в реализацию поведение приложения должно совпадать с предыдущей версией);
- Средство для визуального представления действий пользователя (например, в процессе обнаружения и исправления ошибок в приложении).

Система TIPE производит запись событий, которые изменяют состояние приложения, и реакцию приложения на эти события. Приложение рассматривается как конечный автомат. Записываемые и воспроизводимые события являются входными данными для этого автомата. Среди таких событий не только события, непосредственно выполняемые пользователем (движения мыши и нажатия клавиш на мыши и клавиатуре), но и другие события, приводящие к смене состояний приложения: сигналы программных таймеров, системные сообщения OS и прочие. Сам модуль TIPE и его использование являются “прозрачными” для программиста – для использования функциональности TIPE в процессе программирования не нужно прилагать никаких специальных усилий. Модуль TIPE автоматически перехватывает значимые события и реакцию приложения и производит их запись.

Для успешного функционирования системы TIPE необходимо выполнение следующих условий:

- Приложение при записи и воспроизведении должно быть тем же самым;
- Начальное состояние приложения должно совпадать;
- Изменения состояний приложения, вызванные одинаковыми действиями, должны быть одинаковыми (в частности, приложения, использующие генераторы псевдо-случайных чисел, должны иметь возможность повторять ранее записанную последовательность случайных чисел);
- Все события, приводящие к изменению состояния приложения, должны записываться TIPE;

При нарушении одного из условий воспроизведение записанной сессии невозможно. Такими нарушениями могут быть: различное начальное состояние приложения (например, отличаются загруженные данные); различное поведение приложения в ответ на одинаковые действия; изменение состояний приложения зависит от оборудования; критические ситуации в процессе воспроизведения, отсутствовавшие в процессе записи (например, нехватка дискового пространства).

Основные свойства

Основными свойствами системы TIPE являются:

- Надежное воспроизведение записанных действий – гарантируется, что при соблюдении вышеуказанных условий записанная сессия работы приложения может быть воспроизведена (независимо от того, на каких компьютерах производится запись и воспроизведение). При изменении версии приложения по возможности обеспечивается обратная совместимость: если изменения в приложении не влияют на записанную последовательность изменения состояний приложения, то новая версия приложения будет успешно воспроизводить сессию TIPE, записанную на предыдущей версии приложения.
- Проверка условий воспроизведения – при старте воспроизведения система проверяет соответствие нескольких геометрических параметров (таких как разрешение экрана, размеры шрифтов, размеры окон приложения).

- Проверка совпадения последовательности событий – в процессе воспроизведения система проверяет совпадение записанных событий с событиями, генерируемыми приложением при изменении его состояния.
- Операции с мышью и клавиатурой визуализируются – движения мыши визуализируются перемещением курсора, нажатия клавиш мыши и клавиатуры отображаются на экране в виде всплывающих сообщений.
- Имеется возможность добавления комментариев – в процессе записи сессии возможно в любое время вставить текстовый комментарий, который будет показан во всплывающем окне в процессе воспроизведения. Комментарии могут быть изменены, удалены или вставлены так же после окончания записи, путем исправления файла, содержащего записанную сессию TIPE.

Рис. 1. Пример воспроизведения сессии

При невозможности дальнейшего воспроизведения сессии (например, несовпадение начальных условий, или несовпадение реакции приложения с записанной) воспроизведение останавливается и выводится сообщение об ошибке. Кроме того, в процессе записи и воспроизведения создается лог-файл, содержащий все сообщения об ошибках.

Скорость воспроизведения регулируется в широких пределах, в любой момент в процессе записи и воспроизведения процесс может быть временно приостановлен, или полностью прекращен.

На рис. 1 приведен пример воспроизведения сессии, демонстрирующей работу программы с источниками освещения. Показаны: окно с описанием текущей

операции (комментарием), визуальное отображение нажатия мыши и окно управления параметрами воспроизведения.

Сессия TIPE хранится в файле текстового формата, что допускает внесение изменений без перезаписи всей сессии, путем изменения, добавления или удаления частей сессии. Файл имеет малый размер, что допускает его пересылку любым электронным способом.

Возможна запись новой сессии в процессе воспроизведения имеющейся. В случае остановки воспроизведения запись новой сессии может быть продолжена по другому сценарию.

Файл TIPE может содержать идентификационную информацию: название, автор, дата записи, наименование и версия приложения, другая текстовая информация.

Запись сессии приложения производится, даже если явной команды записи нет. В таком случае сессия записывается во временный файл; это позволяет в дальнейшем воспроизвести действия, приведшие к трудновоспроизводимой ошибке в приложении.

Регрессионное тестирование

В процессе выпуска новых версий программного продукта большое значение имеет регрессионное тестирование – обеспечение совпадения поведения новой реализации приложения и его предыдущей реализации. Система TIPE позволяет обеспечить автоматизированное регрессионное тестирование компонент пользовательского интерфейса. Для каждой новой версии приложения производится воспроизведение всех имеющихся сессий TIPE. Наблюдение за процессом тестирования не является обязательным – результаты тестирования проверяются по окончании процесса, путем анализа лог-файлов и сравнения результатов работы программы (обычно файлов, содержащих изображения или числовые данные). Тестирование происходит в пакетном режиме, круглосуточно, с использованием нескольких компьютеров – такой подход позволяет минимизировать время и трудозатраты.

Заключение

Описанная система успешно используется для тестирования и создания презентаций возможностей программных комплексов компьютерной графики и оптического моделирования Inspirer [4] и Specter [5]. Она позволяет значительно сократить время и снизить трудозатраты на выпуск, тестирование и сопровождение новых версий.

Список литературы

1. http://www.autoitscript.com
2. http://www.testautomationfx.com
3. http://smartbear.com/products/qa-tools/automated-testing-tools
4. Kopylov E., Khodulev A., Volevich V. The Comparison of Illumination Maps Technique in Computer Graphics Software // Proc. 8th International Conference on Computer Graphics and Visualization, Russia, Moscow, September 7-11, 1998, p. 146-153.
5. Khodulev A., Kopylov E. Physically accurate lighting simulation in computer graphics software // Proc. 6th International conference on Computer Graphics and Visualization, Russia, St.Petersburg, July 1-5, 1996, p. 111-119.

Управление требованиями к программному обеспечению для поддержки контроля достижения целей

В.О. Каковский, Т.Н. Заболотняя
кафедра программного обеспечения компьютерных систем,
Национальный технический университет Украины
«Киевский политехнический институт»
г. Киев, Украина
slavkoff@gmail.com

Аннотация. Данная статья посвящена вопросу управления требованиями к программному обеспечению, предназначенному для поддержки контроля достижения целей, а именно решению задач идентификации, анализа и документирования требований к ПО, являющихся составляющими процесса управления требованиями, с целью разработки спецификации требований к программному обеспечению (SRS). На основании результатов исследования существующих программных средств, используемых для учета целей, а также анализа отзывов их пользователей сформулированы требования к разрабатываемой системе.

Ключевые слова: выбор задач для исполнения, контроль достижения целей, управление задачами, управление требованиями.

Актуальность прикладной задачи

Ежедневно миллионам людей приходится сталкиваться с большими объемами информации – начиная от специализированной литературы и заканчивания новостями в социальных сетях. Соответственно, возрастает и количество задач, связанных с обработкой этих данных и принятием необходимых решений. По мнению авторов, наиболее актуальной для человека является та информация, использование которой влияет на скорость достижения поставленных им целей. К этим сведениям относятся:

1. Информация о поставленной цели – желательно, чтобы цель соответствовала SMART-критериям [1] (тогда ее достижение проще всего отслеживать).
2. Перечень шагов или действий, которые следует предпринять для достижения данной цели.
3. Данные о выполненных и начатых задачах, которые описывают текущее соостояние достигаемой цели и позволяют оценивать прогресс.

Анализируя эти сведения, человек имеет возможность выбрать для выполнения именно те приоритетные задачи, успешное завершение которых приведет к достижению необходимого результата.

Автоматизация обработки вышеописанных данных должна предоставить возможность оперативного получения ответа на вопрос «что я должен сделать в ближайшее время для достижения этой цели». Особенно важным это становится тогда, когда выделение подобной информации из обширного потока входных данных перестает быть тривиальной задачей.

Таким образом, разработка программного обеспечения, в функции которого входят поддержка учета шагов, направленных на достижение целей, приоритизация этих шагов, а также оценка прогресса в получении необходимого результата, является актуальной задачей.

Для того, чтобы разрабатываемое ПО удовлетворяло потребностям пользователей, необходимо тщательно определить требования к системе. Решению именно этой задачи посвящена статья.

Классификация требований к ПО

В соответствии со стандартом разработки требований ISO/IEC 29148, требование — это утверждение, которое идентифицирует эксплуатационные, функциональные параметры, характеристики или ограничения проектирования продукта или процесса, а также является однозначным, проверяемым и измеримим.

Существует несколько подходов к классификации требований, выдвигаемых к программному обеспечению:

1. Классификация требований согласно SWEBOK [2] выделяет следующие категории: требования к продукту и процессу его создания, функциональные и нефункциональные, системные требования, а также независимые свойства (требования, которые не могут быть адресованы одному из компонентов системы, а проявляются при их взаимодействии).
2. Схема Лефингвелла [3] отражает подход, рекомендованный RUP, и имеет следующие уровни требований: потребности, характеристики продукта (features) и, собственно, требования к ПО.
3. Модель Вигерса [4] является удобным способом организации представления и группировки требований.

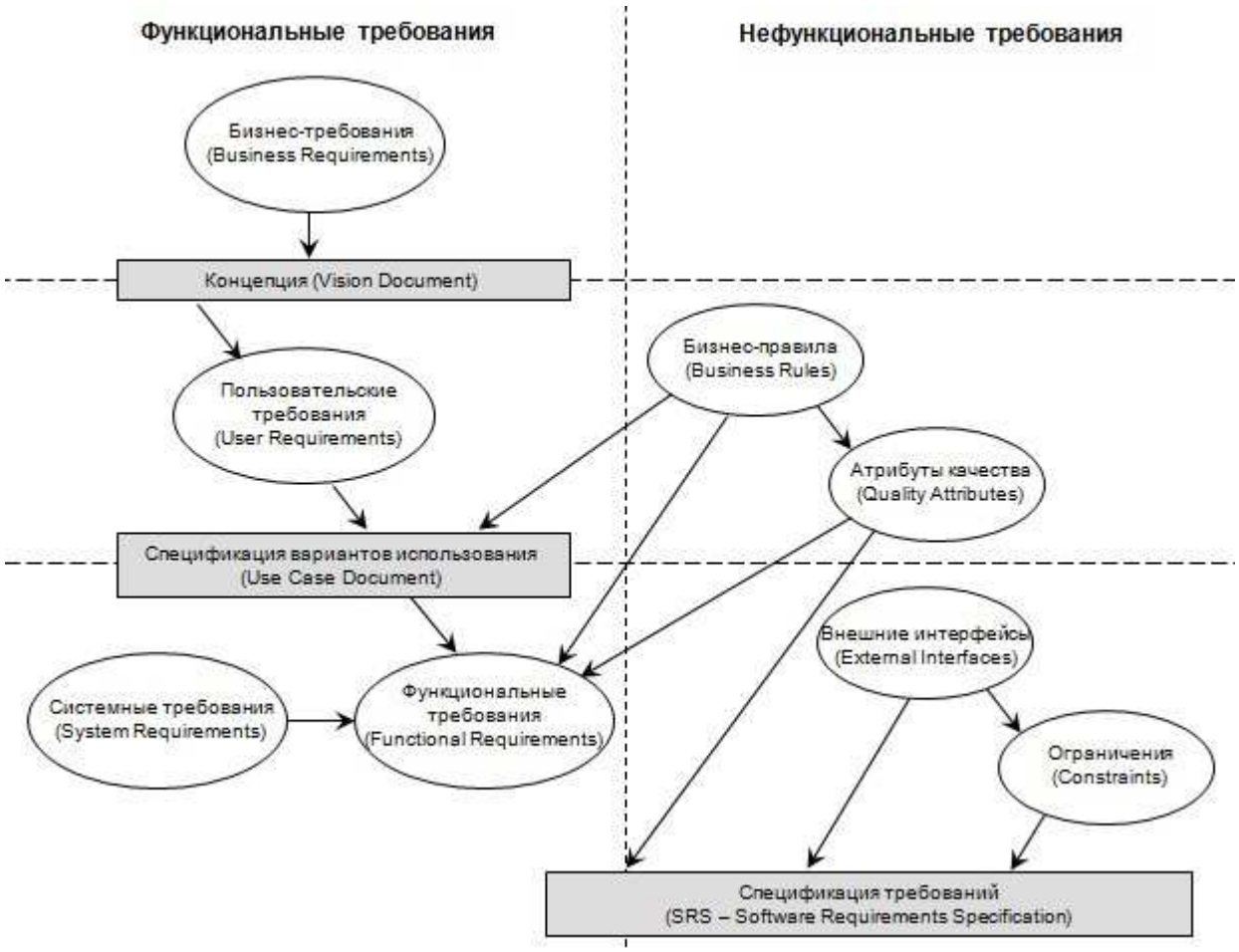


Рис. 1. Уровни требований согласно модели Вигерса

В качестве основы для определения спецификации требований к разрабатываемому ПО авторами выбран именно последний упомянутый подход, т.к. модель Карла Вигерса позволяет отобразить требования на структуры документов, определяемые методологиями и стандартами IEEE 830, ГОСТ 34.

Основная идея классификации требований согласно модели Вигерса базируется на определении групп (функциональные, нефункциональные) и уровней требований к продукту (см. рис. 1).

Верхний уровень описывает саму задачу или проблему, которая должна быть решена, благодаря системе. Суть этих *бизнес-требований* сводится к постановке задачи и описанию назначения програмного обеспечения в повествовательной форме; называется этот документ концепцией проекта (Vision Document).

Следующий уровень – *пользовательские требования (User Requirements)*, которые детально описывают цели/задачи пользователей системы, которые должны достигаться/выполняться пользователями при помощи создаваемой программной системы. Эти требования часто представляют в виде диаграмм вариантов использования (Use Cases).

Нефункциональные требования этого же уровня – это *бизнес-правила* и *атрибуты качества*. *Бизнес-правила (Business Rules)* – требования, которые основываются на корпоративных регламентах, политиках, стандартах, законодательных актах, алгоритмах, и т.д. *Атрибуты качества (Quality Attributes)* – дополнительные характеристики продукта, важные для пользователей и/или разработчиков. Функциональные и нефункциональные требования второго уровня описываются в Спецификации вариантов использования (Use Case Document).

Третий уровень требований – собственно *функциональные требования* и *системные требования*. *Функциональные требования (Functional Requirements)* – определяют функциональность (поведение) программной системы, которая должна быть создана разработчиками для предоставления возможности выполнения пользователями своих обязанностей в рамках бизнес-требований и в контексте пользовательских требований. *Системные требования (System Requirements)* классифицируются как составная часть группы функциональных требований и описывают высокоуровневые требования к программному обеспечению, содержащему несколько или много взаимосвязанных подсистем и приложений.

Существуют следующие виды *нефункциональных требований* этого уровня:

1. *Ограничения (Constraints)* - условия, ограничивающие выбор возможных решений по реализации отдельных требований или их наборов.
2. *Внешние интерфейсы (External Interfaces)* – требования, которые содержат конкретизацию аспектов взаимодействия разрабатываемого ПО с другими системами, например, операционной средой, аппаратным оборудованием, пользователями и т.д.

Все вместе функциональные требования и нефункциональные требования составляют последнюю ступень модели Вигерса – Спецификацию требований к программному обеспечению (Software Requirement Specification).

В данной статье основное внимание уделено выявлению и формулировке требований первых двух уровней модели Вигерса, так как их достаточно для понимания общей картины функционирования ПО. Получение конечного программного продукта не является целью исследования, поэтому требования третьего уровня тут не определены.

Ниже приведены рекомендации к разрабатываемому ПО, на основании которых в данной статье сформулированы требования двух вышеупомянутых уровней модели Вигерса.

Характеристики ПО

Вначале определим базовые понятия предметной области, используемые далее в статье.

Веха – контрольная точка или ключевой момент (например, переход к следующей фазе) во время достижения цели.

Декомпозиция – процесс, в результате которого решение сложной проблемы представляется в виде решения серии более простых взаимосвязанных задач.

Задача – деятельность, которая должна быть завершена в указанный срок.

Игрофикация - применение подходов, характерных для компьютерных игр, в программных инструментах для неигровых процессов с целью повышения вовлечённости пользователей в решение прикладных задач [5].

Цель – является желаемым результатом действия или действий человека.

Ценность – любое явление, которое имеет значение для человека; то, ради чего он действует, тратит силы и живет. Примеры ценностей: здоровье, семья, дружба, самореализация, работа, социальный статус, свобода, материальное благосостояние.

В результате исследования предметной области, проведенного ранее авторами [6], были определены следующие рекомендации, которые целесообразно принять во внимание при разработке нового ПО для поддержки контроля достижения целей:

1. Соответствие целей методике SMART и возможность присвоения задачам приоритета. Приоритет задачи является именно тем критерием, на который можно опираться при выборе задач для выполнения среди множеств возможных вариантов.
2. Обеспечение управления как целями, так и задачами, выполнение которых приближает к достижению желаемого результата. Удобным является выделение вех. Декомпозиция цели на задачи позволяет получить список простых шагов, отслеживание выполнения которых делает возможным наблюдение за прогрессом в целом.
3. Поддержка работы програмного обеспечения в наиболее распространенных операционных системах (Linux, Windows, Mac OS).
4. Возможность просмотра прогресса по достижениюкак конкретной цели, так и группы целей, которые соответствуют одной и той же жизненной ценности.
5. Возможность просмотра ожидаемой загруженности человека на определенный период времени. Позволяет оценить вероятность достижения определенной цели или выполнения задач в течение определенного промежутка времени. Данный функционал должен учитывать все актуальные цели и задачи.

Также после изучения возможностей существующих программных средств был определен перечень дополнительных функций, которые стоит реализовать в ПО, предназначенном для поддержки контроля достижения целей:

1. Интеграция с Google Calendar – позволит визуализировать задачи и вехи в виде событий на календаре.
2. Возможность выбора цели из определенного списка популярных целей, который формируется пользователями ресурса.
3. Наличие настраиваемого генератора отчетов, который позволит получить статистику по целям и задачам.

В следующем разделе статьи описанные выше рекомендации преобразованы в требования к программной системе для поддержки контроля достижения целей.

Формализованные требования к системе

Первый уровень требований

Бизнес-требования к системе можно представить в виде Vision:

Программное обеспечение предназначено для поддержки контроля достижения целей. ПО должно предоставить возможность хранить информацию о целях, которые соответствуют SMART-критериям. Информация о цели включает данные о тех задачах, выполнение которых ведет к достижению результата. ПО позволит планировать путь достижения цели, отслеживать прогресс в получении необходимого результата, а так же оценивать ожидаемую загруженность человека в определенный период времени. Система должна быть реализована в виде веб-приложения.

Как видим, бизнес-требования к системе полностью совпадают с постановкой задачи разработки, приведенной выше.

Второй уровень требований. Функциональные требования

Требования пользователей удобно сопровождать диаграммами вариантов использования (Use Case). Для разрабатываемого ПО определен следующий перечень вариантов использования:

1. Ознакомиться с информацией о возможностях системы
2. Зарегистрироваться
3. Авторизоваться
4. Пройти интерактивный тур
5. Работать с главным меню
6. Управлять данными о ценностях, целях, задачах
7. Вносить данные о выполненных задачах и достигнутых целях
8. Управлять статистикой.

Диаграммы вариантов использования системы, которые несут наибольшую тематическую нагрузку, приведены ниже.

Use Case 1: “Пройти интерактивный тур”. После первой авторизации пользователю предлагается пройти интерактивный тур по возможностям приложения (см. рис. 2). Тур состоит из следующих шагов:

1. Пользователь определяет жизненные ценности, к которым будут относиться цели.
2. Пользователь вносит в систему информацию о SMART-целях и мечтах (цели, которые не соответствуют критериям SMART).
3. Пользователь совершает декомпозицию целей на задачи.

В результате прохождения тура пользователь ознакамливается с ключевыми функциями ПО. Также в любой момент использования системы он имеет возможность снова пройти этот тур.

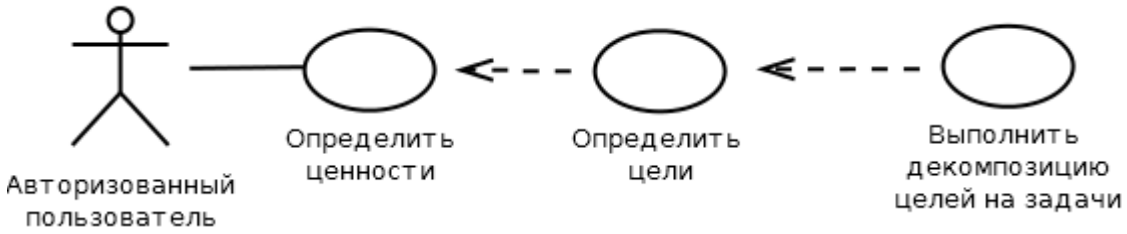


Рис. 2. UseCase “Пройти интерактивный тур”

Use Case 2: “Управлять данными о ценностях” – описывает работу пользователя с объектом класса «ценность» (см. рис. 3).

1. Пользователь просматривает список ценностей.
2. Пользователь создает/редактирует/удаляет экземпляр класса ценность.
3. Пользователь просматривает список целей, которые соответствуют выбранной ценности.
4. Пользователь просматривает статистику о ценности.

Структура объектов (ценности-цели-задачи) системы для поддержки контроля управления целей реализована в веб-приложении Lifetick [7].

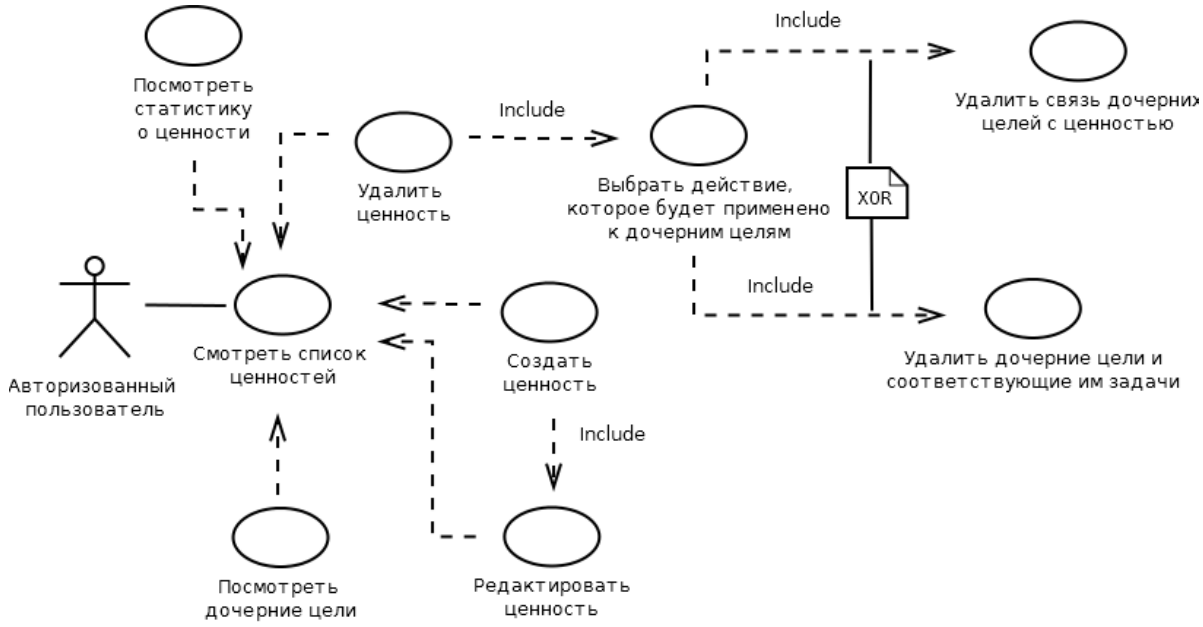


Рис. 3. Use Case “Управлять ценностями”

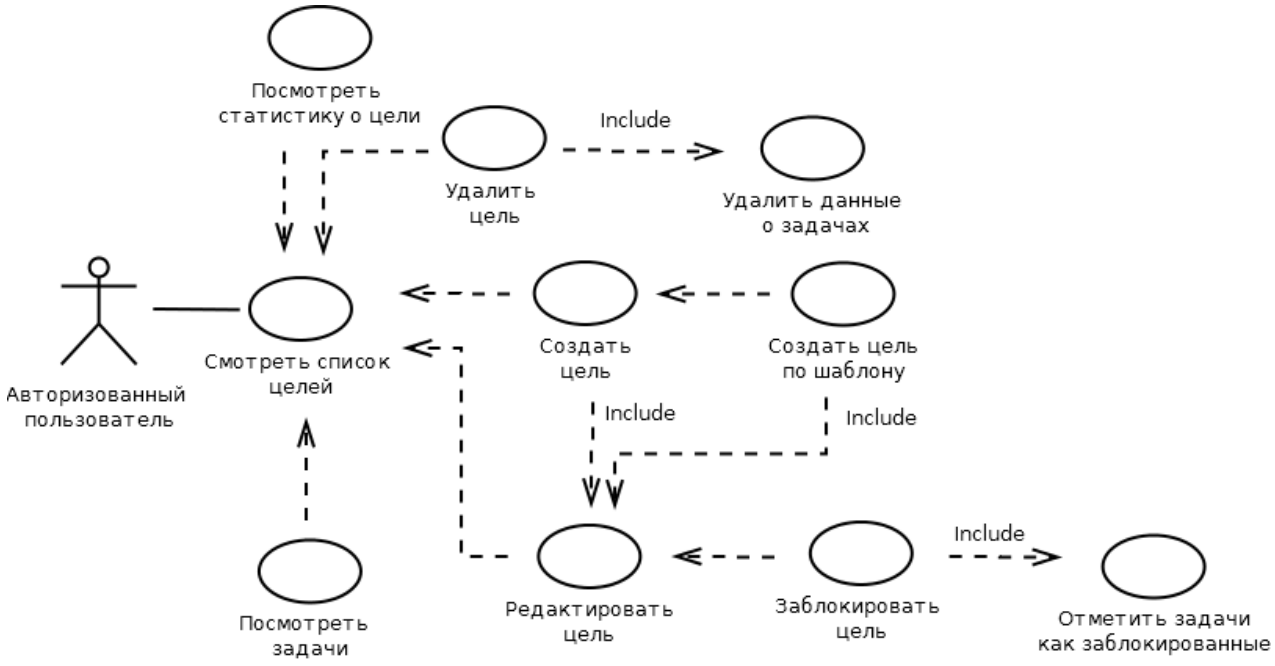


Рис. 4. Use Case “Управлять целями”

<u>Use Case 3: “Управлять данными о целях”</u> – описывает работу пользователя с объектом класса «цель» (см. рис. 4).

1. Пользователь просматривает список целей.
2. Пользователь создает/редактирует/удаляет экземпляр класса «цель».
3. Пользователь отмечает цель как уже не актуальную, то есть блокирует ее.

4. Пользователь просматривает список задач, которые соответствуют выбранной цели.
5. Пользователь просматривает статистику о цели.

Use Case 4: “Управлять задачами” – описывает работу пользователя с объектом класса «задача» (см. рис. 5).

1. Пользователь просматривает список задач.
2. Пользователь создает новые списки задач.
3. Пользователь создает/редактирует/удаляет экземпляр класса задача.
4. Пользователь отмечает цель как неактуальную (блокирует ее).
5. Пользователь просматривает статистику о задаче или списке задач.

Use Case 5:“Управлять статистикой” - описывает работу пользователя со статистикой (см. рис. 6).

1. Пользователь просматривает общую статистику достигнутых целей.
2. Пользователь просматривает прогресс.
3. Пользователь просматривает данные о планируемой загруженности.

Функционал «просмотр планируемой загруженности» является уникальным относительно проанализированных систем.

Use Case 6: “Работать с главным меню” – описывает навигацию по системе (см. рис. 7).

1. Пользователь управляет объектами (ценностями, целями, задачами).
2. Пользователь управляет статистикойданных о достижении целей.
3. Пользователь формирует список задач для исполнения.
4. Пользователь просматривает календарь событий.

Реализация функции «Сформировать список задач для исполнения» обеспечивает оперативное получение ответа на вопрос «что я должен сделать в ближайшее время для достижения своих целей».

Второй уровень требований. Нефункциональные требования

Бизнес-правило в системе одно – целями являются только те цели, которые можно поставить в соответствии критериям SMART.

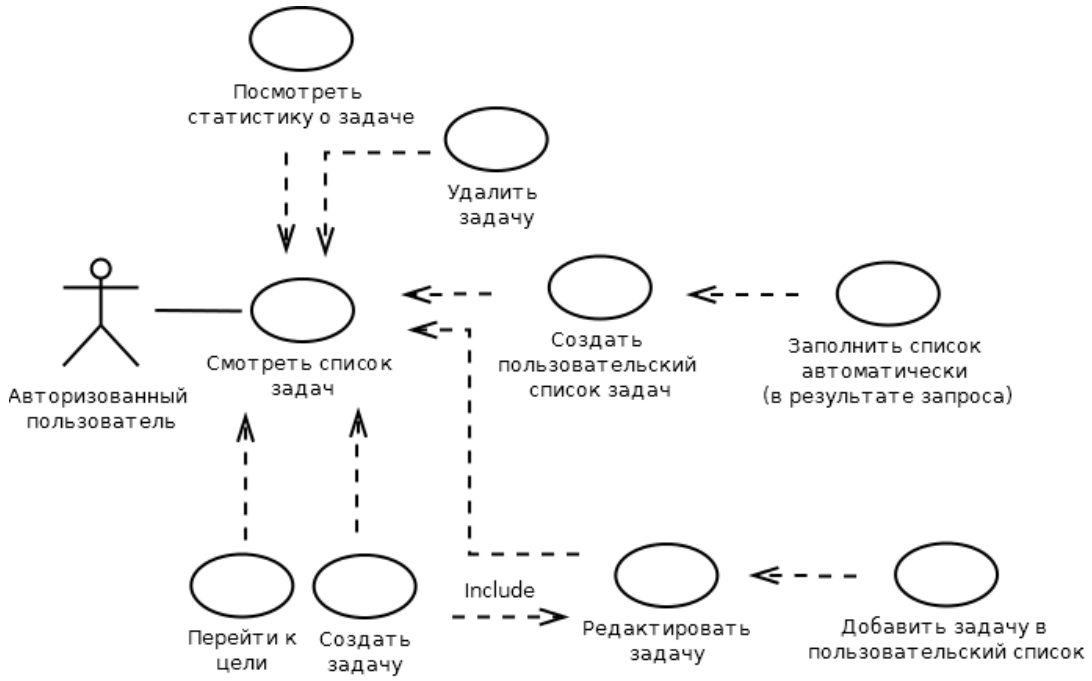


Рис. 5. Use Case “Управлять задачами”

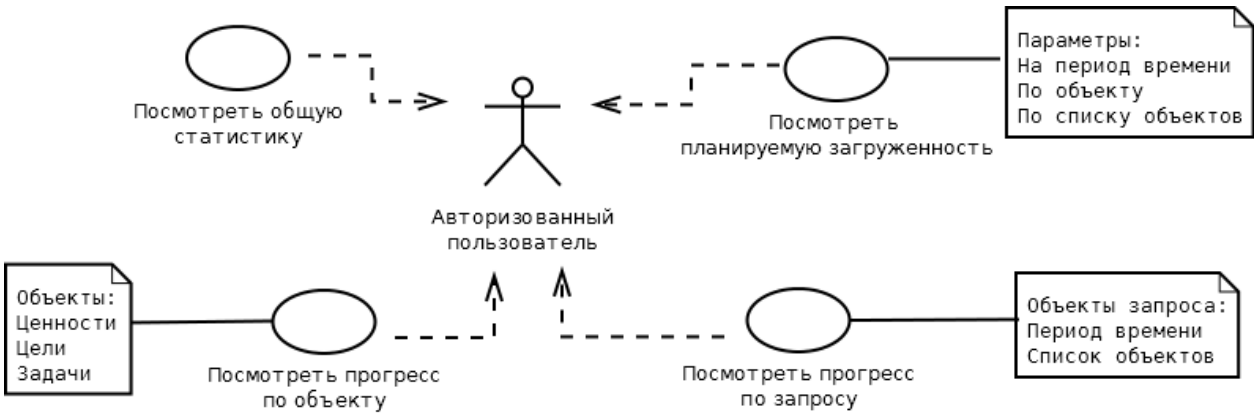


Рис. 6. Use Case “Управлять статистикой”

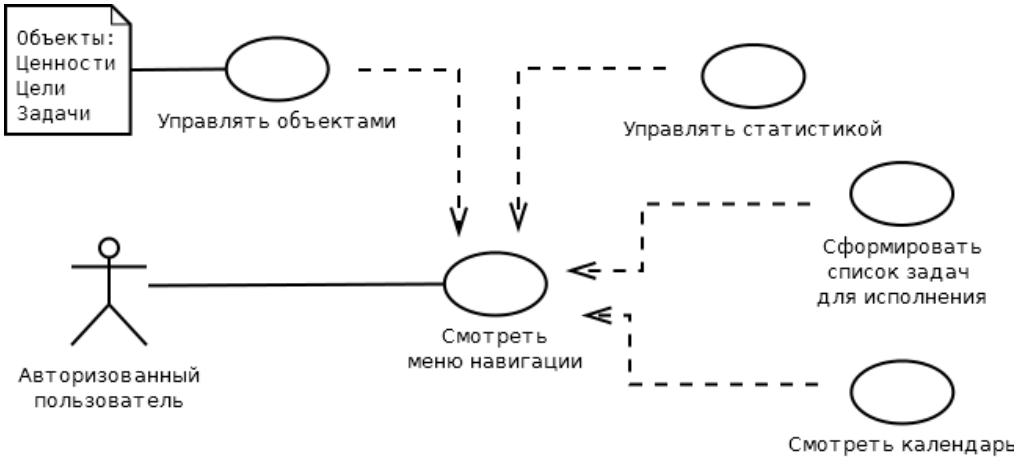


Рис. 7. Use Case “Работать с главным меню”

Атрибуты качества описывают важные для пользователей характеристики продукта. Отметим, что требования, описываемые в этом разделе, сделаны на основе приведенных выше рекомендаций к ПО, поэтому они не содержат всех без исключения видов требований, выделяемых в теории (например, требований к времени хранения данных или к доступности системы).

Требования к *безопасности приложения* описаны в табл. 1.

Таблица 1. Требования к безопасности

Код	**Описание требования**
SE-1	Учетные данные пользователей должны храниться в зашифрованном виде.
SE-2	Приложение должно обеспечивать защиту от выполнения стороннего кода (DB Injection, XSS, CSRF).
SE-3	Приложение должно производить проверку всех данных, поступающих от пользователя.
SE-4	Личные данные пользователя должны быть доступны для чтения и редактирования только ему.

Надежность приложения зависит от конфигураций сервера, исполняющего серверный код приложения.

Требования к *удобству использования* представлены в табл. 2.

Требования к *производительности приложения* представлены в табл.3.

Таблица 2. Требования к удобству использования

Код	Описание требования
US-1	Для авторизованного пользователя на каждой странице веб-приложения должно присутствовать меню, которое позволяет попасть к каждому из функциональных разделов веб-приложения не более, чем за 3 перехода.
US-2	Для мотивации пользователей к достижению своих целей в системе должны быть использованы элементы игрофикации.

Таблица 3. Требования к производительности

Код	Описание требования
PE-1	Максимальное время полной загрузки страницы, которая является результатом запроса пользователя к серверу, при использовании Интернет-подключения со скоростью 3Мбит/сек - 4 секунды.
PE-2	Максимальное количество одновременных подключений к веб-приложению - 1000.
PE-3	Необходимо обеспечить кросплатформенную работу приложения. Поддерживаемые ОС - Windows (2000 и выше), Linux, Mac OS.
PE-4	Необходимо обеспечить адекватную работу приложения в следующих версиях браузеров: Google Chrome версии 15.0.874 и выше, Mozilla Firefox 8.0 и выше, Opera 12.00 и выше, Safari 5.1.1 и выше.

Ограничения. Апаратное обеспечение состоит из сервера, с установленной операционной системой FreeBSD, который удовлетворяет следующим требованиям:

1. Мощность центрального процессора - 2 GHz.
2. Объем оперативной памяти - 2 GB.
3. Свободное место на диске, выделенное для кода и файлов приложения – 10GB.

Компьютер пользователя веб-приложения должен отвечать следующим требованиям:

1. Наличие установленой одной из операционных систем Windows (2000 и выше), Linux, Mac OS;
2. Наличие одного из браузеров: Google Chrome версии 15.0.874 и выше, Mozilla Firefox 8.0 и выше, Opera 12.00 и выше, Safari 5.1.1 и выше;
3. Наличие подключения к Интернет.

Модульность. Компоненты приложения необходимо реализовать в виде подключаемых модулей.

Остальные атрибуты качества в статье не определены, т.к. они не являются существенными для эффективной работы ПО.

Выводы

В данной статье сформулированы требования к ПО для поддержки контроля достижения целей, соответствующие двум верхним уровням модели К.Вигерса. Наличие этих данных существенно упрощает последующую разработку системы, поскольку система, которую разработают программисты, будет соответствовать потребностям пользователей.

Требования, описанные в статье, могут быть основой для построения требований третьего уровня модели Вигерса, которые, в свою очередь, лягут в основу разработки SRS, определения структурно-алгоритмической организации ПО, а также рекомендаций по его тестированию.

Список литературы

1. Meyer P. Attitude Is Everything: If You Want to Succeed Above and Beyond. Waco : Meyer Resource Group, 2003, p. 26.

2. ISO IEC TR 19759 - 2005, Software Engineering - Guide to the Software Engineering Body of Knowledge (SWEBOK). URL: http://materjalid.tmk.edu.ee/heikki_eljas/SWEBOK.pdf (дата обращения: 22.01.2013).

3. Леффингуэлл Д., Уидриг Д. Принципы работы с требованиями к программному обеспечению. Унифицированный подход. М.: Вильямс, 2002. 448 с.

4. Вигерс К. Разработка требований к программному обеспечению, М. : Русская редакция, 2004. 576 с.

5. Cunningham C., Zichermann G. Gamification by Design: Implementing Game Mechanics in Web and Mobile Apps. Sebastopol, California : O'Reilly Media, 2012, p. 288.

6. Каковський В.О. До питання вибору програмного забезпечення для підтримки контролю досягнення цілей // Міжнародна науково-практична конференція аспірантів і студентів «Інженерія програмного забезпечення 2012»: Зб. тез доповідей. - К., 2012. С. 15.

7. Lifetick. Online goal setting software. URL: http://lifetick.com (дата обращения: 22.01.2013).

Программирование голосом, как необходимость

Магистрант 2 курса Ильский А. А.
МИЭМ, НИУ ВШЭ, кафедра ИТАС

Аннотация. Тема работы предложена жизнью и связана с разработкой программного обеспечения для программирования голосом, для специализированной аудитории пользователей.

Данный проект ставит перед собой две цели: первая, разработка программного обеспечения компонентного типа для разработки компьютерного кода с применением голосового интерфейса, с совместимостью с любым пакетом программного обеспечения распознавания речи, для основных и мобильных платформ; второй целью является оценка возможности улучшения задачи программирования, а также функционала программного обеспечения, с голосовым интерфейсом распознавания речи, предоставляемым для этих целей.

В проекте не ставится целью разработать, реализовать или улучшить в какой-либо форме существующее программное обеспечение распознавания речи, а только использовать функционал, который им предоставляется, для выполнения специальных задач. Следовательно, точность распознавания речи зависит от программного обеспечения распознавания речи, а не программного обеспечения разрабатываемого в данном проекте.

Программное обеспечение распознавания речи сконцентрировано на диктовке, позволяя печатать документы, электронные письма, помогая специальностям, где работа связанна с длительными периодами печати, с повышенным риском получения заболевания и травм опорно-двигательной и нервной системы. Ряд профессий таких как журналисты, программисты, писатели и ученые, в условиях сжатых сроков продолжают работу несмотря на рекомендации и останавливаются только когда утомлены и в особенности при боли или дискомфорте,.Что в последствии проявляется в тендините, сильных шейных и спинных болях и приводят к потере самооценки, снижению качества жизни, ухудшению семейных отношений.

В настоящее время от Repetitive Stress Injury (RSI) или Repetitive Motion Injury и Синдром Канала Запястья (Carpal Tunnel Syndrome) страдают огромное количество 80% из них пользуются клавиатурой в течение длительных периодов времени ежедневно. В странах Европы лечение обходится индустрии не менее чем 3 миллиардов евро ежегодно. Но обычно люди не сообщают о заболевании RSI, т. к. это боятся, что скажется на их работе. По оценкам только в 40% случаев сообщаются/обращаются за помощью, что делает проблему еще серьезнее. Хотя RSI наиболее известная причина невозможности использовать клавиатуру, существует еще множество состояний влияющих на возможность пользоваться компьютером, на пример, ампутация рук или кистей, припадки, врожденные заболевания. Чтобы продолжить работу необходимо сменить метод ввода, например, на распознавание речи.

Круг заинтересованных лиц расширяется, при использовании мобильных платформ. Быстро растущая отрасль мобильных платформ и приложений IT индустрии позволит выполнять часть работ вне офиса и здоровым специалистам.

Использование обычных программ распознавания речи затруднено тем, что хотя и ключевые слова могут быть распознаны, но большая часть кодовых текстов состоит из имен переменных, названий процедур, которые представлены словосочетаниями

и/или аббревиатурами. Также в языках программирования использование символов отношения, равенства, объявления переменных и работы с параметрами синтаксически отличается от естественных языков. В результате требуется как минимум адаптированная или специализированная, а может и в целом иная по типу система для распознавания в программировании.

Программирование голосом позволит людям с ограниченными возможностями открыть себе двери в огромную индустрию в противном случае недоступную для них, а также программистам, которые в связи с травмами более не способны печатать, чтобы продолжать свою работу.

В настоящее время мало инструментов для программирования голосом и большинство из них продукты текущих исследований, а не коммерческое программное обеспечение.

Программное обеспечение программирования голосом

VoiceCode - Open Source инициатива Института Информационной Технологии Национального Исследовательского Консульства Канады. Задачей является разработка инструментария совместимых компонентов поддерживающего текущие лучшие практики программирования голосом на всех основных платформах[1].

ShortTalk and EmacsListen — разработка специализированного разговорного языка для человека-компьютерного взаимодействия.[2]

Voice Grip[5] – дополнительный макрос для редактора Emacs созданный с целью упрощения использования коммерческого программного обеспечения распознавания речи программистами программистам.

Java by voice[6] — серия макросов для редактора Emacs спроектированные для упрощенного ввода кода на языке Java

Cache Pad – макрос для редактора Emacs для кэширования недавно продиктованных имен функций и переменных для последующего переиспользования.[3]

Emacs VR Mode - макрос для редактора Emacs добавляющий особенность Select and Say в редактор из ПО Dragon Naturally Speaking.[4]

Программирование голосом - область находящаяся еще в стадии исследования и конечно не применяемая большим числом программистов. Достаточно исследований было проведено доказывающих возможность программирования голосом, но пока не существует программного обеспечения специально спроектированного для этой задачи. Существующие проекты это демонстрируют.

Альтернативный подход был предложен Adam Janin[7] состоял в разработке нового «speech friendly» языка, но его исследование продвинулось дальше обсуждения особенностей такого языка т. к. его RSI пошло на поправку

Задача программирования

Программирование как задача можно разбить на 4 области:

- Написание
- Отладка
- Компиляция
- Исполнение

Программист обычно пишет небольшую секцию для решения конкретной проблемы. Затем эта часть кода компилируется, если компиляция успешна , запускается и тестируется с применением инструментов отладки. Когда код

исполняется правильно — он расширяется для выполнения большей задачи. Процесс повторяется до того как код решает всю поставленную задачу.

Таким образом за написанием следует исправление ошибок, тестирование и отладка. Программист обычно работает в среде, которая предоставляет инструменты для выполнения всех четырех функций, на пример, Microsoft Visual Studio.

Написание кода

В свою очередь обычно делится на последующие задачи (зависит от языка программирования)

[1] Определение имени класса
[2] Определение используемых функций
[3] Определение переменных используемых в функциях
[4] Манипуляция переменными с использованием вызова других функций или специфических конструкций for и while циклов или if и switch конструкций.

Если одно и тоже задание дать группе программистов результат будет разный, потому что у программистов есть личный стиль.

Несколько вещей определяющих стиль программиста:

1. Способ записи и именования переменных, классов и функций
2. Структура и группирование кода
3. Положение скобок
4. Отступы в коде
5. Использование комментариев

Иерархический анализ задачи.

Последующий иерархический анализ показывает различные шаги, которые пользователь выполняет на каждой стадии написания кода.

Основные стадии написания кода:

7. Открытые файл
8. Редактирование файла
9. сохранение файла
10. Компилирование файла
11. Выход

Каждая стадия задачи будет раскрыта далее. Анализ проводится для объектно-ориентированных языков, таких как Java или C++, но основные задачи и их субзадачи будут одинаковы для любого используемого языка программирования.

Все задачи ниже второго уровня содержат все себе процесс печатания или диктовки в редакторе, а для задачах ниже уровнем редакторы предоставляют соответствующие инструменты.

Задачи выше второго уровня обобщенно больше представляют собой мыслительный процесс, чем реально исполняемые действия.

Где возможно задачи ниже четвертого уровня необходимо автоматизировать.

Открытие файла:

1. Выбор Открыть файл опции
2. Ввод имени открываемого файлы или
3. Щелчок курсором по имени файла в отображаемом списке

Редактирование Файла:

1. Уровень 1:

- Подключение необходимых библиотек

- Определение нового класса
- Определение новой функции/метода

2. Уровень 2:

- Ввод ‘#include’
- Ввод имени используемой библиотеки
- Ввод ключевого слова ‘class’
- Ввод имени класса
- Определение области класса
- Определение классовых переменных
- Определение конструктора класса
- Определение области функции
- Определение типа результата функции
- Определение входных параметров функции
- Определение тела функции
- Определение новой переменной
- Присвоение значение переменной
- Использование функции

Уровень 3:

- Определение наследуемого/расширяемого класса
- Ввод директивы наследования/расширения
- Ввод имени родительского/супер класса
- Ввод пар (тип/класс, имя) параметров
- Определение области переменной
- Определение типа переменной
- Определение имени переменной
- Определение параметров конструктора
- Определение тела конструктора
- Определение используемой функции/метода
- Определение параметров используемой функции/метода
- Определение используемых функцией переменных
- Определение переменной для присвоения
- Присвоение значений используемым функцией переменным
- Присвоение значения
- Использование ‘if ‘ оператора
- Использование ‘else’ оператора
- Использование ‘for’ оператора
- Использование ‘while’ оператора
- Использование ‘do.. while’ оператора

Уровень 4:

- Ввод ключевых слов ‘if ‘, ‘else’, ‘for’, ‘while’, ‘do.. while’
- Инициализация начальных значений переменных/итераторов
- Задание условия сравнения с переменной/константой или остановки цикла
- Постановка открывающих/закрывающих скобок
- Постановка символьных операторов языка
- Постановка точки с запятой

Сохранение файла:

- Выбор Сохранить файл опции
- Ввод имени, с которым сохраняется файл или
- Выбор Сохранить опции (если файл был предварительно сохранен)

Закрытие файла

- Выбор Закрыть файл опции
- Если запрошено сохранение изменений выбор между опциями Да или Нет

Закрытие Приложения

- Выбор Выйти из приложения опции

Ниже третьего уровня действия синтаксически ориентированы (зависят от языка). Это там, где диктовка синтаксиса наиболее затруднена, и где происходит наибольшее количество опечаток. Если бы синтаксис языков содержал бы меньше скобок и позволял пробелы после пунктуации, тогда бы программирование голосом было проще.

Сложности при программировании речью

Все задачи связанные с написанием кода сложны, когда проговариваются. Трудно выполнимы или утомительны, доводя пользователя, до отказа от использования речи.

Определение имен переменных, классов и функций.

Если бы все имена переменных и классов были одиночным английским словом, тогда это не было бы проблематичным. В реальности это крайне редко, имена должны отражать исполняемую функцию или содержимое переменной. Компиляторы требуют чтобы имена переменных не содержали пробелов, так что программистам приходится находить различные методы концентрации слов, сохраняя простоту чтения. Проблема заключается в том что программное обеспечение распознавания речи спроектировано для задач диктовки текстов и автоматически добавляет пробел между словами. Например, продиктовать «CamelCase» или «under_score» используя Dragon Naturally Speaking потребуется произнести:

«came, no space, capitalise next, case» или «under, no space, underscore, no space, score»

Имена обычно длиннее, чем два слова, и чем длиннее имя, тем более значительно замедляется диктовка.

Написание конструкций for и while.

Все конструкции циклов и «if» утверждение обычно определяется наличием логического условия в скобках, при котором секция кода исполняется. Именно скобками затрудняется диктовка и использование этих конструкций становится затруднительным.

Пример кода:

```
for(counter=0;counter<10;counter++)
{
printf(counter);
|}
```

Потребует произнести: «For, no space, open brackets counter equals−sign zero. semi−colon. counter less−than ten semi−colon. counter plus−sign plus−sign .close brackets. new line. open braces. print. no space. open brackets counter. close brackets. semi−colon. new line. cose braces».

Навигация

Эта проблема распространяется на все что связано с использованием речи. Мышь — великолепный инструмент для быстрой навигации в документах, т. к. позволяет перемещать курсор, «прокручивать» документ вверх и вниз в поисках необходимой секции. Но этот недоступен тем кто не может пользоваться своими руками. Типовыми решениями программного обеспечения распознавания речи являются переходы в начало или конец предложения, или абзаца, или номеру строки, или на определенное количество строк, но для программирования голосом необходимы более специализированные решения.

Использование опций меню

Из иерархического анализа видно, что множество задач требует использования опций меню (открытие, закрытие, сохранение файлов). Что легко реализуется мышью, но трудно когда при помощи одного голоса. Традиционно реализуется индикаций своих намерений воспользоваться опциями меню при помощи голосовой команды, на что в ответ программное обеспечение переходит режим «управления». Задача программирования потребует реализовать дополнительные режимы, диктуемые из потребности задачи программирования. Но основным ограничением является то, что обычно интерфейсы графически ориентированы и используют, флаг элементы, списки и другие диалоговые элементы. Затруднительное использование списков и диалоговых элементов возможно, но использование флаг элементов на первый взгляд кажется невозможным.

Список литературы

[1] http://sourceforge.net/apps/mediawiki/voicecode/index.php?title=VoiceCode/Programming_by_voice_ressources
[2] http://shorttalk-emacs.sourceforge.net/ShortTalk/index.html
[3] http://www.voicerecognition.org/developers/jepstein/
[4] http://emacs-vr-mode.sourceforge.net/
[5] Desilets, Alain. «Voice Code, Programming by Voice Toolkit».
[6] Epstein, Jonathan. «Writing and Debugging Code by Voice»
[7] Janin,Adam «Design of a Speech Orientated Programming Language».

Проектирование интерфейса и модель пользователя

Иванов С.Ю., доц., к.т.н.
МИЭМ НИУ ВШЭ, каф. ИТАС

В настоящее время создание новых интерфейсов является основным источником доходов для разработчиков программного обеспечения. Новые интерфейсы и обновления выходят постоянно и пользователя не спрашивают, нужны ли ему эти изменения. Интерфейсы обычно имеют жесткую конструкцию и работают в режиме «меню». Новые функции, часто не используемые, новое расположение кнопок, изменения в логике реализаций операций часто снижает популярность программного продукта. В первую очередь это относится к программным продуктам в интернете. Так популярность портала «Одноклассники» значительно снизилась за счет постоянных изменений в интерфейсе, введении множества мало востребованных функций.

Алан Купер в своей книге, посвященной разработке интерфейсов четко показывает, что в массе своей программы создаются для программистов: на основе их догадок о потребностях пользователя. «Неудобные, сложные продукты окружают нас, и этот круг становится все более тесным …» [1] Часто пользователям не нужно такое количество функций для решения своих задач. Более того усложнение интерфейсов ведет к повышению вероятности возникновения ошибок, которые потом списывают на «человеческий фактор».

У пользователя формируется набор представлений и правил, которые, по его мнению, должны обеспечивать взаимодействие с программной системой. Эту совокупность можно рассматривать как модель пользователя. Если пользователь уже работал с подобной программой, он будет думать, что и эта программа будет работать точно также.

У программной системы тоже есть своя модель. Эта модель является продуктом тех людей, которые разработали программный продукт, и людей, разработавших саму вычислительную систему. То есть интерфейс разрабатывается в соответствии с требованиями пользователя, но по правилам и стандартам разработчиков программного продукта, которые подчиняются правилам и стандартам разработчиков системного программного обеспечения. Часто интересы разработчика расходятся с интересами пользователя. Пользователь хочет дешевый, удобный и простой продукт, программист хочет реализовать все функции, указанные в техническом задании, а запросы пользователя для него остаются на втором плане.

В итоге мы имеем конфликт между моделью и программной моделью, а источником проблем является пользовательский интерфейс.

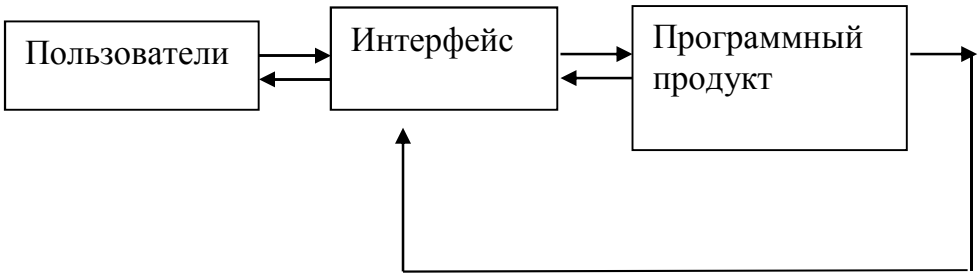


Рис.1 Традиционная схема организации диалога

При традиционном подходе основное внимание разработчиков уделяется реализации необходимых функций программного продукта и избавлению от ошибок при его работе. Интерфейсной части уделяется внимание в основном с точки зрения визуальной привлекательности. Руководители проектов не жалеют средств на дизайнеров и создателей различных визуальных эффектов. Обычно версия интерфейса программы рассматривается с точки зрения целесообразности. Если есть время (обычно его нет), то несколько человек выполнят на ней задания. В это время разработчик должен наблюдать за их действиями, цель - выяснить, чего они ожидают от программы[2]. Наиболее успешные разработчики программных продуктов формируют модели пользователя. С помощью таких моделей они адаптируют интерфейсы к определенным группам пользователей, пытаются понизить вероятность ошибок пользователя и исключить аварийные ситуации. При анализе действий пользователя необходимо учитывать психологические особенности различных групп пользователей, даже менталитет пользователей из различных стран и регионов. Авторы исследований, проведенных в рамках российско-белорусского проекта АИСТ, формирование комплексной модели пользователя для систем дистанционного обучения, отмечают, что существует 2 основных способа построения модели пользователя в интеллектуальных системах. Формирование модели в процессе наблюдений за взаимодействием пользователя с системой, а также построение модели в процессе тестирования пользователя.[3] Здесь же приводятся примеры использования ряда психологических тестов. Модели пользователя становятся сложными и многофакторными. За обилием рассматриваемой информации теряется основная цель – дать пользователю удобный интерфейс, а программисту удобную и понятную модель пользователя.

В данной статье предлагается использовать некоторую простую модель пользователя и модель программы, основанные на принципах систем имитационного моделирования. Предлагается использовать простую систему имитационного моделирования, ориентированную на события. В качестве такой системы берется собственная реализация системы моделирования, принципы работы которой похожи на принципы работы системы SIMPL. Наиболее полезными функциями системы являются функции планирования и выполнения событий, а так же сбор статистики. Планирование и выполнение событий соответствуют действиям пользователя при работе интерфейса, а средства ведения статистики позволяют запротоколировать все действия пользователя для дальнейшего анализа. Такой подход может служить основой для создания автоматизированных средств по формированию модели пользователя и имитатору работы программного продукта. Основной проблемой здесь является синхронизация имитационной модели, которая работает в модельном времени, и реального объекта, который работает в реальном времени. Такую синхронизацию можно реализовать за счет использования принципов, изложенных в работе [4].

Рассмотрим пример реализации интерфейса для реализации достаточно простого взаимодействия пользователя с программной системой, обеспечивающей биржевые операции. Существуют различные реализации электронной биржевой торговли. Они достаточно сложны и основное внимание в них уделяется наиболее важной проблеме – защите информации. Предлагается максимально простая система для пользователя, которая в автономном режиме позволяет реализовывать важные для него операции и получать только нужную ему информацию и необходимую статистику. Программа

использует сразу два режима диалога: заполнение бланка и ввода сообщения на ограниченном естественном языке. Предполагается, что упрощение системы приведет к уменьшению ошибок пользователя и формированию им обоснованных решений. Разделение основной системы электронной биржевой торговли дает пользователю дополнительное время и средства для оценки своих предполагаемых действий на бирже. Соревноваться в скорости проведения биржевых операций с профессиональными брокерами, а тем более с программами роботами не имеет смысла.

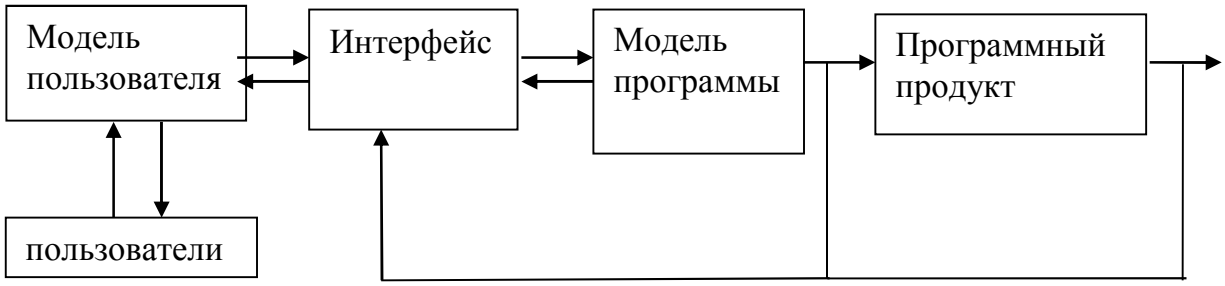


Рис.2 Схема организации диалога с использованием модели пользователя и модели программы

Пример взаимодействия пользователя с программой, ориентированной на анализ котировок и проведение операций на бирже драгоценных металлов. Пользователь задает ряд команд, связанных с покупкой и продажей металлов. Важным критерием для диалога в программе, который реализуется в режиме «бланк», является надежность и осмысленность при проведении операций. Существует возможность потерять деньги при необдуманном поведении пользователя, при ошибочном использовании интерфейса. Так как основное внимание пользователя привлечено к операциям на бирже, то интерфейс должен быть максимально простым в использовании. На самом деле основных операций всего две, продать или купить металл. Однако вариантов входного предложения на языке пользователя больше. Например, пользователь хочет купить 100 грамм золота. Возможные фразы: «купить 100 грамм золота», «100 грамм золота купить», «купить золота 100 г», «очень хочу купить золота 100 г» и т.п. Программа должна однозначно и безусловно понять приказ пользователя. Для этого в программе была использована фреймовая структура в виде сети. Причем анализируются во входной фразе только те слоты, которые соответствуют значащим для проведения операции словам. Именно они используются для анализа заданной фреймовой структуры. Остальные слова игнорируются программой, а пользователю дается возможность в удобном для него виде задавать команду. Для данного примера ядром фреймовой структуры будет сеть следующего вида (см. рис. 3).

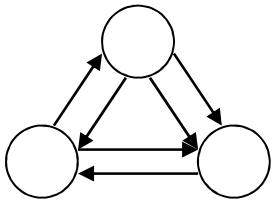


Рис.3 Ядро фреймовой структуры

Данный граф отображается матрицей «вершина - потомки» и массивом, где вершинам соответствуют определенные словари. Числовое значение не включается в структуру и обрабатывается отдельно. Словари могут быть вынесены из программы и модифицироваться без изменения программного кода.

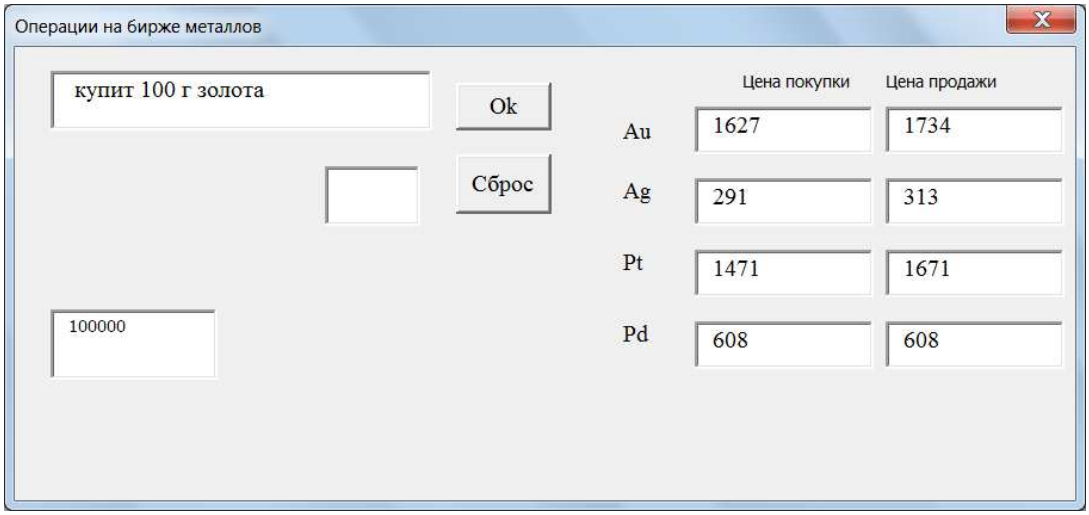


Рис.4 Пример интерфейса

При таком представлении фразы анализируются только значащие 3 слова. Причем фразу можно дополнять другими словами. Слова можно использовать в любом порядке, переставлять. Допускается возможная модификация слова. Например, вместо слова «грамм» может использоваться слово «грамм» или сокращение «г». Пользователь очень любит сокращения и не любит ограничения. Такую простую программу, с небольшим количеством реакций системы обычно реализуют с помощью жесткого аппарата меню, который заставляет его работать по жестким правилам. Однако сеть фреймов дает большие возможности для расширения системы. Особенно хорошо этот подход будет работать при распознавании голосовых команд. Но самое главное, уменьшается количество ограничений и безусловных правил для пользователя.

Список литературы

1. Купер А. // Психбольница в руках пациентов. Символ-Плюс. ISBN 978-5-93286-168-4, 0-672-31649-8; 2009 г.

2. Joel Spolsky. User Interface Design for Programmers, Apress, 2001. ISBN 1-893115-94-1

3. Gavrilova T., Vasilyeva E. One Approach to Individualised Interface Design // Proceedings of X-th Conference «Knowledge-Dialogue-Solution», Varna, Bulgaria, 2003. P. 221–226. (in English).

4. Иванов С.Ю. Синхронизация программных подсистем и реальных объектов при натурном и полунатурном моделировании. //Научные материалы Первой международной научно-технической конференции «Аэрокосмические технологии», МГТУ им. Р.Э. Баумана, Реутов, НПО«Машиностроение», 2004г.,с.186-188

Алгоритм равномерного распределения вычислительной нагрузки на кластере методом деревьев нитей

Гусев А.П.
Студент МГТУ им. Н.Э. Баумана,
кафедра «Компьютерные системы и сети»
Научный руководитель: Руденко Ю.М.,
к.т.н., доцент кафедры «Компьютерные системы и сети»
xitren@ya.ru

Аннотация. Представлен алгоритм распределения вычислительной нагрузки на кластере. Для разработки системы распределения нагрузки предстоит разрешить две основные проблемы. Во-первых, разработать алгоритм для анализа структуры сети и вычислительной мощности узлов. Во-вторых, разработать алгоритм для анализа и разделения программы на блоки вычислений в соответствии с используемыми ресурсами и необходимой вычислительной мощностью. Далее, в Разделе 1 рассматривается структура программы использующей параллельные вычисления. Раздел 2 рассматриваются существующие системы поддержки параллельного программирования. Раздел 3 содержит подробный анализ существующих алгоритмов распределения задач. Раздел 4 представляет описание разработанного алгоритма распределения.

Ключевые слова: вычислительная нагрузка, кластер, дерево нитей, алгоритм распределения.

Оптимизация исполняемой программы

Использование параллельных вычислений позволяет значительно ускорить вычисления, задействуя все имеющиеся ресурсы, однако встает вопрос о том, насколько это использование полномерно. Оптимизация выполняемых вычислений является одной из наиболее сложных задач в параллельном программировании. Проанализируем структуру выполняемой параллельной программы:

- Загрузка исполняемого приложения в оперативную память,
- Определение блока входных данных,
- Чтение входных данных,
- Непосредственно вычисления,
- Формирование и вывод результатов.

Производительность программы увеличивается при снижении времени исполнения любого из вышеперечисленных действий.

Как правило, загрузка исполняемого приложения происходит довольно быстро, так как его размер колеблется в пределах 20 Мб в самых сложных случаях, тогда как массив данных для вычислений может быть ограничен по размеру лишь вместительностью устройств хранения данных (несколькими терабайтами, например).

Нашей задачей на данном этапе является разработка системы позволяющей распределить запросы к данным и блоки вычислений таким образом, чтобы уменьшить количество пересечений запросов к данным.

Анализ планировщиков задач

Первое, что необходимо при работе с распределенными кластерными системами – это их интеграция в единый многопроцессорный комплекс, который используется пользователями кластерной системы, для размещения задач в общей очереди, их мониторинга и модификации.

Именно для этой цели были разработаны такие программные продукты как: Менеджеры Ресурсов (Resource Manager), Кластерные Системы Управления (Cluster Management System) или Системы управления пакетной обработкой (СПО). Известны более двадцати различных СПО. такие как PBS, SGE и Condor, а также коммерческие системы LoadLeveler и LSF.

Все СПО имеют примерно одинаковые наборы функций, но отличаются по отдельным характеристикам. Так, системы PBS и SGE примерно равны по своим возможностям и предполагают полное выделение машин под пакетную обработку. Ими поддерживаются сетевые кластеры с однопроцессорными машинами, а также мультипроцессоры SMP и MPP. Отдельные процессоры могут использоваться в монопольном режиме (только одна задача на процессор) или в режиме разделения времени (несколько заданий могут разделять процессор). В последнем варианте при распределении заданий контролируется уровень процессорной загрузки.

Рассмотрим несколько планировщиков задач более подробно, для выявления их положительных качеств.

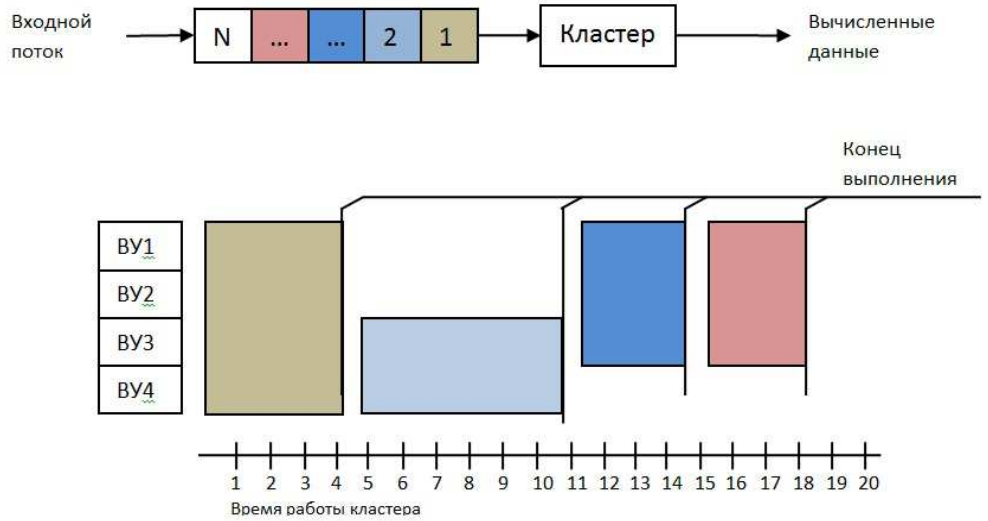


Рис.1. Схема алгоритма FCFS и график распределения задач

Бесплатно распространяемая система планировки задач Portable Batch System (PBS), имеет в своей основе технологию First-Come-First-Served (FCFS), схема которого представлена на рисунке 5, и технологию Shortest-Job-First (SJF), пример временной диаграммы выполнения задач этого алгоритма представлен на рисунке 6. Может использоваться совместно с системой управления заданиями Torque и планировщиком задач MAUI. Как и в других СПО планировщик PBS позволяет выравнивать нагрузку узлов, опираясь на текущие, полученные от операционной системы, значения (LoadLeveling). Такой планировщик имеет наклон в сторону распределения ресурсов по текущему состоянию системы и поддерживает систему приоритезации групп пользователей.

Другим представителем является свободно распространяемая система Sun Grid Engine (SGE). SGE имеет более оптимизированный планировщик чем система OpenPBS и лучше справляется с загрузкой сети большим потоком малых задач используя алгоритм FCFS и дополнительные алгоритмы приоритетов и поддержку миграции задачи. В отличие от PBS эта система не имеет развитого API для запуска и управления процессами, именно по этой причине она редко применяется с MPI

приложениями. SGE включает в себя модуль для определения политики разделения ресурсов между независимо работающими пользователями.

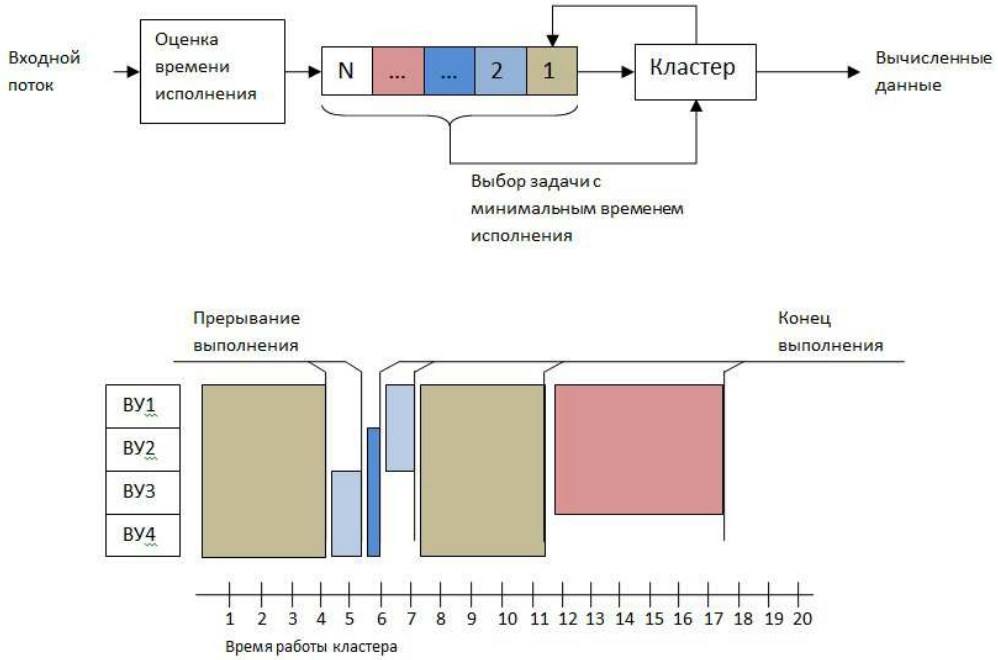


Рис. 2. Схема алгоритма SJF и график распределения задач

Продукт компании IBM – LoadLeveler, предназначенный для пакетной обработки последовательных и параллельных заданий на кластерах. В своей основе имеет алгоритмы FCFS и backfilling (схема которого представлена на рисунке 7). Последние версии этой системы поддерживают множество дополнительных механизмов контрольных точек, улучшенное восстановление буфера заданий, механизмы взаимодействия, для улучшения масштабируемости и производительности при запуске заданий. Данная система также поддерживает дополнительное планирование выполнения нескольких шагов задания, устанавливая минимальные ограничения на тип и число необходимых ресурсов.

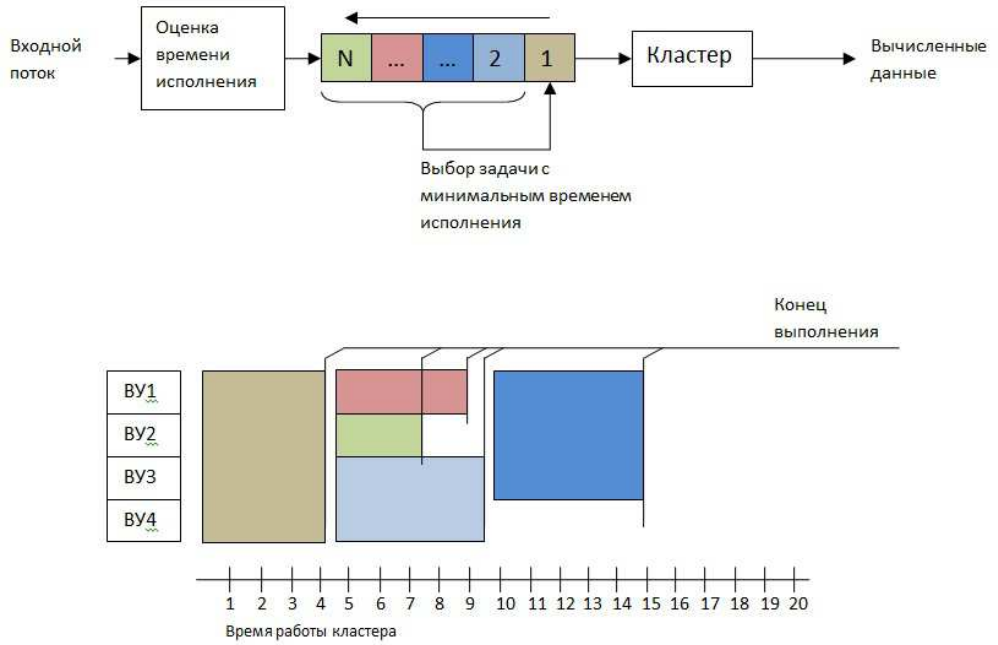


Рис. 3. Схема алгоритма Backfilling и график распределения задач

Коммерческая кластерная система LSF (Load Sharing Facility) компании Platform Computing Corporation для управления пакетной обработкой, представляет собой полномасштабную систему, обеспечивающую все существенные требования к СПО: алгоритмы планирования FCFS, fair-share, preemptive (схема представлена на рисунке 7), backfilling, поддержку очередей заданий, сбор информации о наличии и занятости ресурсов в кластере, нахождение компьютеров с подходящими ресурсами для выполнения заданий, поддержку режима контрольных точек (checkpointing), миграцию заданий и др.

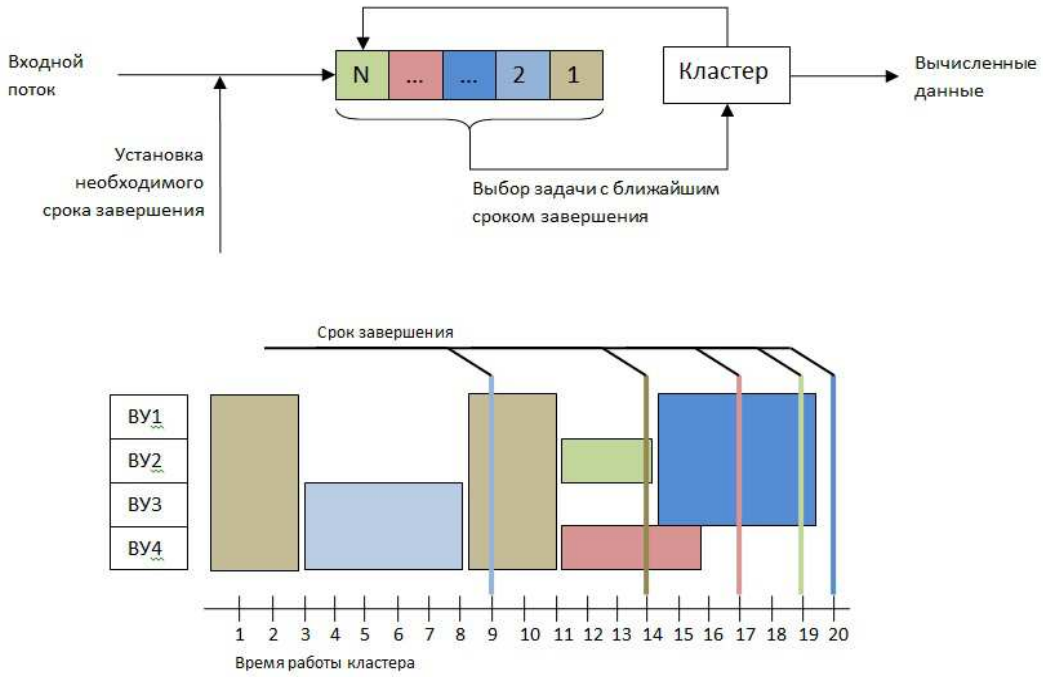


Рис. 4. Схема алгоритма EDF и график распределения задач

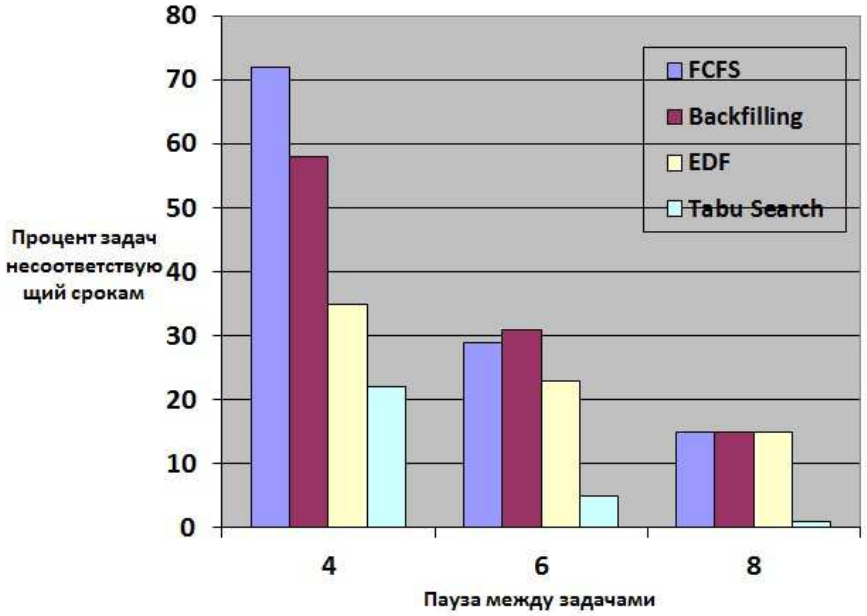


Рис. 6. Процент задач несоответствующие срокам при использовании различных алгоритмов планирования

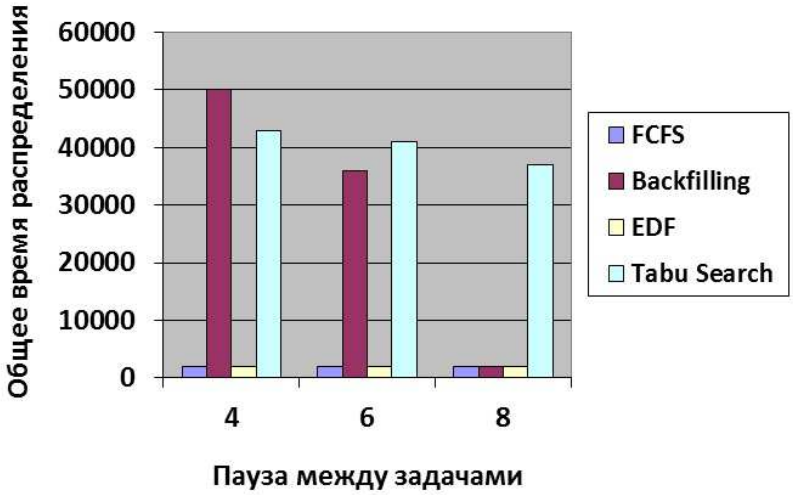


Рис. 7. Время необходимое для планирования каждым методом

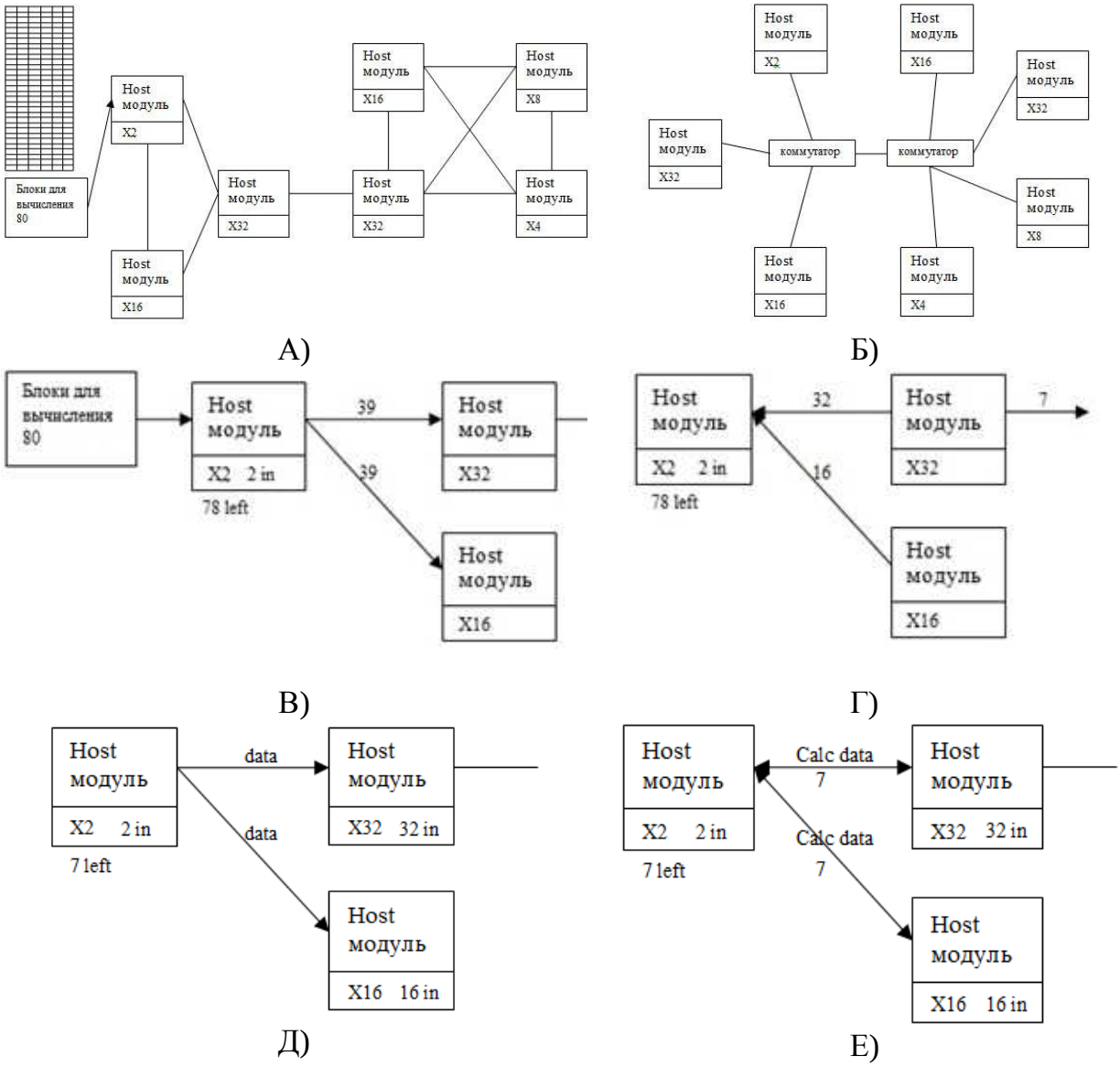


Рис.12. Этапы работы предложенного алгоритма распределения

Внешний планировщик MAUI , который может использоваться взамен штатных планировщиков для нескольких СПО: PBS, SGE, Loadleveler, LSF, Wiki. MAUI - открытый продукт, который отличается большим набором режимов (политик) планирования и наличием механизма предварительного резервирования (Backfilling). MAUI представляет большой интерес в связи с тем, что это единственная из свободно распространяемых СПО, способная обеспечивать автоматический запуск многопроцессорных заданий, избегая при этом неоправданного простоя ресурсов.

Также важно отметить, что совершенствование планировщиков задач происходит постоянно, в связи с чем, за последнее время было разработано множество новых методов планирования, таких как: Earliest Deadline First (EDF), Easy Backfilling, Minimum Tardiness Earliest Deadline First, Tabu Search.

Опираясь на данные исследований ERCIM, составим таблицы эффективности рассмотренных методов, характеризующих соответствие срокам выполнение задачи кластера (Рисунок 10) и времени планирования (Рисунок 11).

Предлагаемый алгоритм распределения вычислительных задач с использованием модульной системы и дерева нитей

Для улучшения распределения задач на кластере, предложен алгоритм с использованием асинхронного распределения нитей вычислений между узлами вычислительной сети для организации процесса взаимодействия вычислительных узлов.

Для примера функционирования предложенного алгоритма возьмем неоднородную вычислительную систему, представленную на рисунке 12.Б.

Модули сканируют определенный конфигурацией порт каждого компьютера в сети, и в случае обнаружения аналогичного модуля, сохраняют его в карте сети с средним временем полученным с помощью команды ping и значении скорости передачи данных, запуская соответствующий тест.

Следующим шагом каждого модуля является минимизация полученного дерева сети, с определением коммутационных групп и реконфигурации сети (Рис. 12.А), оставляя взаимодействие между группами узлам соответствующим условию:

max a*<производительность>i+b*<скорость сети>i+c*<интервал *ping*>i,

где a,b,c – коэффициенты задаваемые конфигурацией.

При поступлении на вход блока данных (80 блоков) *host*-модули начинают процедуру согласования распределения задач:

1. Резервирование блоков для самостоятельного вычисления
2. Рассылка запроса на вычисление оставшихся блоков по таблице вычислителей через промежутки времени (Рис 12.В).
3. Пересылка запрошенным вычислительным модулем согласия на вычисление, при наличии свободных мощностей и запроса на вычисление следующим вычислительным узлам (Рис 12.Г).
4. Получение согласия на вычисление и передача входных данных (Рис 12.Д).
5. Вычисление данных починенным узлом.
6. Пересылка вычисленных данных узлу-источнику, либо назначенному узлу (Рис 12.Е).
7. Повторение операций.

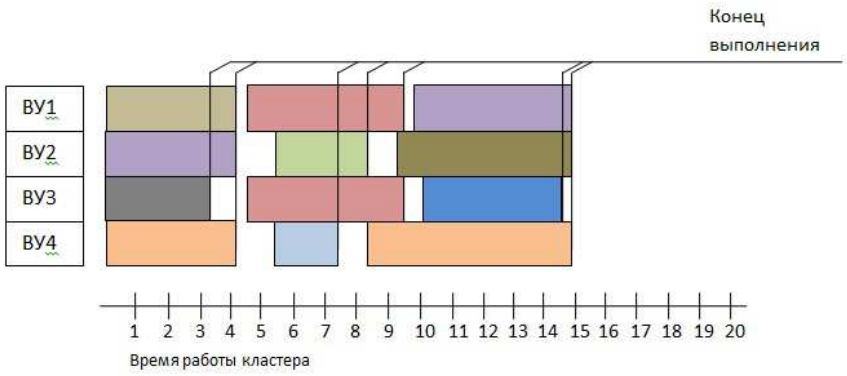


Рис. 9. Диаграмма нагрузки

Как видно из приведенной на рисунке 5 диаграммы, падение производительности в данном случае не превышает 11%, которые теряются на задержки перед запуском процесса для передачи информации по сети.

Литература

1. B. Wilkinson and M. Allen. Parallel Programming Techniques & Applications Using Networked Workstations & Parallel Computers, 2nd ed. Toronto, Canada: Pearson, 2004.
2. F. Gebali. Analysis of Computer and Communication Networks. New York: Springer, 2008.
3. T.G. Lewis and H. El - Rewini. Introduction to Parallel Computing. Englewood Cliffs, NJ: Prentice Hall, 1992.
4. B. Burke. NVIDIA CUDA technology dramatically advances the pace of scientific research. http://www.nvidia.com/object/io_1229516081227.html?_templated=320, 23.09.2012.
5. Cilk Arts. Smooth path to multicores. http://www.cilk.com/, 26.09.2012.
6. OpenMP. OpenMP: The OpenMP API specification for parallel programming. http://openmp.org/wp/, 2009.
7. C.E. Leiserson. The Cilk + + Concurrency Platform. Journal of Supercomputing, 51 (3), 2009.
8. MIP Forum. Message passing interface forum. http://www.mpi - forum.org/, 6.09.2012.
9. R.D. Blumofe and C.E. Leiserson. Scheduling multithreaded computations by work stealing. Journal of theACM (JACM), 46 (5), 1999.
10. OpenMP. Summary of OpenMP 3.0 c/c + + syntax. http://openmp.org/mp - documents/OpenMP3.0 -SummarySpec.pdf, 23.09.2012.
11. Maui Administrator's Guide. http://www.adaptivecomputing.com/resources/docs/maui/pdf/mauiadmin.pdf, 2.10.2012.
12. TORQUE Administrator Manual. http://www.clusterresources.com/torquedocs21/, 3.10.2012.

«Машинная лингвистика»

Автоматическая классификация текстов на основе заголовка рубрики

О.В. Гуляев (*futujaos@gmail.com*)
Факультет информатики и систем управления МГТУ, Москва
Н.В. Лукашевич (*louk_nat@mail.ru*)
НИВЦ МГУ, Москва

1. Введение

Одной из востребованных технологий обработки текстовой информации является автоматическая классификация (рубрикация) текстов. Широко используемым подходом к автоматической классификации текстов являются подходы, основанные на машинном обучении [Sebastiani, 2002]. Однако для обеспечения хорошего качества работы таких методов требуется значительные объемы качественной обучающей текстовой коллекции, которая представляет собой совокупность текстов с вручную проставленными рубриками. Создание такой коллекции обычно требует достаточно много затрат, и поэтому часто в реальных задачах такая коллекция для обучения отсутствует [Агеев и др., 2008].

Вместе с тем названия рубрик в рубрикаторах обычно не случайны и достаточно хорошо передают предполагаемое основное содержание документов, которые должны быть к ним отнесены. Это позволяет развивать алгоритмы, которые учитывали бы соответствие между названием рубрики и текстами для построения обучающей выборки.

Кроме того, особенностью построения рубрикаторов является то, что в идеальном случае каждый текст обрабатываемой текстовой коллекции должен относиться хотя бы к одной рубрике рубрикатора [Bates, 1988]. Это позволяет предположить, что даже если схожесть некоторого текста с той или иной рубрикой не удалось установить при непосредственном сравнении текста документа с рубрикой, то полезным является выявление схожести между документами уже отнесенными к рубрике и оставшимися неразмеченными.

В данной статье предложен метод рубрикации текстов, основанный на построении векторов рубрик в пространстве терминов при помощи электронного тезауруса WordNet [Miller, 1998] и дальнейшем уточнении рубрик документов методом k ближайших соседей (англ. k-nearest neighbor algorithm, kNN). Метод рубрикации был протестирован на коллекции Reuters-10[1].

2. Методы классификации на основе заголовка рубрики

Задача классификации текстов на основе заголовка рубрики уже ставилась в ряде работ.

В работе [Gliozzo et al., 2005] предлагается метод, состоящий из двух шагов. На первом шаге создавалось начальное множество классифицированных документов с помощью метода без учителя, работающего на основе сокращения признакового пространства текстов (англ. Latent Semantic Indexing, LSI) и алгоритма Гауссовой

[1] 10 самых частых рубрик в коллекции Reuters-21578 - *earn, acquisitions, money, grain, crude, trade, interest, ship, wheat, corn*. Содержит 8493 документа.

смеси (англ. Gaussian Mixture algorithm). На втором шаге производилось обучение классификатора на начальном множестве документов.

В работе [Rodriguez et al., 1997] применяется комбинация обучающей коллекции и информация из WordNet как данных для алгоритмов Рошио (англ. Rocchio) и Видроу-Хоффа (англ. Widrow-Hoff), с помощью которых проводилась классификация.

Задача определения лексической связи между термином и текстом решается в работе [Glickman et al., 2006]. Данные из RTE-1[1] были использованы для составления набора лексических связей, состоящего из пар термин-предложение. Предполагалось, что термин и предложение из составленного набора лексически связаны друг с другом. Проводились эксперименты над различными моделями определения лексических связей. Лучшие результаты показала комбинированная модель, использующая WordNet для определения явных связей между терминами и наивный байесовский классификатор для определения скрытых связей между термином и предложением, получаемых из контекста.

Дальнейшее исследование метода классификации с помощью заголовка рубрики происходит в работе [Barak et al., 2009]. Мера сходства между рубрикой и документом определяется на основе контекстного сходства и связи между ними. Контекстное сходство определяется с помощью представления заголовка рубрики и документа как векторов в LSI пространстве. Для определения связи между документом и рубрикой используется представление рубрики в виде вектора, состоящего из терминов, схожих с заголовком рубрики, извлекаемых из WordNet.

3. Построение вектора рубрики

Чтобы решить, к каким рубрикам приписать документ, необходимо рассчитать меру сходства между документом и рубриками. Для этого документы и рубрики представляются как векторы в пространстве терминов коллекции.

Представление документов и рубрик как векторов производилось стандартным способом с помощью модели мешка слов (англ. bag of words). Для построения векторов рубрик использовался тезаурус WordNet, из которого с помощью специального алгоритма извлекались термины, похожие на заголовок рубрики.

Сходство между двумя документами, или между документом и рубрикой, вычисляется с помощью косинусной меры сходства между их векторами v(a) и v(b).

$$\mathrm{sim}(a, b) = \frac{(v(a), v(b))}{|v(a)||v(b)|}$$

Числитель представляет собой скалярное произведение векторов v(a) и v(b), а знаменатель равен произведению евклидовых норм этих векторов.

При запросе по некоторому термину WordNet в качестве результата выдает набор синсетов[2], относящихся в различным частям речи термина. В синсете перечислены соответствующие термины, описание смыслового значения и пример использования. Ниже приведен результат запроса по слову *interest*:

Noun:

- (62)S: (n) interest, involvement (a sense of concern with and curiosity about someone or something) "an interest in music"

[1] Recognising Textual Entailment Challenge, см. http://pascallin.ecs.soton.ac.uk/Challenges/RTE/

[2] Синсет - базовая словарная единица в WordNet, объединяющая слова со схожим значением, является узлом семантической сети WordNet.

- (32)S: (n) sake, interest (a reason for wanting something done) "for your sake"; "died for the sake of his country"; "in the interest of safety"; "in the common interest"
- (21)S: (n) interest, interestingness (the power of attracting or holding one's attention (because it is unusual or exciting etc.)) "they said nothing of great interest"; "primary colors can add interest to a room"
- ...

Verb:

- (5)S: (v) interest (excite the curiosity of; engage the interest of)
- (2)S: (v) concern, interest, occupy, worry (be on the mind of) "I worry about the second Germanic consonant shift"
- …

Число в скобках в описании синсета отображает частоту использования смысла термина, определенного данным синсетом, называемое далее *частотой синсета*. Так, для слова *interest* наиболее вероятный смысл - “a sense of concern with and curiosity about someone or something”, этот смысл имеет частоту 62, затем идет синсет со смыслом “a reason for wanting something done” с частотой 32 и т.д.

Алгоритм извлечения похожих терминов приведен ниже:

1 Извлечь *k* наиболее частых синсета названия рубрики.
2 Извлечь подходящие синонимы для каждого синсета названия рубрики. Синоним считается подходящим, если *k* его наиболее частых синсетов содержат название рубрики.
3 Извлечь подходящие дериваты[1] для каждого синсета названия рубрики. Дериват считается подходящим, если *k* его наиболее частых синсетов содержат соответствующий синсет названия рубрики.
4 Для каждого синсета названия рубрики извлекаем соответствующие синсеты гипонимов, из которых берем подходящие гипонимы. Гипоним определяется как подходящий аналогично деривату.
5 Если не было извлечено ни одного гипонима, аналогичным образом извлекаем список подходящих меронимов.
6 Для всех добавленных в результат гипонимов и меронимов извлекаются их *k* наиболее частых синсетов. В результат добавляются все подходящие дериваты для каждого синсета гипонима или меронима.

С помощью этого алгоритма в вектор категории включались как термины, синонимичные названию рубрики, так и термины, обозначающие более узкий смысл рубрики. Это обеспечивает большую семантическую полноту вектора рубрики, что позволяет более точно определять меру сходства между документами и данной рубрикой.

Таблица 1. Примеры векторов рубрик

Рубрика	Элементы вектора рубрики
grain	grain, cereal, wheat, rice, malt, Indian rice, maltster, coarse-grained
interest	interest, involvement, sake, enthusiasm, concern, occupy, apt, vexation
trade	trade, craft, merchandise, selling, marketing, import, fair trade, plumber

[1] Дериваты в WordNet - это список словообразовательных вариаций термина, полученных из некоторого синсета этого термина. Например, дериваты для синсетов термина *money* включают термины *monetary* и *moneyer*, а для синсетов термина *grain* - *grainy*, *granular* и *granulate*.

В таблице 1 приведены примеры рубрик и некоторые из соответствующих рубрикам элементов векторов. Как видно из примеров, при извлечении терминов из WordNet также рассматриваются словосочетания.

Оптимальным значением для параметра *k* оказалось 4.

4. Взвешивание вектора рубрики

В векторе рубрики, сформированном с помощью WordNet, изначально все элементы имеют одинаковый единичный вес. Однако, если взвесить каждый элемент вектора в соответствии со степенью сходства с рубрикой, то определение сходства между этой рубрикой и документами коллекции станет точнее.

Было рассмотрено несколько способов взвешивания векторов рубрик:

1 **Увеличение веса названия рубрики**. Все элементы вектора получали единичный вес за исключением элемента, совпадающего с названием категории, который получал вес, равный $w = 0.5n$, где n- размерность вектора.

2 **Взвешивание на основе наличия термина в коллекции**. Все элементы вектора получали единичный вес, за исключением тех, которые не встречались в документах коллекции - они получали нулевой вес.

3 **Взвешивание на основе наличия термина в коллекции и увеличения веса названия рубрики**. Как следует из названия, данный способ взвешивания является комбинацией двух предыдущих.

4 **Взвешивание, основанное на корреляции между рубрикой и элементом вектора**. Для этого способа взвешивания каждый элемент получал вес, равный $w = mut/occ$, где occ- число документов, в которых появлялся элемент, а mut- число документов, в которых появлялся и элемент, и название рубрики. Очевидно, что веса всех элементов будут находится в интервале [0, 1], причем элемент, совпадающий с названием рубрики, всегда будет иметь вес, равный единице.

Самый лучший результат дал последний способ, основанный на корреляции элементов вектора с рубрикой. Если из WordNet был извлечен термин, никогда не встречающийся в коллекции, вес соответствующего элемента вектора будет равен нулю. Если термин и название рубрики никогда не встречались вместе в одном документе, вес соответствующего элемента вектора также будет нулевым. За счет вычисления взаимной информации обеспечивается проверка на контекстное сходство между элементом вектора и рубрикой.

В таблице 2 представлены примеры элементов вектора рубрики *interest* с использованием разных способов взвешивания (вектора нормализованы):

Таблица 2. Веса элемента вектора interest для разных способов взвешивания

№ сп.	interest	involvemen	sake	enthusiam	concern	occupy	apt	vexation
1	0.971	0.030	0.030	0.030	0.030	0.030	0.030	0.030
2	0.189	0.189	0.189	0.189	0.189	0.189	0.189	0
3	0.987	0.031	0.031	0.031	0.031	0.031	0.031	0
4	0.912	0.094	0	0.152	0.104	0.130	0	0

5. Рубрикация неразмеченных документов с помощью kNN

Степень сходства рубрики и документа определяется через косинус между их векторами. Чтобы разметить документ некоторой рубрикой, необходимо, чтобы степень сходства превышала некоторое пороговое значение. Однако для некоторых документов степени сходства со всеми рубриками оказались ниже порогового значения, и таким образом они остались неразмеченными. Как уже указывалось во введении, мы можем предположить, что все документы коллекции должны быть отнесены к какой-либо рубрике и применить дополнительные меры сходства уже между документами.

Для рубрикации неразмеченных документов применяется алгоритм kNN. Сначала выбирается *k* наиболее похожих на данный документов. Степень сходства документов определяется также через косинус между векторами.

Каждый из похожих документов может быть отмечен несколькими рубриками. Среди похожих документов осуществляется поиск максимальной рубрикации, то есть наиболее часто встречаемого способа пометить документ. Если максимальный способ рубрикации единственный, то документ размечается этим способом. Если нашлось несколько максимальных способов рубрикации, то среди всех похожих документов осуществляется поиск единственной одиночной максимальной рубрики. Если она нашлась, то документ размечается ей, иначе документ остается неразмеченным.

В таблице 3 приведены примеры работы данного этапа. В третьем примере два максимальных способа рубрикации - {*grain*, *trade*} и {*money*, *crude*} с частотами 2, поэтому ищется максимальная одиночная рубрика, которой является рубрика *crude*, с частотой 3. В четвертом примере аналогичная ситуация, за исключением того, что максимальных одиночных рубрик оказалось несколько (*money*, *crude*, *grain* и *trade* с частотой 2), следовательно, документ остается неразмеченным.

Таблица 3. Примеры применения kNN для рубрикации документа при k = 5

сосед 1	сосед 2	сосед 3	сосед 4	сосед 5	результат
grain	grain	Grain	trade	trade	grain
grain, trade	grain, trade	trade	money	trade, money	grain, trade
grain, trade	grain, trade	crude	money, crude	money, crude	crude
grain, trade	grain, trade	corn	money, crude	money, crude	-

Были рассмотрены параметры *k*, принимающие значения 3, 5, 7, 9 (всего 10 рубрик). Оптимальным оказалось значение *k* = 9.

В результате применения kNN из 4573 неразмеченных документов осталось 66 неразмеченных.

6. Эксперименты по рубрикации текстов

В таблице 4 приведены результаты рубрикации при использовании только вектора, извлеченного из WordNet, затем при дополнительном взвешивании векторов рубрик, и затем при подключении kNN. В таблице используются следующие сокращения: *ma_p* - макроусредненная точность, *ma_r* - макроусредненная полнота,

ma_f - макроусредненная f1-мера, *mi_p*, *mi_r*, *mi_f* - аналогично микроусредненные характеристики [Агеев, Кураленок, 2004] .

Полученная разметка документов может рассматриваться как обучающая выборка для методов машинного обучения. Поэтому был проведен эксперимент, использующий полученную таким образом разметку документов для автоматической классификации методом SVM.

Таблица 4. Результаты рубрикации коллекции Reuters-10

	ma_p	ma_r	ma_f	mi_p	mi_r	mi_f
ТолькоWordNet	0.6375	0.5983	0.6173	0.7017	0.6343	0.6663
Со взвешиванием	0.6231	0.7235	0.6696	0.5953	0.7422	0.6607
С kNN	0.7128	0.6684	**0.6899**	0.7540	0.7512	**0.7526**

При тестировании разбиение на обучающую и тестовую коллекции производилось методом кросс-валидации с параметром 10, результаты классификации усреднены для всех 10 разбиений. В качестве классификатора SVM использовалась библиотека libsvm[1] вместе с дополнительными инструментами для рубрикации несколькими рубриками[2]. Были получены следующие результаты:

ma_f = 0.4836

mi_f = 0.7494

В работе [Barak et al. 2009] рубрикатор был также испытан на коллекции Reuters-10. При построении обучающей коллекции на основе сходства с рубрикой была получена f1: *mi_f = 0.76* (ср. последнюю строку в таблице 4). Однако по сравнению с нашим подходом в данной работе в качестве дополнительного ресурса использовалась еще и Википедия, которая требует значительных усилий по ее предобработке.

Кроме того, в этой же работе рубрикатор также использовался для обучения SVM. Эксперименты проводились для двух способов представления документов: как векторов в пространстве взвешенных с помощью tf-idf терминов, и как векторов в LSI пространстве. Значения максимальных f1-мер таковы (величины макромеры f1 в цитируемой статье не приводятся):

mi_f = 0.65 для tf-idf, и

mi_f = 0.79 для LSI.

Таким образом, и в данном случае для сравнимого текстового представления в нашей работе были получены более качественные результаты, что означает, что наше предположение о том, что использование метода kNN для разметки текстов соответствует принципами построения рубрикаторов для текстовых коллекций.

7. Заключение

В данной работе продемонстрировано, что, имея в распоряжении лишь заголовки рубрик, можно достичь неплохих результатов при рубрикации текстов. Для этого было использовано расширение вектора рубрики с помощью данных из WordNet, взвешивание вектора рубрики, основанное на взаимной информации между

[1] http://www.csie.ntu.edu.tw/~cjlin/libsvm/index.html

[2] http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/multilabel/

рубрикой и элементом вектора, и рубрикация неразмеченных документов с помощью алгоритма kNN.

Список литературы

[Barak et al., 2009] Barak L., Dagan I., Shnarch E. University of Toronto, Bar-Ilan University: Text Categorization from Category Name via Lexical Reference // Proceedings of NAACL HLT 2009: Short Papers, pages 33-36, Boulder, Colorado, June 2009.

[Bates, 1988] Bates M. 1988. How to use controlled vocabularies more effectively in online searching // Online archive V.12, Issue 6, pp. 45-56.

[Gliozzo et al., 2005] Gliozzo A., Strapparava C., Dagan I. Istituto per la Ricerca Scientifica e Tecnologica, Bar Ilan University: Investigating Unsupervised Learning for Text Categorization Bootstrapping // Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP), pages 129–136, Vancouver, October 2005.

[Glickman et al., 2006] Glickman O., Shnarch E., Dagan I. Bar Ilan University: Lexical Reference: a Semantic Matching Subtask // Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP 2006), pages 172–179, Sydney, July 2006.

[Miller, 1998] Miller G. Nouns in WordNet // WordNet – An Electronic Lexical Database / Fellbaum, C (ed). The MIT Press, 1998. P. 23-47.

[Rodriguez et al., 1997] Rodriguez M., Gomez-Hidalgo J., Diaz-Agudo B. Universidad Complutense de Madrid: Using WordNet to complement training information in text categorization // Recent Advances in Natural Language Processing, 1997.

[Sebastiani, 2002] Sebastiani F. Machine Learning in Automated Text Categorization // ACM Computing Reviews. 2002. V.34, N 1.

[Агеев, Кураленок, 2004] Агеев М.С., Кураленок И.Е. Официальные метрики РОМИП’2004. // Российский семинар по Оценке Методов Информационного Поиска. Пущино, 2004.

[Агеев и др., 2008] Агеев М.С., Добров Б.В., Лукашевич Н.В. Автоматическая рубрикация текстов: методы и проблемы // Ученые записки Казанского государственного университета. Серия Физико-математические науки. 2008. Том 150. книга 4. С. 25-40.

Кластеризация оценочных слов по тональности на основе Марковских цепей

Четвёркин И.И. (ilia2010@yandex.ru)
Факультет вычислительной математики и кибернетики МГУ

1. Введение

Задача анализа отзывов пользователей (по отношению автора к объекту, тональности, эмоциям) является очень актуальной и имеет ряд возможных приложений, например, сбор и обработка мнений пользователей о новых продуктах по блогам.

Отправной точкой для различных задач анализа мнений является создание словаря оценочной лексики. Каждое слово (или выражение) в таком ресурсе должно обладать некоторой тональностью, например положительной или отрицательной. С помощью такого рода словарей решаются различные задачи, такие как классификация или извлечение мнений (Chetviorkin & Loukachevitch, 2012).

В данной работе описывается подход к определению тональности набора оценочных слов на основе текстовой коллекции в заданной предметной области. Предлагаемый подход не использует словарей или каких-либо других семантических ресурсов, а опирается только на ряд предположений относительно зависимости тональности подряд идущих слов от расстояния между ними..

При построении алгоритма используется список оценочных слов, извлеченный по методу, описанному в (Chetviorkin & Loukachevitch, 2012). Для оценочных слов в заданной предметной области строится неориентированный граф, где каждая вершина представляет случайную величину, обозначающую тональность некоторого слова из вышеупомянутого списка. Данный граф представляет собой Марковскую сеть, в которой заданы попарные и унарные потенциальные функции. Реализуемая модель схожа по построению с моделью Изинга, математической моделью статистической физики, предназначенной для описания намагничивания материала.

Для поиска наиболее вероятного, согласованного состояния поля, используется алгоритм распространения доверия, который показывает более высокие результаты, чем метод самосогласованного поля (Weiss, 2001), используемый в схожих ситуациях (Takamura et al., 2005).

Дальнейшее изложение статьи будет построено следующим образом: В разделе 2 будет проведен обзор существующих методов построения словаря, в разделе 3 мы формально опишем Марковские случайные поля и модель Изинга. В разделе 4 будет описано построение Марковской сети по коллекции текстов и в завершении, в разделе 5 будут описаны эксперименты и оценки качества предлагаемого подхода.

2. Методы построения словарей оценочных слов

Большинство подходов к автоматическому построение словарей оценочных слов основано на различных способах расширения начального, маленького множества «посевных» слов. В качестве такого множества обычно используется некоторый общий, не зависящий от предметной области список (Kanayama & Nasukawa, 2007; Lau et al., 2011; Qiu et al., 2011). Одним из наиболее распространенных методов является итеративное формирование словаря оценочных выражений на основе тезауруса WordNet или других семантических ресурсов. Основной принцип в данном методе заключается в том, что синонимы и антонимы оценочного слова также

являются оценочными. Таким образом, из начального множества слов может быть получено новое, более полное множество оценочных слов (Hu & Liu, 2004; Neviarouskaya et. al, 2009). В (Esuli & Sebastiani, 2005) для конструирования оценочного словаря используются толкования слов. Основной идее данной работы является предположение, что слова с одинаковой оценочной ориентацией имеют схожие толкования. В таких подходах важным является правильное задание начального множества слов для расширения, т.к. от этого существенно зависит качество конечного ресурса.

Один из наиболее близких методов, к предлагаемому в данной работе алгоритму, описан в работе (Takamura et al., 2005). В этой работе авторы также строят Марковскую сеть из слов на основе похожести толкований и связей в семантическом ресурсе WordNet. В качестве дополнительного источника информации также используют совместную встречаемость слов через союзы И, ИЛИ и НО. Отличие нашего подхода заключается в том, что мы используем в построении только заранее отобранные потенциальные оценочные слова в конкретной предметной области и не опираемся на какие-либо словарные ресурсы.

Вторая группа подходов основана на поиске закономерностей, правил и шаблонов в текстовых коллекциях (Kanayama & Nasukawa, 2006; Lu et al., 2011). В некоторых работах используются статистические характеристики для извлечения оценочных слов: χ^2 (Jijkoun et el., 2010), DFR, характеристика которая определяет дивергенцию между распределением слов в множестве оценочных документов (релевантных) и в множестве нейтральных документов (He et al., 2009).

В (Velikovich et al., 2010) авторы тестируют качество оценочного словаря, полученного по огромному массиву текстовой информации, собранному с четырех миллиардов Веб страниц. Авторы формируют граф похожести слов по контекстам и используют алгоритм распространения меток на нем. Полученный словарь позволил получить более высокое качество работы в нескольких задачах обработки мнений по сравнению с уже существующими словарями на английском языке. Кроме того, данный ресурс содержит большое количество сленговых, вульгарных и других несловарных слов.

В литературе также встречаются некоторые работы, которые комбинируют словарный подход и закономерности, найденные в текстовых коллекциях (Ding et al., 2008).

Наш подход отличается от вышеописанных тем, что состоит из двух этапов:

- На первом этапе с помощью метода описанного в (Chetviorkin & Loukachevitch, 2012) извлекаются потенциальные оценочные слова, которые характерны для заданной предметной области;
- На втором этапе строится Марковская сеть из извлеченных оценочных слов на основе их совместной встречаемости и без использования каких-либо дополнительных словарных ресурсов.

За счет этого получается качественный список оценочных слов с тональностями для заданной предметной области.

3. Марковские сети и модель Изинга

В данном разделе будет сделано краткое введение в теорию Марковских случайных полей и модель Изинга.

Систему случайных величин, зависящих от нескольких параметров, обычно называют случайным полем. Понятие случайного поля есть естественное обобщение

понятия случайного процесса $\xi_t(\omega)$ на случай, когда параметр t имеет размерность, большую единицы. В данной работе мы рассматриваем случайные величины с дискретными аргументами.

Пусть $t = (t^1, t^2 \ldots t^v)$ совокупность векторов в евклидовом пространстве, где каждая компонента принимает целочисленные значения. Множество таких точек обозначим за T^v. Сопоставим каждой точке $t \in T^v$ случайную величину $\xi(t)$. Если $\xi(t)$ принимает значение в некотором конечном множестве X, то совместное распределение имеет вид:

$$P(\xi(t_1) = x_1, \ldots \xi(t_n) = x_n)$$

где $x_i \in X, t_i \in T^v, i = 1,2,\ldots,n$

Введем понятие условной вероятности того, что случайное поле приняло некоторые значения $x_1, x_2, \ldots, x_n$ на точках множества $S = \{t_1, t_2, \ldots, t_n\}$, при условии, что вне этого множества оно приняло значения $x(t) \in X$ (Добрушин, 1968),

$$q_{S,x(t)}(x_1, \ldots, x_n) = P\{\xi(t_1) = x_1, \ldots \xi(t_n) = x_n \mid \xi(t) = x(t), t \in T^v \setminus S\}$$

Обозначим $\bar{S}$ множество точек T^v не принадлежащих S и находящихся на единичном расстоянии от S. Тогда поле является **Марковским** если:

$$P\{\xi(t_1) = x_1, \ldots \xi(t_n) = x_n \mid \xi(t) = x(t), t \in T^v \setminus S\} = P\{\xi(t_1) = x_1, \ldots, \xi(t_n) = x_n \mid \xi(t) = x(t), t \in \bar{S}\}$$

То есть случайная величина условно независима от всех других величин, при заданных значениях её соседей. В более простой форме:

Марковское случайное поле – это вероятностная модель, в которой зависимости между случайными величинами представлены в виде неориентированного графа и множество случайных величин обладает Марковским свойством.

Для параметризации Марковских сетей часто используют Гиббсовскую факторизацию, т.к. это зачастую более удобный и эквивалентный способ определения (Аверинцев, 1970):

$$q_{s,x(\cdot)}(y) = \frac{1}{Z(s, x(\cdot))} \exp\left\{ \sum_{t \in \{\bar{s}\}} U(y, x(t), s, t) \right\}$$

где нормирующий множитель:

$$Z(s, x(\cdot)) = \sum_{y \in X} \exp\left\{ \sum_{t \in \{\bar{s}\}} U(y, x(t), s, t) \right\}$$

Действительную функцию $U(y, x(t), s, t)$ часто называют потенциалом переменных $x, y \in X$ и $s, t \in T^v$, она может принимать любые вещественные значения, и симметрична относительно перестановки переменных x, y.

Модель Изинга. Пусть задана система из N частиц в пространстве, каждая из которых имеет свой спин (равный +1 или -1). Спины соседних частиц взаимодействуют друг с другом и энергетически пытаются иметь одинаковое направление. Такая модель называется моделью Изинга и её энергетическая функция системы может быть представлена в следующем виде:

$$E(x, S, H) = -\sum_{ij} s_{ij} x_i x_j - \sum_i h_i x_i$$

Где x_i задает спин частицы i, s_{ij} – энергия взаимодействия между двумя частицами, h_i величина воздействия некоторого внешнего поля.

Вероятность принять то или иное состояние для системы задается распределением Больцмана (частный случай распределения Гиббса):

$$P(x \mid S,H) = \frac{1}{Z(S,H)} \exp(-\beta \cdot E(x,S,H))$$

где коэффициент β пропорционален обратной температуре. В традиционной модели Изинга от температуры зависит фазовое состояние системы:

- При высокой температуре, частицы имеют случайные спины (парамагнетики)
- При низкой температуре, большинство спинов направлены в одну сторону (ферромагнетики)
- Также известно, что при некоторой промежуточной температуре ферромагнетики становятся парамагнетиками. Этот процесс называется фазовым переходом.

Непосредственно перед фазовым переходом спины всех частиц локально согласованы. Именно это состояние сети нас и будет интересовать, т.к. мы предполагаем, что тональности оценочных слов также локально согласованы. Для поиска данного состояния мы будет проверять состояние системы при различных значениях β.

4. Построение и инициализация модели

В данной работе описывается подход к определению тональности набора оценочных слов на основе текстовой коллекции отзывов о фильмах. При построении аналога модели Изинга, в которой вместо электронов представлены оценочные слова, каждое из которых может иметь тональность -1 или +1, используются унарные и парные потенциальные функции, учитывающие взаимодействия между словами, а также начальные значения для тональностей всех слов.

Для задания унарных потенциалов мы воспользовались оценками, поставленными авторами отзывов. Для каждого слова можно вычислить среднюю оценку тех отзывов, где оно встречалось. Эта величина будет обладать достаточно большой зашумленностью, но, тем не менее, будет коррелированна с правильной тональностью слова. Введем определение унарных потенциалов более формально.

Каждому отзыву в коллекции соответствует численная оценка по 10 балльной шкале, поставленная его автором. Пусть $C = \{1\ldots10\}$ будет множество возможных оценок в коллекции отзывов, а $f(w,c)$ частота слова w в отзывах с оценкой c.

Определение 1. Условная вероятность каждой категории в зависимости от слова:

$$P(c \mid w) = \frac{f(w,c)}{\sum_{c_i \in C} f(w,c_i)}$$

Определение 2. Условная вероятность для каждого слова в зависимости от категории:

$$P(w \mid c) = \frac{f(w,c)}{\sum_{w_i \in C} f(w_i,c)}$$

Определение 3. Условное математическое ожидание для каждой категории в зависимости от слова:

$$E(c \mid w) = \sum_{c_i \in C} c_i \cdot P(c_i \mid w)$$

Определение 4. Математическое ожидание каждой категории в коллекции отзывов:

$$E(c) = \sum_{c_i \in C} c_i \cdot P(c_i)$$

С учетом данных определений, для каждого оценочного слова w_i мы вычисляли его **унарный потенциал** исходя из отклонения от средней оценки по корпусу, которая вычислялась по формуле:

$$h_i = E(c \mid w_i) - E(c)$$

Для потенциалов всех слов производилась нормировка к шкале [-1;1].

Попарные потенциалы слов зависят от частоты совместной встречаемости слов и среднего расстояния между ними. Для примера можно рассмотреть отзыв о фильме: *Хороший+, трешовый+ фильм, с отличным+ чувством юмора. Для любителей гая ритчи самое то, вот только картинка нищенская⁻, но ничего страшного+. Это даже колорит+ какой-то придает.*

Можно заметить, что чем ближе слова друг к другу, тем более вероятно совпадение их тональности. Для проверки этой гипотезы этапе были вычислены вероятности:

$$P(tn(w_i) = tn(w_j) \mid d(w_i, w_j)) = \frac{f(tn(w_i) = tn(w_j) \mid d(w_i, w_j))}{f(w_i, w_j \mid d(w_i, w_j))}$$

где $f(w_i, w_j)$ частота совместной встречаемости слов w_i, w_j, а $d(w_i, w_j)$ – среднее расстояние между словами w_i, w_j, а $tn(w_i)$ – тональность слова, принимает значение +1 или -1.

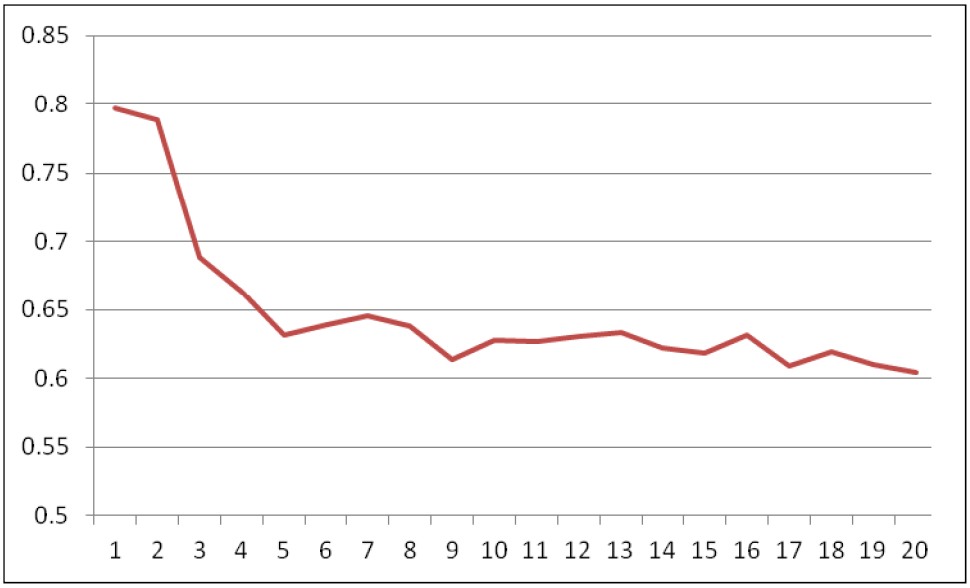


Рис. 1. Зависимость вероятности сохранения тональности от расстояния

На графике изображена эмпирическая зависимость между вероятностью сохранения тональности и расстоянием между двумя словами (знаки препинания не учитываются). Можно заметить, что при расстоянии большем четырёх вероятность

существенно падает и остается примерно на одинаковом уровне с дальнейшим увеличением расстояния.

Исходя из эмпирических данных, для всех попарных связей вес вычислялся по следующей формуле:

$$s_{ij} = \frac{f(w_i, w_j)}{3} \cdot \left(0.5 - \max\left(\frac{d(w_i, w_j)}{d(w_i, w_j) + 4}, 0.5 \right) \right)$$

Кроме того, в некоторых случаях тональность близко стоящих оценочных слов может меняться на противоположную, в зависимости от некоторых отрицательных частиц, таких как *НЕ* или *НИ* и союзов, таких как *НО* или *А*. Чтобы учесть влияние данных частиц и союзов, для каждой пары оценочных слов вычисляется частота встречаемости этих слов с отрицательными частицами перед ними, либо с вышеописанными союзами в контекстах между ними. Если число случаев употребления слов с отрицаниями превосходит 60% от общего числа совместных появлений в коллекции, тогда знак попарного потенциала s_{ij} меняется на противоположный.

5. Эксперименты и оценка качества

На базе текстовой коллекции из 28773 отзывов о фильмах (каждому отзыву соответствует численная оценка от 1 до 10), был извлечен список оценочных слов с использованием алгоритма, описанного в (Chetviorkin & Loukachevitch, 2012). Для работы были оставлены 3000 наиболее вероятных оценочных слов.

При построении Марковской сети были отброшены все связи с частотой совместной встречаемости меньше двух, после чего осталось 1906 связанных оценочных слов.

Данные слова были размечены двумя экспертами по следующей шкале:

- положительное слово
- отрицательное слово
- оценочное слово, тональность которого зависит от контекста
- нейтральное слово, извлеченное вследствие ошибки алгоритма на первом этапе

Для оценки качества кластеризации было принято решение оставить только слова с точно известной тональностью: положительной или отрицательной. Всего оставшихся слов с заданной тональностью – 1014. Метрикой качества в данной задаче являлась правильность, т.е. количество правильно классифицированных слов, деленное на количество всех слов с заданной тональностью.

В качестве базового алгоритма (baseline) было взято отклонение от средней оценки, такое же, как и вес h_i в графе. Если отклонение от средней оценки было больше 0, то слово считалось положительным, иначе отрицательным. Качество разделения слов на два класса с учетом данного правила составило 79.2%.

Основной задачей в рамках построенной модели является поиск минимума энергетической функции или поиск наиболее вероятного состояния Марковского случайного поля. Общая энергия системы задается функцией от парных взаимодействий и начальных значений для каждого оценочного слова:

$$E(x, S, H) = -\sum_{ij} s_{ij} x_i x_j - \sum_i h_i x_i$$

Где x_i задает тональность оценочного слова (+1 или -1), s_{ij} вес связи между словами, h_i начальный потенциал каждого слова.

При поиске наиболее вероятного состояния построенного Марковского случайного поля использовался алгоритм распространения доверия (Weiss, 2001) из библиотеки для дискретного приближенного вывода в графических моделях libDAI (Mooij, 2010). Для применения алгоритмов оптимизации Марковская сеть была представлена в виде фактор графа – неориентированного двудольный графа, в котором рёбрами соединены факторы и случайные переменные. В качестве факторов выступали описанные выше унарные и попарные потенциальные функции, представленные в виде таблиц значений для всевозможных конфигураций, которые они могут принимать.

Как уже было описано в разделе 3, мы пытаемся найти состояние сети, при котором тональности всех слов являются локально согласованными. Для этого эксперименты были проведены при различных значениях β.

Таблица 1. Зависимость качества кластеризации от параметра β

β	Правильность
0.2	81.2%
0.4	82.5%
0.6	83.2%
0.8	83.6%
0.9	83.9%
1.0	**84.1%**
1.1	83.1%
1.2	75.3%

В результате работы алгоритма было найдена оценка апостериорного максимума (MAP) состояния сети, согласно которой качество кластеризации составило 84.1% . Как и описывалось в разделе 3, при возрастании значения параметра β тональности слов начинают принимать случайные значения (фазовый переход). В процессе перехода от низкого β к высокому нам удалось найти состояние, когда тональности слов локально согласованы и при этом качество кластеризации достигает наивысшего значения. Примеры оценочных слов разделенных по тональности: *ненапряжный+*, *несмешной–*, *неподражаемый+*, *невнятный–*, *неправдоподобный–*, *нежный+*, *отличнейший+*, *бездарный-*, *обалденный+* и т.п.

6. Заключение

В данной работе предложена адаптация модели Изинга, для задачи кластеризации оценочных слов по тональности. При построении данной модели используется только коллекция отзывов о фильмах с оценками пользователей без каких-либо дополнительных семантических ресурсов. Из данной коллекции извлекается набор оценочных слов, на основе которого строится Марковская сеть для заданной предметной области. Часть извлеченных слов имеет нейтральную тональность, либо тональность, зависящую от контекстов, что усложняет поиск согласованного состояния в сети. Тем не менее, результирующее качество кластеризации предложенной модели превосходит базовый уровень и составляет 84.1%. Предложенный метод является полностью автоматическим и не нуждается в ручной разметке.

Благодарности. Работа частично поддержана грантом РФФИ N11-07-00588-а.

Литература

1. Chetviorkin I. I., Loukachevitch N. V. Extraction of Russian Sentiment Lexicon for Product Meta-Domain *In Proceedings of COLING 2012: Technical Papers* , pages 593–610

2. Ding X., Liu B., and Yu P. S. 2008. A holistic lexicon-based approach to opinion mining. *In WSDM '08*, pages 231–240.

3. Esuli A., Sebastiani F. 2005 Determining the Semantic Orientation of Terms through Gloss Classification. In: *Conference of Information and Knowledge Management*

4. He B., Macdonald C., He J., and Ounis I. 2009. An effective statistical approach to blog post opinion retrieval. In *Proceedings of the 17th ACM CIKM*, pp. 1063-1072.

5. Hu M., Liu B. 2004. Mining and Summarizing Customer Reviews. *KDD*

6. Jijkoun V., de Rijke M., and Weerkamp W. 2010. Generating focused topic-specific sentiment lexicons. *In ACL '10*, pages 585–594

7. Kanayama H. and Nasukawa T. 2006. Fully automatic lexicon expansion for domain-oriented sentiment analysis. *In EMNLP '06*, Morristown, NJ, USA. pages 355–363

8. Lau R., Lai C., Bruza P. and Wong K. 2011. Pseudo Labeling for Scalable Semi-supervised Learning of Domain-specific Sentiment Lexicons. In *20th ACM Conference on Information and Knowledge Management*.

9. Lu Y., Castellanos M., Dayal U. and Zhai C. 2011. Automatic Construction of a Context-Aware Sentiment Lexicon: An Optimization Approach *In Proceedings of the World Wide Web Conference (WWW)*

10. Mooij J. M. 2010. libDAI: A free & open source C++ library for Discrete Approximate Inference in graphical models; *Journal of Machine Learning Research*, pages 2169–2173,

11. Neviarouskaya A., Prendinger H., and Ishizuka M. 2009. Sentiful: Generating a reliable lexicon for sentiment analysis. *In ACII*, pages 1–6

12. Qiu G., Liu B., Bu J. and Chen C. (2011). Opinion word expansion and target extraction through double propagation. *Computational Linguistics*, 37(1).

13. Takamura H., Inui T., and Okumura M. 2005. Extracting Semantic Orientations of Words using Spin Model. *In ACL*, pp 133–141

14. Velikovich, L., Blair-Goldensohn, S., Hannan, K. and McDonald, R. 2010. The viability of web-dervied polarity lexicons. In Proceedings of Human Language Technologies: *The 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics*.

15. Weiss Y. 2001. Comparing the mean field method and belief propagation for approximate inference in MRFs. In Saad and Opper, editors, *Advanced Mean Field Methods*. MIT Press,

16. Аверинцев М. Б. 1970. Об одном способе описания случайных полей с дискретным аргументом, *Проблемы передачи информации*, 6:2 100–108

17. Добрушин Р. Л. 1968. Описание случайного поля при помощи условных вероятностей и условия его регулярности, *Теория вероятностей и ее применения*, 13:2 201–229

Автоматизация оценки благозвучия текстов

Иванов К.В.
МИЭМ НИУ ВШЭ

Аннотация. Статья посвящена описанию метода автоматизированной оценки благозвучия текстов на русском языке.

Введение

Впервые явление эвфонии начали изучать философы Древней Греции. Они ставили искусство красивой речи на один уровень с сочинением музыки и стихосложением. Аристотель утверждал: «Написанное должно быть читабельно и легко произносимо, что одно и то же».

Благозвучный текст оказывает положительный эффект на качество восприятия информации слушателем. Наличие такой своеобразной звуковой гармонии делает информацию более доступной и запоминающейся. Неестественное звучание фраз отвлекает внимание слушателя, мешает общему восприятию. Человек на подсознательном уровне теряет интерес к соответствующей теме еще на раннем этапе пути к ее пониманию, не успевая задуматься о смысле поступающей информации.

Необычный эффект эвфонии можно также почувствовать, послушав определенный музыкальный мотив, способный остаться в памяти слушателя на долгое время. Похожая своеобразная мелодичность сопровождает и устную речь, порождая особый подсознательный резонанс у субъекта, от чего информация кажется более приятной.

Постановка задачи

Если попытаться подойти к вопросу об оценке благозвучия того или иного текста с научной точки зрения, окажется, что эта оценка является исключительно субъективной мерой. В первую очередь следует отметить отличия в межнациональных языковых структурах. Даже один и тот же текст может восприниматься по-разному людьми, разговаривающими на различных языках. Это следует из того, что каждый народ имеет свою уникальную фонетическую систему, постепенно формировавшуюся в течение многих веков. Некоторые фонемы одного языка могут вовсе отсутствовать в другом, что может привести к некоторым сложностям взаимосвязи людей разных национальностей. Диалектические особенности того или иного языка делают затруднительным общение внутри одной и той же страны. Поэтому чтобы иметь научное основание для оценки благозвучия, в первую очередь, следует проводить исследования в рамках лишь одного национального языка.

Однако даже субъекты одного региона могут иметь различные представления о красоте речи. Тут следует учитывать массу факторов, на которых строится базис восприятия каждого человека в отдельности. Вопросами восприятия речи как синтеза языковых средств активно занимаются психологи [1, 2], поэтому в данной работе не будут рассматриваться частные случаи, подразумевающие индивидуальное мышление. Для наших целей необходимо выделить особые критерии, по которым оценка благозвучия стала бы наиболее объективной, актуальной для большинства людей.

Кроме того, существует ряд особенностей, которые могут исказить качество информации того или иного источника (в тексте – шрифт, материал бумаги; в речи – голос, интонация). Поэтому благозвучие следует рассматривать при моделировании идеальных условий.

Но даже если таким образом сузить область рассмотрения вопроса, остается непонятным, по каким критериям оценивать благозвучность конкретных текстов.

Обзор существующих решений

Анализ благозвучия и читабельности текстов по своей идеологии преследует одну и ту же цель – оценить и, по возможности, стилистически улучшить исходный текст. Однако из-за отсутствия четко разработанной научной базы, исследования и разработки в области эвфонической оценки текстов за все время были немногочисленны. Оценка параметров удобочитаемости представляется более понятной и осязаемой. Поэтому именно в этой области уже имеется множество наработок.

Эксперт в области анализа читабельности Рудольф Флеш разработал метод [3], оценивающий сложность текста по формуле:

FRE = 206.835 − (1.015 × ASL) − (84.6 × ASW), где:

ASL – средняя длина предложения в словах;

ASW – средняя длина слова.

Позже эта формула была модернизирована для русского языка в связи с тем, что средняя длина предложения в русском языке меньше, а средняя длина слова – больше:

FRE = 206.835 − (1.3 × ASL) − (60.1 × ASW).

Данный индекс удобочитаемости признан эталонной мерой оценки читабельности и широко используется во всем мире, в частности, для анализа текстов на экзаменах в школах и университетах.

Автоматизированный индекс читабельности был разработан для ВВС США в 1967 году. Данная оценка зависит от количества символов в слове и от количества слов в предложении:

ARI = (4,71 × ASW) + (0,5 × ASL) – 21,43

В американской журналистике часто применяется Индекс Фога. Метод включает учет так называемых сложных слов, не являющихся именами собственными и имеют более 3 слогов:

FI = (NWS + NWT) × 0,4, где:

NWS – среднее число слов в предложении текста;

NWT – среднее число слов с длиной 3 и более слогов.

В работе [4] приведено описание программы LightReader, производящей оперативную оценку качества восприятия текстов. Система собирает количественные характеристики текста, такие как среднее количество слогов в слове, среднее число слов в предложении, среднее количество согласных звуков в слове, среднее число знаков препинания в предложении и процент многосложных слов в тексте, содержащих более трёх слогов. Легкость чтения рассчитывается по вышеупомянутой формуле Флеша.

В работе [5] описан алгоритм разработки программного продукта для экспертной оценки сложности учебных текстов. В основе этой методики – поиск признаков, характеризующих трудность восприятия: по количеству найденных формулировок, абстрактных слов и сложносочиненных предложений, формируется конечная оценка.

Офисный пакет Office от корпорации Microsoft предоставляет достаточно внушительный набор средств по работе с текстом. Среди них имеется инструмент, собирающий статистику удобочитаемости. Помимо оценки легкости чтения и подсчета количества сложных фраз, программа способна оценивать благозвучие. Описание метода расчета оценки удобочитаемости нигде не приведено. Эмпирическим путем было установлено, что алгоритм представляет собой следующие шаги:

[1] Присвоение нулевого коэффициента буквам «с», «ц», «ч», «ш», «щ».

[2] Присвоение единичного коэффициента остальным буквам.

[3] Результат – отношение количества букв с единичным коэффициентом на общее количество букв в тексте.

Очевидно, что метод, основанный лишь на исключении каких-то неблагозвучных букв, не способен дать качественную эвфоническую оценку.

Разработка метода

В учебном пособии [6] представлен наиболее подробный свод стилистических правил русского языка, позволяющих добиться максимального качества звуковой сочетаемости при соединении слов в словосочетания и предложения. На основе этих данных был отобран ряд критериев, которые, в той или иной мере, затрудняют артикуляцию:

Диагностирующие признаки, ухудшающие благозвучие слова (диссонирующие единицы):

- Наличие более 2 идущих подряд уникальных согласных;
- Наличие идущих подряд гласных (внутреннее зияние);
- Наличие более 1 шипящего звука.

Диагностирующие признаки, ухудшающие благозвучие предложения на стыке слов (диссонирующие переходы):

6. Наличие более 3 идущих подряд уникальных согласных на стыке слов;
7. Наличие одинаковых или парных друг другу согласных на стыке слов;
8. Наличие гласных на стыке слов (внешнее зияние).

Также были предложены гипотезы о влиянии длины слов и частоты употребления той или иной буквы в предложении на общую эвфоническую картину. В ходе исследований был детально проанализирован корпус скороговорок, на основе которого удалось получить конкретные показатели неблагозвучия:

12. Наличие слов, у которых более 5 слогов;
13. Отношение частоты определенной согласной к общему количеству букв в предложении превышает значение 0,12.

В результате исследований была разработана мера оценки благозвучия – эвфонический индекс (*Euphony Index*):

$$\mathbf{EI} = \frac{\sum\left(\left(\mathbf{1} - \frac{\mathbf{NDT} + \mathbf{NDU}}{\mathbf{CSL}}\right)\frac{\mathbf{NIL}}{\mathbf{NUL}} \times \frac{\mathbf{NCW}}{\mathbf{CSL}}\right)}{\mathbf{TNS}}, \text{ где}$$

NDT – количество диссонирующих переходов (*Number of Dissonant Transitions*);

NDU – количество диссонирующих единиц (*Number of Dissonant Units*);
CSL – длина текущего предложения (*Current Sentence Length*);
NIL – количество нечастых букв (*Number of Infrequent Letters*);
NUL – количество уникальных букв (*Number of Unique Letters*);
NCW – количество недлинных слов (*Number of Compact Words*);
TNS – общее количество предложений (*Total Number of Sentences*).

Чем больше полученное значение, тем благозвучнее текст.

Результаты экспериментов

Для анализа работы метода алгоритм был применен к нескольким отрывкам текстов на русском языке.

Произведение	Кол-во предложений	EI
Художественные тексты	273	0.815
Новостная лента	145	0.712
Скороговорки	20	0.586

Результат в интервале от 0,6 до 1 является удовлетворительным.

Результат менее 0,6 является неудовлетворительным. Исходный текст необходимо доработать.

Выводы

В данной работе был изложен алгоритм, позволяющий измерить благозвучие текста. В ходе исследования был отобран ряд признаков русского языка, которые негативно сказываются на общей эвфонической картине. Для представления оценки в числовом виде, была введена специальная мера – эвфонический индекс.

Приведенный метод актуален только для русского языка, но может быть также модернизирован и переработан и для других языков при наличии соответствующей теоретической базы. Разработанный алгоритм может послужить основой для создания автоматизированной системы улучшения благозвучия. Метод также может быть включен в уже существующие системы оценки сложности и читабельности текстов.

Список литературы

[1] *Лебединский С. И.* Восприятие устной научной речи: стратегии сегментации звучащей речи и идентификации терминов: монография // С. И. Лебединский. – Минск: БГУ, 2008. – 535 с.

[2] *Чистович Л. А., Венцов А. В., Гранстрем М. П.* Физиология речи. Восприятие речи человеком // Издательство «Наука», 1978.

[3] *Rudolf Flesch.* A new readability yardstick, Journal of Applied Psychology, 32, С. 221—233, 1948.

[4] *Оборнева И.В.* Автоматизация оценки качества восприятия текста // Вестник МГПУ. Серия «Информатика и информатизация образования». – М.: МГПУ, 2005.

[5] *Криони Н.К., Никин А.Д., Филиппова А.В.* Автоматизированная система анализа сложности учебных текстов. Уфа: УГАТУ, 2008.

[6] *Голуб И.Б.* – Стилистика русского языка // М.: Рольф; Айрис-пресс, 1997 - 448 с.

Синтаксический анализ музыкальных текстов

Ирина Голубева[1], Андрей Юрьевич Филиппович[2]
[1]*МГТУ имени Н.Э. Баумана, Москва, Россия.* irinadanshina@mail.ru
[2]*МГТУ имени Н.Э. Баумана, Москва, Россия.* philippovich@list.ru

Аннотация. Статья посвящена изучению синтаксиса музыкальных текстов. Показана возможность применения лингвистических методов для проведения анализа. В качестве конкретного материала для исследований выбраны музыкальные рукописи XI-XVII веков. Для их анализа предложены типы отношений между знаками, составлена синтаксическая модель, проведен статистический анализ.

Ключевые слова: модель языка, синтаксический анализ, статискический анализ, анализ музыки, знаменные песнопения, семиография, древние рукописи

Введение

В настоящее время актуально изучение вопросов невербальной коммуникации человека. В сообщениях, которые передаются, могут быть не только текстовые составляющие, но и музыкальные, эмоциональные, жестовые. Вопросы познания, восприятия, понимания музыки получили названия музыкальные инфо-когнитивные технологии. Инфо-когнитивные технологии (ИКТ) – технологии, которые связанны с тем, как построены процессы обработки и восприятия информации человеком. Это позволяет рассматривать не только то, как расположены знаки в тексте, но и то, как они зависят друг от друга (синтаксис) и как музыка воспринимается человеком (семантика).

В качестве конкретного материала для исследований выбраны музыкальные рукописи XI-XVII веков, записанные в знаменной нотации (рисунок 1). Знаменные песнопения появились на Руси в XI веке вместе с принятием христианства. Они представляют собой музыкальные тексты, записанные с помощью специальных знаков, отличающихся по своей природе от современных нот. Основные их отличия в том, что одному знаку («знамени») может соотвестсвовать несколько звуков и нет точного соответствия между знаком и высотой звука.

В результате реформ музыкальных нотаций был утрачен «ключ» к расшифровке знаменных мелодий, который позволяет точно перевести старинные песнопения в ното-линейную систему [7].

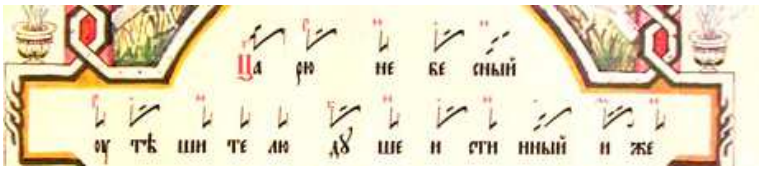


Рис. 1. Страница музыкальной рукописи в знаменной нотации

Для достоверного перевода необходимо выявлять в знаменной нотации внутренние законы, в силу которых мелодии записывались с помощью одних знамен, а не других.

Данная работа выполняется в рамках проекта «Автоматизированная система научных исследований в области компьютерной семиографии (АНСИ КС)», а также поддержана грантом РГНФ №110412025в. Было проведено исследование и выявлено, что знаменные песнопения имеют аналогичную структуру, что и естественный язык. В качестве подтверждения этого можно привести примеры аналогий между ними: Знаменный алфавит → Лексикон, Последовательность знамен → Словосочетания,

предложения, Знамя (крюк) → Слово, лексема, Пометы, Признаки → Диакритические знаки. Это позволяет применять лингвистические методы для обработки и анализа песнопений.

Описанные ниже результаты получены с помощью программ SemioStatistik [3] и Semantic_Statistik [4], специально разработанных для анализа знаменных песнопений.

Дополнительная информация о проекте и проведенных исследованиях представлена в Интернете по адресу http://it-claim.ru/semio.

Статистический анализ

Для того, чтобы проверить гипотезу о наличии в знаменных песнопениях семиотической и синтаксической структуры, знаменная нотация была представлена как знаковая система [4].

Знамя — это графическое изображение, используемое для обозначения определенной высоты, длительности и характера исполнения. Алфавит знаменной нотации — список знамен.

Были проведены статистические исследования. На рисунке 2 приведен график распределения частот знамен для книги «Октоих», содежащей знаменные песнопения.

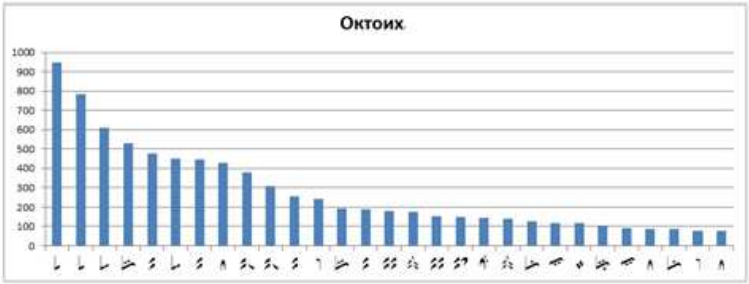


Рис.2. Частота встречаемости знамен

В результате данного этапа, было подтверждено, что знамена распределены в песнопении неравномерно, существуют «специальные» знаки для начала или окончания фразы, а также определенные сочетания знамен по 2, 3 и 4 знамени встречаются чаще, чем другие.

	257
	194
	174
	157
	137
	136
	136
	133
	131

Рис.3. Наиболее частотные сочетания по три знамени

В дополнение было проверено, что данное распределение соответсвует закону Ципфа-Мальденброта и закону Г.С. Хипса, которые справедливы для текста на естественном языке.

Последующий анализ позволил выделить группы знамен по частоте их встречаемости:

Табл. 1. Распределение знамен на 4 частотные группы

№ группы	I	II	III	IV
Число повторений каждого знамени	1000-3000	500-1000	150-500	1-150
Количество знамен в группе	6	7	30	323
Базовые знамена группы				…

Изучение синтаксиса

Для того, чтобы определить как связаны знамена между собой, было предложено 3 типа отношений, представленных в таблице 2.

Табл. 2. Три типа отношений в синтаксисе знаменных песнопений

Вид отношения	**Описание**	**Пример**
α – отношение: $Z_1 \xrightarrow{\alpha} Z_2$	Знамя Z_1 состоит в α – отношении со знаменем Z_2, если Z_2 является производным от Z_1.	
β – отношение: $Z_1 \xrightarrow{\beta} Z_2$	Знамя Z_1 состоит в β – отношении со знаменем Z_2, если в структуре песнопения знамя Z_2 непосредственно следует за знаменем Z_1	⇔
Вероятностное β – отношение: $Z_1 \xrightarrow{\beta} Z_2\ (P_i)$	Если за знаменем Z_1 может следовать несколько знамен, то конкретное знамя (Z_2) следует с вероятностью P_i.	(0,56)
γ – отношение: $Z_1 \xrightarrow{\gamma} Z_2$ $Z_2 \xrightarrow{\gamma} Z_1$	Знамя Z_1 состоит в γ – отношении со знаменем Z_2, если эти знамена находятся в общем контексте (фразе, предложении, песнопении).	⇔

Совокупность правил, соответствующих каждому из отношений, образует модель данного отношения. Таким образом, синтаксическая модель знаменных песнопений представляется в виде тройки:

M = $\{M_\alpha, M_\beta, M_\gamma\}$, где

$M_\alpha = \{(Z_1 \xrightarrow{\alpha} Z_2), p_\alpha\}_{n_\alpha}$: α-модель знаменных песнопений,

p_α- вероятность употребления α-правила,

n_α – количество α-правил,

$M_\beta = \{(Z_1 \xrightarrow{\beta} Z_2), p_\beta\}_{n_\beta}$: β-модель знаменных песнопений,

p_β-коэффициент непосредственной силы связи между знаменами,

n_β- количество β-правил,

$M_\gamma = \{(Z_1 \xrightarrow{\gamma} Z_2), p_\gamma\}_{n_\gamma}$: γ-модель знаменных песнопений,

p_γ-коэффициент контекстной силы связи между знаменами,

n_γ- - количество γ-правил.

Были построены модели для каждого типа отношений по каждой из частей одного из крупнейших источников древнерусских музыкальных текстов - рукописи «Круг знаменных песнопений» под ред. Разумовского.

Фрагмент $M_\beta^{\text{Октоих}}$ (β – модели для части «Октоих») приведен в таблице 3 и на рисунке 4.

Для удобства анализа в строках и столбцах записаны цифровые коды знамен, а в соответствующих ячейках – коэффициенты непосредственной силы связи знамен.

Табл. 3. Фрагмент β – модели для «Октоиха»

	76	77	121	138	89	122
76	66	4	12	0	0	0
77	5	33	0	0	0	1
121	0	1	0	1	4	0
138	1	1	0	0	0	1
89	0	0	0	8	0	0
122	1	0	0	2	0	0

На рисунке 4 таблица 3 представлена в виде гистограммы. На горизонтальной оси расположены знамена, а по вертикальной оси откладываются соответствующие значения коэффициентов силы в процентном соотношении.

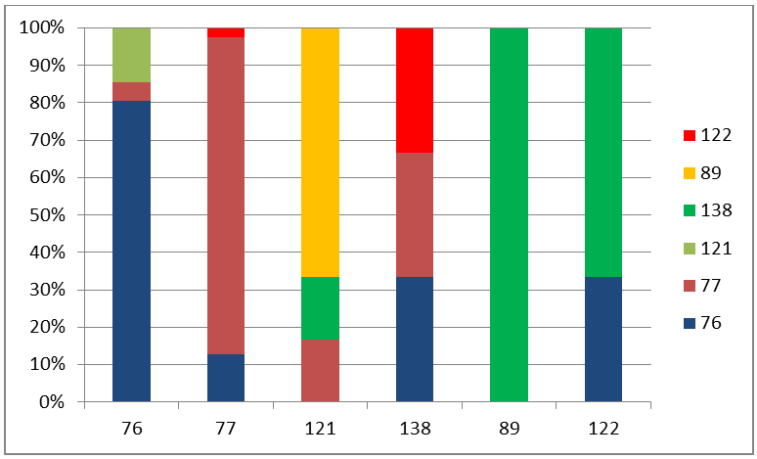


Рис.4. Фрагмент β – модели для «Октоиха»

Фрагмент $M_\gamma^{\text{Октоих}}$ (γ – модели для части «Октоих») приведен в таблице 4 и на рисунке 5. Значения в них заданы аналогично β-модели.

Табл. 4. Фрагмент γ – модели для «Октоиха»

	76	77	121	138	89	122
76	0,142	0,024	0,04	0	0	0
77	0,024	0,05	0	0	0	0,002
121	0	0,009	0	0,013	0,011	0
138	0,022	0,007	0	0	0	0,009
89	0	0,004	0	0,018	0	0
122	0,002	0	0	0,009	0	0

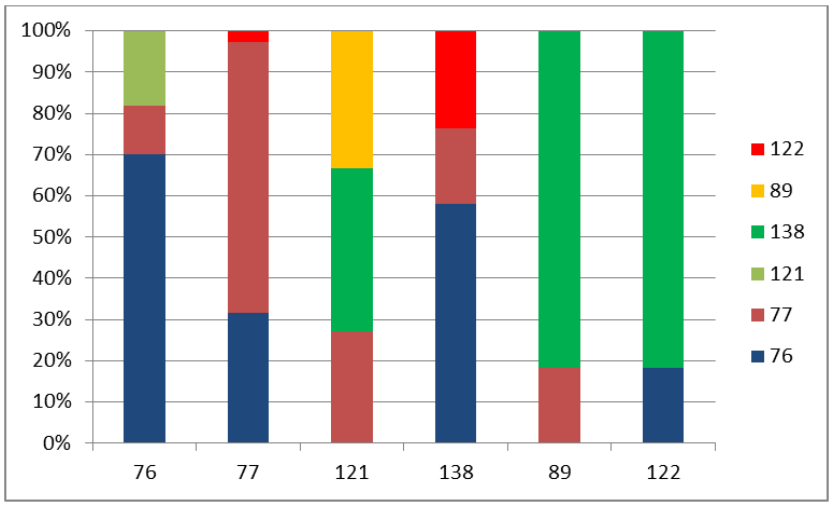


Рис.5. Фрагмент γ – модели для «Октоиха»

Данный этап позволил выявить особый характер связей между знаменами, который не виден на письме. Построение γ – модели позволяет выявить контекстно-связанные знамена, что дополняет общую картину связности знамен. Например, для приведенного отрывка выявлено, что знамя с кодом «89» связано не только со знаменем «138», но и с «77».

Таким образом, знаменная нотация была представлена как знаковая система, что позволило применить к ее изучению лингвистические методы. В дальнейшем предполагается использовать построенные модели для решения задач ИКТ – восприятия и понимания информации человеком.

Выводы

1. Эмпирически было доказано наличие в музыкальных текстах семиотической и синтаксической структур

2. Было предложено три типа отношений между знаками в музыке

3. Была предложена синтаксическая модель знаменных песнопений, которая может быть применена для решения задач восприятия, понимания и обработки музыкальной информации.

Список источников

1. Даньшина И.В., Даньшина М.В. Статистическое исследование знаменной системы на примере двух гласов Октоиха // Интеллектуальные технологии и системы. Сборник учебно-методических работ и статей аспирантов и студентов. Выпуск 9. М.: НОК «CLAIM», 2007. Даньшина 2. И.В., Даньшина М.В. Структура и обработка древнерусских певческих рукописей.// Сборник тезисов докладов «Печатные средства информации в современном обществе (к 80-летию МГУП)». Секция «Электронные средства информации в современном обществе», М. 2010

3. Даньшина И.В. Программа для проведения статистических исследований «SemioStatistik» // Интеллектуальные технологии и системы. Сборник учебно-методических работ и статей аспирантов и студентов. Выпуск 10. М.: НОК «CLAIM», 2009.

4. Даньшина И.В. Анализ зависимости слов и знамен с помощью программы Semantic_Statistik. Сборник тезисов докладов общеуниверситетской научно-технической конференции «Студенческая научная весна - 2011», посвященной 50-летию полета Ю.А. Гагарина в космос. Том XI, часть 2. М.: МГТУ им. Н.Э. Баумана, 2011. - 310 с.

5. Голубева И.В. Исследование знаменных песнопений как знаковой системы. Информационные технологии и письменное наследие: материалы IV междунар. науч. конф. (Петрозаводск, 3-8 сентября 2012 г.), 2012

6. Филиппович А.Ю., Голубева И.В. Исследование синтаксиса семиографических песнопений. Проблемы полиграфии и издательского дела, 2012

7. М.Бражников. «Древнерусская теория музыки». - «Музыка», 1972г.

Применение методов машинного перевода для анализа древнерусских музыкальных рукописей

Марина Даньшина
МГТУ имени Н.Э. Баумана, Москва, Россия. marina_danshina@mail.ru

Введение

Предшественником линейной нотации, используемой в настоящее время для фиксирования мелодии, была знаменная нотация. Ее особенностью было то, что мелодия записывалась не с помощью нот на линейках, а специальными знаками – крюками, которые имели сложную структуру. Изначально рукописи не содержали подсказок исполнителю о высоте или длительности ноты, однако спустя некоторое время в музыкальные книги стали добавлять пометы, облегчающие чтение песнопения. С течением времени знания о том, как необходимо воспроизводить мелодию стали фиксировать в специальных книгах (азбуках), позже появились рукописи, содержащие мелодию в двух нотациях – знаменной и нотной. Такие книги являются аналогами параллельных корпусов текстов и именно они являются главным источником информации для расшифровки знаменных песнопений, несмотря на то, что данные в них неполны и иногда противоречивы. Помимо этого следует учитывать, что рукописи содержат специальные структуры (фиты, лица), которые, аналогично фразеологизмам в тексте, необходимо переводить особенным образом. [7]

Общее количество знамен, с помощью которых производилась запись, оценивается по-разному. В нашем исследовании были экспериментально выявлено 202 знамени. При этом каждое знамя может переводиться одной или несколькими нотами. Пример знаменной рукописи приведен на рисунке 1.

Рис. 1. Пример знаменной рукописи

Обработка рукописей

В рамках проекта «Компьютерная семиография» реализуются задачи по созданию конкретных инструментов, позволяющих автоматизировать рутинные операции перевода знаменных песнопений в линейную нотацию. Данная работа поддержана грантом РГНФ №110412025в.

В качестве основных исходных данных были выбраны четыре типа музыкальных рукописей:

- музыкальные азбуки;
- кокизники (сборники фит и лиц);
- сборники попевок;
- двоезнаменники.

Для обработки каждого типа рукописей предложены отдельные инструменты и технологии. Например, для перевода на основе азбук можно составить список продукционных правил с приоритетами и осуществить экспериментальную

дешифровку. Для этого создан музыкальный проигрыватель, который показывает результаты перевода не только визуально, но и позволяет проанализировать мелодию на слух. Приоритеты используются в тех случаях, когда при дешифровке нужно переводить сочетания знамен.

Пример результата перевода музыкальным проигрывателем приведен на рисунке 2.

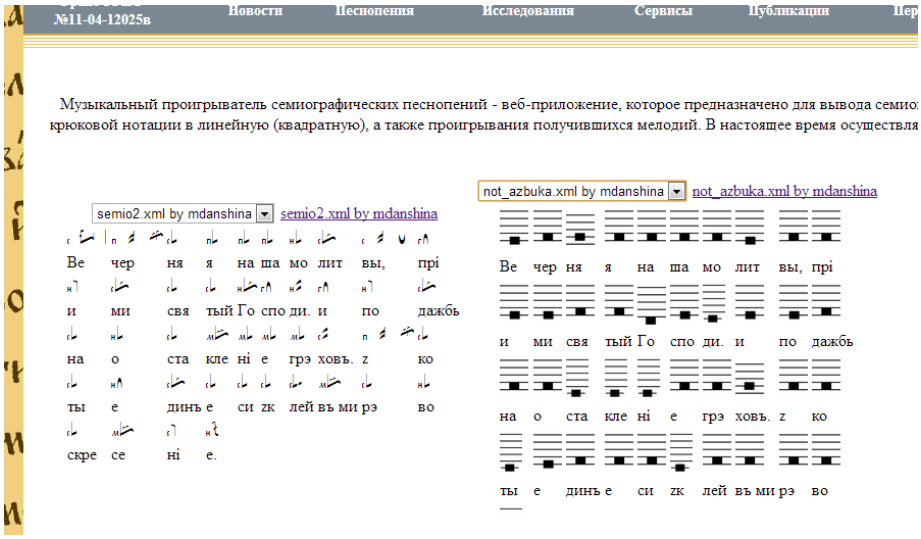


Рис. 2. Пример перевода музыкальным редактором

Исходными данными для музыкального редактора являются:

1. Знаменное песнопение в формате XML (Рис. 3)

```
▼<ROWDATA>
  <ROW Znam="а" Slog="ко" Stil=" обычный" VPom="м" DPom=""/>
  <ROW Znam="Ар" Slog="кэч" Stil=" обычный Italic" VPom="в" DPom=""/>
  <ROW Znam="а" Slog="но" Stil=" Bold" VPom="п" DPom=""/>
  <ROW Znam="а" Slog="му" Stil=" Bold" VPom="п" DPom=""/>
  <ROW Znam="а" Slog="t" Stil=" Bold" VPom="п" DPom=""/>
  <ROW Znam="а" Slog="от" Stil=" Bold" VPom="п" DPom=""/>
  <ROW Znam="а" Slog="шу" Stil=" Bold" VPom="п" DPom=""/>
  <ROW Znam="а" Slog="ся" Stil=" Italic" VPom="п" DPom=""/>
```

Рис. 3. Пример песнопений в XML-формате

2. Словарь для перевода в формате XML (Рис. 4), в котором закодированы ноты, которыми переводится знамя или последовательность знамен, а также соответствующие длительность и приоритет.

```
▼<ROWDATA>
  <ROW cod="200" note="e_1" length="04" p="0.7"/>
  <ROW cod="144" note="f_1" length="04" p="0.9"/>
  <ROW cod="304" note="g_1" length="04" p="0.9"/>
  <ROW cod="269" note="f_1" length="04" p="0.4"/>
  <ROW cod="177" note="e_1" length="04" p="0.4"/>
  <ROW cod="256" note="f_1" length="04" p="0.5"/>
```

Рис. 4. Пример словаря для дешифровки рукописи

Для анализа двузнаменников разработаны и апробированы различные технологии статистического перевода:

• методы построения "модели языка": на основе N-граммной модели – вероятность следования знамени определяется с учетом вероятностей предшествующих знамен.

• построение "модели перевода" в зависимости от характера "знаменных конструкций" (их размерности) может быть реализовано на основе:

1) текстовых фраз, которые сопровождают нотную запись – выбираются последовательности знамен, соответствующие предложению или его части (до знака препинания);

2) попевок – устойчивых сочетаний знамен из соответствующих сборников, составленных вручную древними авторами или исследователями;

3) фиксированного контекстного окна – выбранного количества знамен (используется в N-граммной модели).

Построение модели языка и модели перевода

В качестве модели языка строится триграммная языковая модель. Исходными данными для построения является двоезнаменный Ирмологий, который переведен в электронный вид и хранится в базе данных (Рис. 5).

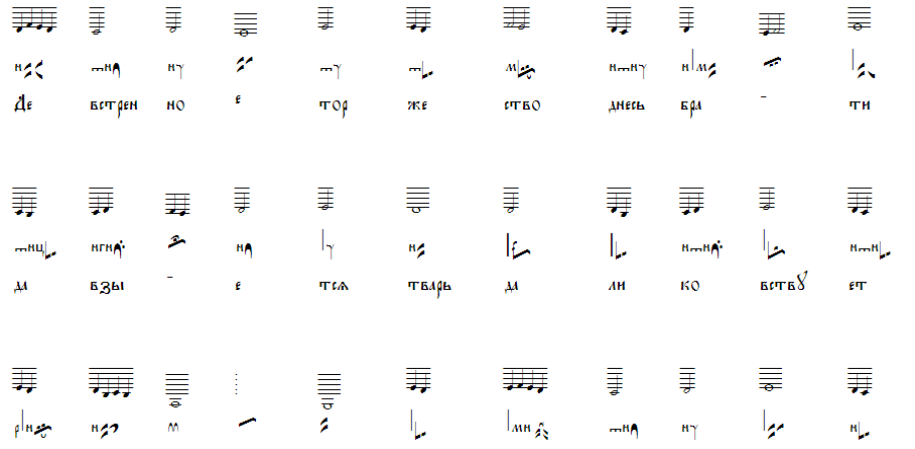


Рис. 5. Пример двоезнаменника в электронном виде

N-грамма	Вероятность со сглаживанием	Вероятность без сглаживания	N
	0,01139	0,666667	3
	0,015909	0,857143	2
	0,00431	0,032258	3
	0,056338	0,22963	2
	0,004348	0,037037	3
	0,025271	0,04	2
	0,004228	0,025	3
	0,049878	0,102828	2

Рис. 6. Пример языковой модели песнопений

Согласно статистическому машинному переводу модель языка назначает наибольшую вероятность наиболее частотным строкам (словам или фразам). В качестве «граммы» для знаменных песнопений выбраны последовательности нот, которые соответствуют знамени. Для учета недостатка неполноты исходных данных

используется метод сглаживания Лапласа, в соответствии с которым вероятность каждой n-граммы вычисляется следующим образом:

$$P_{lap}(w_i|w_{i-1}) = \frac{C(w_{i-1}w_i)+1}{C(w_{i-1})+|V|} \qquad (1)$$

Где c – исходное количество триграммы в тексте, |V| – число уникальных грамм в тексте [5,6,9].

Результаты построения языковой модели представлены в электронном виде на сайте проекта.

Модель перевода вычисляется по двуязычному корпусу и назначает наибольшую вероятность парам строк (слов или фраз) с одним значением.

Для построения модели перевода рассчитывается вероятность для каждой пары P(n|z), где z – последовательность знамен, а n – перевод этой последовательности. Данная вероятность рассчитывается по формуле (2.2).

$P(n|z) = \frac{C(n,z)}{C(z)}$ (2), где C(n,z) – количество раз, когда последовательность знамен z переводится нотами n.

Триграмма			Перевод			Вероятность
						0,017327
						0,318182
						0,014851
						0,272727
						0,073529
						0,3125
						0,153846
						0,111111
						0,333333
						0,2

Рис. 7. Пример модели перевода

Построение общего словаря для расшифровки знаменных песнопений

Итоговый словарь строится на основе предварительно полученных на основе азбук, сборников попевок и двоезнаменников словарей. А также у экспертов имеется возможность добавить в словарь новые правила, полученные на основе анализа материалов. При этом каждому правилу в словаре соответствует приоритет, который определяет очередность замены знамен. Таким образом, учитывается перевод попевок и других специальных структур.

Выводы

Полученный словарь позволяет перевести древнерусские музыкальные рукописи из крюковой нотации в линейную с учетом особенностей знаменных песнопений.

Использование нескольких методов построения словарей позволяет исследователям различным образом анализировать полученные переводы.

Построенные модель языка и модель перевода являются исходными данными для следующего этапа расшифровки знаменных песнопений – декодирования.

Список источников

1. Даньшина И.В., Даньшина М.В. Структура и обработка древнерусских певческих рукописей.// Сборник тезисов докладов «Печатные средства информации в современном обществе (к 80-летию МГУП)». Секция «Электронные средства информации в современном обществе», М. 2010
2. Даньшина М.В. Программа для ввода и обработки семиографических песнопений IPSM. Информационные технологии и письменное наследие: материалы междунар. науч. конф. / отв. Ред. В.А.Баранов. - Уфа;Ижевск: Вагант, 2010
3. Даньшина М.В. Метод выделения, сохранения и обработки попевок в музыкальной рукописи. Информационные технологии и письменное наследие: материалы IV междунар. науч. конф. (Петрозаводск, 2012 г.)
4. Даньшина М.В. Исследование семантической структуры попевок в знаменных песнопенях. Тезисы докладов на международном междисциплинарном форуме по прикладной когнитивистике CrossLingua'2012 “Когниция. Коммуникация. Культура”
5. Даньшина М.В. Использование n-граммной языковой модели для изучения знаменных песнопений. Сборник тезисов и статей Российско-Германской молодежной дистанционной научной школы «Актуальные и перспективные направления создания систем, обеспечивающих семантический анализ данных в режиме реального времени», 2012.
6. Knight K. A Statistical MT Tutorial Workbook. 1999
7. М.Бражников. «Древнерусская теория музыки». - «Музыка», 1972г.
8. Шабалин Д.С. Певческие азбуки Древней Руси. − Кемерово: Кузбассвузиздат, 1991. 211 с.
9. Bird S., Klein E., Loper E. Natural Language Processing with Python. 2009

Извлечение переводного словаря значимых словосочетаний из параллельных текстов с использованием методов дистрибутивной семантики

Морозова Ю.И.
ИПИ РАН

Аннотация: Данная работа посвящена актуальным проблемам исследования семантики лингвистических единиц с использованием корпусных методов. В работе дается описание нового направления лингвистических исследований – дистрибутивной семантики. Предлагается расширение существующих моделей дистрибутивной семантики за счет перехода от описания лексем к описанию значимых словосочетаний. Описывается методика извлечения переводного словаря из параллельных текстов.

Ключевые слова: дистрибутивная семантика, векторные пространства, значимые словосочетания, коллокации, параллельные тексты.

1 Обзор моделей дистрибутивной семантики

Дистрибутивная семантика – область научных исследований, занимающаяся вычислением степени семантической близости между лингвистическими единицами на основании их дистрибуционных признаков в больших массивах лингвистических данных. Модели векторных пространств находят все более широкое применение в исследованиях, связанных с семантическими моделями естественного языка, и имеют разнообразный спектр потенциальных и действующих приложений. Основными сферами применения дистрибутивных моделей являются: разрешение лексической неоднозначности, информационный поиск, кластеризация документов, автоматическое формирование словарей (словарей семантических отношений, двуязычных словарей), создание семантических карт, моделирование перифраз, определение тематики документа, определение тональности высказывания, биоинформатика.

Теоретические основы данного направления восходят к дистрибутивной методологии З. Харриса [7, 8]. Близкие идеи выдвигали основоположники структурной лингвистики Ф. де Соссюр и Л. Витгенштейн. Дистрибутивная семантика основывается на дистрибутивной гипотезе о том, что лингвистические элементы со схожей дистрибуцией имеют близкие значения [12, 15].

В качестве вычислительного инструмента и способа представления моделей используется линейная алгебра. Информация о дистрибуции лингвистических единиц представляется в виде многоразрядных векторов, а семантическая близость между лингвистическими единицами вычисляется как расстояние между векторами. Многоразрядные векторы образуют матрицу, где каждый вектор соответствует лингвистической единице (слово или словосочетание), а каждое измерение вектора соответствует контексту (документ, параграф, предложение, словосочетание, слово).

Для вычисления меры близости между векторами могут использоваться различные формулы: расстояние Минковского, расстояние Манхеттена, Евклидово расстояние, расстояние Чебышева, скалярное произведение, косинусная мера. Наиболее популярной является косинусная мера:

$$\frac{x \bullet y}{|x| \bullet |y|} = \frac{\sum_{i=1}^{n} x_i \bullet y_i}{\sqrt{\sum_{i=1}^{n} x_i^2} \bullet \sqrt{\sum_{i=1}^{n} y_i^2}}$$

Существует множество разновидностей моделей дистрибутивной семантики, которые различаются по следующим параметрам:

- тип контекста (размер контекста, правый или левый контекст, ранжирование);
- количественная оценка частоты встречаемости слова в данном контексте (абсолютная частота, энтропия, совместная информация и пр.);
- метод вычисления расстояния между векторами (косинус, скалярное произведение, расстояние Минковского и пр.);
- метод уменьшения размерности матрицы (случайная проекция, сингулярное разложение и пр.).

Наиболее известными моделями дистрибутивной семантики являются латентный семантический анализ, разработанный для решения проблемы синонимии при информационном поиске [9], и модель языка как гиперпространства, разработанная как модель семантической памяти человека [10].

Концепция семантических векторных пространств (СВП) впервые была реализована в информационно-поисковой системе SMART [14]. Идея СВП состоит в представлении каждого документа из коллекции в виде точки в пространстве, т.е. вектора в векторном пространстве. Точки, расположенные ближе друг к другу в этом пространстве, считаются более близкими по смыслу. Пользовательский запрос рассматривается как псевдодокумент и тоже представляется как точка в этом же пространстве. Документы сортируются в порядке возрастания расстояния, т.е в порядке уменьшения семантической близости от запроса, и в таком виде предоставляются пользователю.

Впоследствии концепция СВП была успешно применена для других семантических задач. Например, в работе [11] контекстное векторное пространство было использовано для оценки семантической близости слов. Данная система достигла результата 92.5% на тесте по выбору наиболее подходящего синонима из стандартного теста английского языка TOEFL, в то время как средний результат при прохождении теста человеком был 64.5%.

В настоящее время ведутся активные исследования по унификации модели СВП и выработке общего подхода к различным задачам выявления семантических связей из корпусов текстов [16].

В работе [13] предлагается методика применения моделей дистрибутивной семантики для извлечения переводных соответствий однословных терминов из выровненных параллельных текстов. Обычно в качестве базовой информации для систем извлечения переводных соответствий используется частота совместной встречаемости терминов из соответствующих друг другу фрагментов на исходном и целевом языках. Однако предположение о том, что перевод основывается на пословных соответствиях, не соответствует действительной сложности процесса перевода. Поэтому авторы предлагают использовать в качестве минимальной единицы анализа не слово, а предложение. Лексические единицы, встречающиеся в одном предложении, связаны друг с другом синтагматическими отношениями, в то время как все предложение целиком связано с его переводом на целевой язык отношениями переводного соответствия. Поэтому каждое слово в исходном предложении связано с каждым словом в целевом предложении.

В предлагаемой модели «контекстом» для слов предложений на исходном языке выступают слова предложений на целевом языке. Контекстные векторы, описывающие слова исходного и целевого языков, помещаются в одну и ту же матрицу. Корреляция между словами вычисляется по формуле косинуса угла между

их контекстными векторами. Слова из различных языков с наиболее близкими векторами считаются переводами друг друга.

Данный подход особенно продуктивен, когда нужно извлечь не только самый лучший перевод данного слова, но и несколько возможных переводов.

2 Выделение значимых словосочетаний

Целью нашей работы является применение модели СВП для извлечения переводного словаря значимых словосочетаний из параллельных текстов. Развитие существующих подходов к построению СВП заключается в использовании значимых словосочетаний (ЗС) вместо отдельных лексем. Под значимыми словосочетаниями мы понимаем лексические последовательности, имеющие тенденцию к совместной встречаемости.

В лингвистике для обозначения значимых словосочетаний используется также термин «коллокация». Этот термин был впервые введен в Словаре лингвистических терминов Ахмановой О. С. [1]. Исследованиям коллокаций русского языка посвящено большое количество литературы, например, монография Борисовой Е. Г. [2]. В теоретической лингвистике под коллокациями понимают словосочетания из двух или более слов, которые обуславливают друг друга семантически и грамматически [4]. В корпусной лингвистике коллокациями называют статистически устойчивые словосочетания, причем они могут быть как фразеологизированными, так и свободными.

Для выделения значимых словосочетаний в компьютерной лингвистике используются различные статистические меры (меры ассоциации, меры ассоциативной связанности, англ. association measures), вычисляющие силу связи между элементами в составе коллокации. В литературе упоминается несколько десятков мер ассоциативной связанности. Чаще других используются MI, t-score и log-likelihood [3].

Мера MI (mutual information), введенная в работе [5], сравнивает зависимые контекстно-связанные частоты с независимыми частотами слов в тексте. Если значение MI превосходит определенное пороговое значение, то словосочетание считают статистически значимым. Мера MI вычисляется по следующей формуле:

$$MI = \log_2 \frac{f(n,c) \times N}{f(n) \times f(c)}$$

где n – первое слово словосочетания; c – второе слово словосочетания; f(n,c) – частота совместной встречаемости двух слов; f(n), f(c) – абсолютные частоты встречаемости каждого слова по отдельности; N – общее число словоупотреблений в корпусе.

Мера t-score также используется при ответе на вопрос, насколько не случайным является сочетание двух или более слов в тексте. Для вычисления t-score используется следующая формула:

$$t-score = \frac{f(n,c) - \frac{f(n) \times f(c)}{N}}{\sqrt{f(n,c)}}$$

Также достаточно часто применяется мера, известная под названием log-likelihood, или логарифмическая функция правдоподобия, введенная в работе [6]. Для вычисления log-likelihood применяется следующая формула:

$$\log - likelihood = 2 \sum f(n,c) \times \log_2 \frac{f(n,c) \times N}{f(n) \times f(c)}$$

3 Построение семантического векторного пространства

В рамках нашего исследования был создан корпус параллельных текстов на французском и русском языках, выровненный на уровне предложений. В корпус вошли тексты научных патентов по различным темам.

Применив различные меры ассоциативной связанности слов к материалам научных патентов, мы составили частотный словарь значимых словосочетаний для предметной области научных патентов. Примеры выделенных значимых словосочетаний: *благородный металл, вспомогательное устройство, жесткий элемент, измерительная ячейка, опорный карниз, оптический луч, система охлаждения, тяжелая фракция.*

В продолжение исследования планируется построить СВП для выделенных значимых словосочетаний с помощью программы S-Space [7]. Модель семантического векторного пространства, которая будет создана, обладает следующими характеристиками:

- тип изучаемых единиц: значимые словосочетания;
- тип контекста: лексемы;
- размер контекста – предложение;
- ранжирование контекста – нет;
- количественная оценка частоты встречаемости изучаемой единицы в данном контексте: абсолютная частота;
- метод вычисления расстояния между векторами: косинусная мера.

4 Заключение

В работе были рассмотрены основные направления и модели нового направления исследований в компьютерной лингвистике – дистрибутивной семантики. На основании автоматической обработки больших массивов лингвистических данных возможно создавать различные лингвистические ресурсы: семантические словари, многоязычные словари, семантические карты предметных областей. В качестве математической модели используются многоразрядные векторы и матрицы линейной алгебры, что представляет собой удобный формализм для компьютерной реализации. В рамках данного направления планируется разработать методику извлечения переводных соответствий из параллельных текстов, где в качестве изучаемых единиц будут выступать значимые словосочетания, выделенные из текстов с использованием мер ассоциативной связанности слов.

Литература

1. *Ахманова О. С.* Словарь лингвистических терминов, М.: 1966.

2. *Борисова Е. Г.* Коллокации. Что это такое и как их изучать. М.: 1995.

3. *Захаров В. П., Хохлова М. В.* Анализ эффективности статистических методов выявления коллокаций в текстах на русском языке // Компьютерная лингвистика и интеллектуальные технологии: Труды Международной конференции Диалог'2010. – М.: Изд-во РГГУ, 2010.

4. *Иорданская Л. Н., Мельчук И. А.* Смысл и сочетаемость в словаре. М.: 2007.

5. *Church K., Hanks P.* Word association norms, mutual information, and lexicography // Computational Linguistics, 1996, № 16(1). - P. 22–29.

6. *Dunning T.* Accurate Methods for the Statistics of Surprise and Coincidence // Computational Linguistics, 1993. Volume 19, №1. - P. 61-74.

7. *Harris Z. S.* Papers in Structural and Transformational Linguistics. - Dordrecht, Reidel, 1954.

8. *Harris Z. S.* Mathematical Structures of Language. - Interscience publishers, 1968.

9. *Landauer Th. K., McNamara D. S., Dennis S., Kintsch W.* Handbook of Latent Semantic Analysis. - Mahwah NJ, Lawrence Erlbaum, 2007.

10. *Lund K., Burgess C.* Producing high-dimensional semantic spaces from lexical co-occurrence // Behavior Research Methods, Instruments & Computers, 1996, 28(2). - P. 203-208.

11. *Rapp R.* Word sense discovery based on sense descriptor dissimilarity // Proceedings of the 9th MT Summit. - New Orleans, LA, 2003. – P. 315–322.

12. *Sahlgren M.* The Distributional Hypothesis. From context to meaning // Distributional models of the lexicon in linguistics and cognitive science (Special issue of the Italian Journal of Linguistics), Rivista di Linguistica, volume 20, numero 1, 2008.

13. *Sahlgren M., Karlgren J.* Automatic Bilingual Lexicon Acquisition Using Random Indexing of Parallel Corpora // Journal of Natural Language Engineering, Special Issue on Parallel Texts, 2005, №11(3).

14. *Salton G.* M. The SMART Retrieval System: Experiments in Automatic Document Processing. - Prentice-Hall, 1971.

15. *Turney P. D., Pantel P.* From frequency to meaning: Vector space models of semantics // Journal of Artificial Intelligence Research (JAIR), 2010, №37. – P. 141-188.

16. *Turney P.* A uniform approach to analogies, synonyms, antonyms and associations // Proceedings of COLING, Manchester, 2008. – P. 905–912.

17. Интернет-сайт программы S-Space: https://github.com/fozziethebeat/S-Space

О синтезаторе таджикской речи по тексту

Х.А. Худойбердиев
Кафедра программное обеспечение и информационных технологий,
Худжандский политехнический институт Таджикского технического Университета,
г. Худжанд, Республика Таджикистан
tajlingvo@gmail.com, www.tajlingvo.tj

Аннотация. В настоящем докладе излагается идея синтезатора таджикской речи по тексту, который реализован на основе метода конкатенации слогов. Принцип работы приведена и описана в виде блок-схемы. На основе разработанной модели создан комплекс программ синтезирования таджикской речи Tajik Text-to-Speech. В конце доклада представлена структурная схема программного комплекса и краткое описание полученных результатов.

Ключевые слова: обработка текста; блок-схема; слог-звук; структура текстовой информации; синтезатор; синтез речи.

Текст – это последовательность предложений, построенных согласно правилам данного языка и данной знаковой системы и образующая сообщение.

В свою очередь, *предложение* будем рассматривать как совокупность упорядоченных элементов 7 типов, называемых *словом*, *числом*, *символом*, *пробелом*, *внутренним знаком препинания* (запятая, двоеточие, точка с запятой, тире), *внешним знаком препинания* (точка, многоточие, знак вопросительный, знак восклицательный) и, наконец, *служебным символом окончания абзаца* (в письменном тексте его нет, но он присутствует в электронном тексте как непечатаемый знак ¶).

Отметим, что смысл, который мы заключаем в названия элементов, следует понимать в общепринятых значениях. Подчеркнем также, что в конкретном предложении некоторые элементы могут отсутствовать (например, числа, символы, внутренние знаки препинания и т.д.) в то время как присутствие других - обязательно (например, внешний знак препинания).

Теперь опишем идею синтезирования речи по тексту в виде принципиальной блок-схемы, (см. рисунок 1).

Работа синтезатора происходит следующим образом. После ввода очередного предложения оно анализируется по составу своих элементов. Если очередной элемент – слово, то в блоке 1 оно разделяется на слоги с указанием ударного слога и затем осуществляется его озвучивание с использованием базы “слог-звук”.

Если очередной элемент – число, то оно в блоке 2 преобразуется в текст и затем его озвучивание происходит через блок 1.

Если очередной элемент – символ, то его озвучивание происходит в блоке 3 путем извлечения соответствующего звучания из базы “символ-звук”.

Если очередной элемент – пробел, внутренний или внешний знак препинания или же знак окончания абзаца, то для них из соответствующего блока извлекается соответствующая пауза.

Синтезатор речи, представленный в виде блок-схемы, подсказывает, что в его основу закладывается принцип конкатенации озвученных слогов. Поскольку слог выступает в качестве основной звуковой единицы речи, то для реализации синтезатора *требуется описать многообразие все слогов соответствующего естественного языка*.

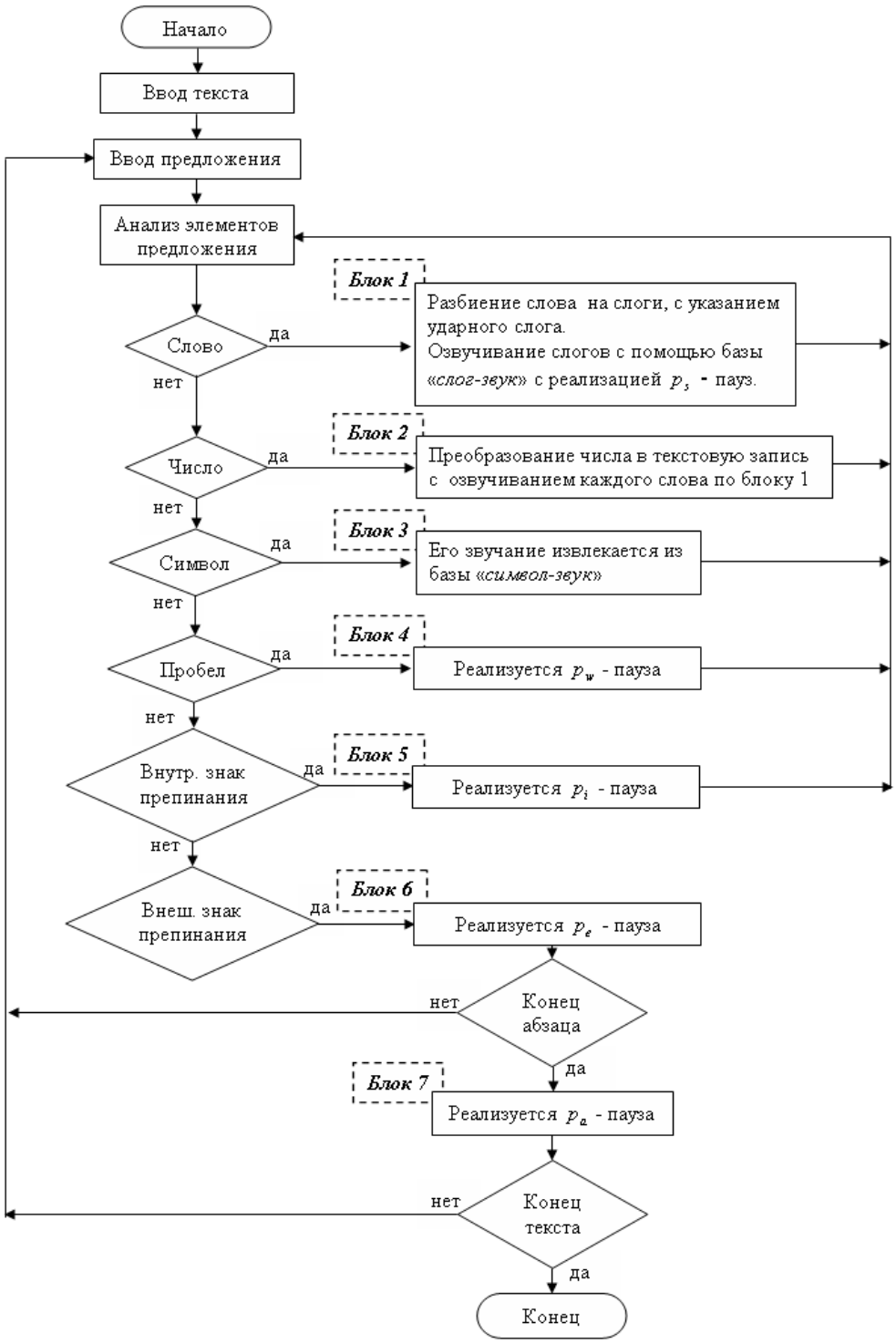


Рисунок 1. Принципиальная блок-схема синтеза речи по тексту

Поскольку каждый слог, представленный в виде цепочки букв, нуждается в его звуковом образе, то *требуется создание базы "слог-звук"*. Поскольку синтезатор предусматривает озвучивание чисел и символов, то для реализации синтезатора разработаны соответствующие алгоритмы и программы для трансформирования числа в текст и создания базы "символ-звук".

На основе детальных проработок концептуальной схемы был создан синтезатор Tajik Text-to-Speech, т.е. комплекс программ для синтезирования таджикской речи по тексту. Настройка параметров синтезатора проводилась путем вычислительных экспериментов и установлено удовлетворительные значения длительностей пауз: для

границ абзацев p_a = 900 мс; для границ предложений p_e = 600 мс; для запятых p_i = 400 мс; для межсловной и межслоговой пауз, p_w = 200 мс и p_s = 20 мс. Структурная схема программного комплекса представлена на рисунке 2.

Блок 1. Подсистема «**Интерфейс пользователя**» состоит из двух компонентов - «**Ввод текста**» и «**Озвученная речь**», которые имеют одностороннюю связь, т.е пользователь имеет возможность ввести текстовую информацию и в результате получить речевой вариант вводимого текста. Для получения результатов блок 1 связывается с блоком 2 по двум направлениям - по предоставлению информации для лингвистического анализа и получению результатов озвучивания. Блок 1 взаимодействует также и с блоком 3 непосредственно для использования необходимых данных о настройках системы (выбор мужского или женского голоса, громкость и скорость озвучивания).

Блок 2. Аналитическая подсистема состоит из двух частей - «**Лингвистического анализа**» и «**Модуля озвучивания**». Первый из них состоит из подмодулей «***Проверка текста***», «***Кодирование текста***» и «***Разделение слов на слоги***». «***Проверка текста***» используется для проверки вводимой информации, которая включает в себя такие текстовые элементы, как слова, целые числа, символы и знаки препинания. Данный подмодуль проверяет текстовые элементы, преобразует в тестовый вариант целые числа и символы и затем передает их для кодирования.

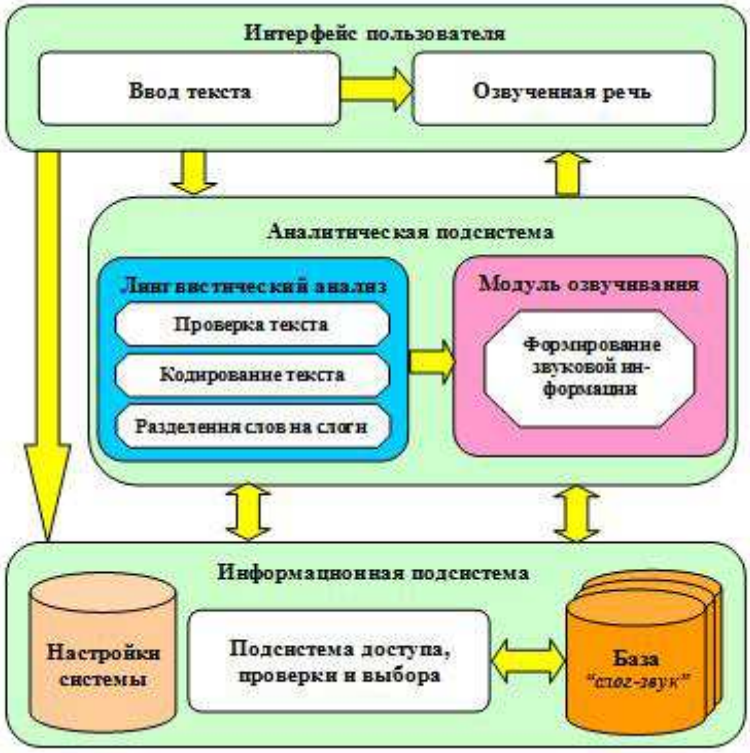


Рисунок 2 - Структурная схема Tajik Text-to-Speech

Процесс кодирования осуществляет одноименный подмодуль, который каждое слово W входного текста преобразует в упорядоченную совокупность $W^*_{0,1}$ нулей и единиц (напомним, что цифрой 1 замещаются гласные буквы, а цифрой 0 – согласные), т.е. все слова представляются своими слоговыми структурами.

Закодированный текст передается подмодулю ***«Разделение слов на слоги»,*** который работает на основе собственного алгоритма. Разделенные на слоги слова лингвистически анализируются и передаются «**Модулю озвучивания**».

В указанном модуле происходит формирование звуковой информации с использованием базы “слог-звук” информационной подсистемы, ударных слогов, межслоговых и межсловных пауз, а также пауз, отмечающих такие знаки препинания как запятая и точка.

Модуль озвучивания является заключительной стадией аналитической подсистемы, и звуковой вариант текстовой информации отправляется в интерфейс пользователя.

Блок 3, «**Информационная подсистема**» содержит базы данных, названные «**Настройкой системы**» и «**Базой *слог-звук***». Первая из них используется для хранения временных данных настройки системы, вторая база «***слог-звук***» - для хранения статистических данных о звуковых файлах 3259 слогов таджикского языка. Для работы с этой базой данных используется модуль обеспечения доступа, проверки и выборки необходимых данных.

Для оценки эффективности работы синтезатора были организованы эксперименты по озвучиванию разнообразной текстовой информации (фрагменты из повестей, романов, научных статьей, учебников, газет, журналов, Интернет - сайтов). Оценка полноты множества слогов, использованных для формирования синтетической речи, связывалась с процентом озвученных слов по отношению к общему количеству слов в пределах выбранных фрагментов текста.

Полученные результаты показали вполне удовлетворительное качество работы комплекса программ Tajik Text-to-Speech по озвучиванию таджикского текста.

Таким образом, комплекс программ Tajik Text-to-Speech, хотя и не решает полностью поставленной проблемы, ***все же в настоящее время оказывается первым программным продуктом, хорошо осуществляющим компьютерное озвучивание таджикских текстов***. На данном уровне разработки комплекс уже сейчас может быть использован людьми с ослабленным зрением.

Другие эксперименты проводились на научных семинарах Института математики АН РТ. Его участники по собственному усмотрению вводили в компьютер таджикские тексты и затем оценивали естественность и разборчивость звучания синтетической речи. Общее мнение семинара - ***компьютерный синтезатор, построенный по принципу конкатенации 3259 таджикских слогов, вполне успешно выполняет функцию озвучивания таджикских текстов.***

Список литературы

1. Советский энциклопедический словарь. -М.: Советская энциклопедия, 1980.1600 с.
2. Л.В.Златоустова, С.В.Кодзасов, О.Ф.Кривнова, И.Г.Фролова. Алгоритмы преобразования русских орфографических текстов в фонетическую запись -М.:МГУ, 1970.
3. Лобанов Б.М. Цирульник Л.И. Жадинец Д.В. Сизонов О.Г. Алгоритмы синтеза просодических характеристик речи по тексту в системе «Мультифон». Объединённый институт проблем информатики НАН Беларуси, Минск. 2007.
4. Лобанов Б.М., Цирульник М.И. Компьютерный синтез и клонирование речи, Минск, Белорусская наука, 2008, - 316 с.
5. Худойбердиев Х.А., О многообразии слогов таджикского языка // Известия АН РТ, № 2 (127), 2007. - с. 31-34.
6. Усманов З.Д., Худойбердиев Х.А. Алгоритм безударного озвучивания таджикского текста // ДАН РТ, Т.50, № 4, 2007. - с. 302-305.
7. Усманов З.Д., Худойбердиев Х.А. Компьютерное озвучивание таджикского текста // Патент (интеллектуальный продукт) зарегистрирован 041TJ 04.09.2007 НПИ центром.

Программный стенд для исследования глагольных конструкций таджикского языка

Д.Д.Собиров
Таджикский национальный университет
jamshed_corp@mail.ru

Аннотация. В статье дается описание разработанных программного стенда и информационных баз данных для исследования глагольных конструкций на основе алгоритма распознавания глаголов таджикского языка.

Ключевые слова: таджикский язык, глагольная конструкция, группа сказуемого, алгоритм, морфологический анализ, программный стенд.

В рамках проводимого исследования *глагольных конструкций* (ГК) таджикского языка (ТЯ) [1], очередным этапом является разработка проблемно-ориентированного программного стенда на основе алгоритма распознавания таджикских глаголов [2]. Под глагольной конструкцией понимается семантически целостная (неделимая) последовательность словоформ, представляющая в предложении сказуемое – глагол (простой, сложно-именной, сложно-деепричастный и составной) или его сочетание с другими частями речи (конструкция).

Алгоритмы, реализуемые в разрабатываемом программном стенде и информационные базы к нему могут найти практическое применение в перспективных системах автоматической обработки текстов на таджикском языке, в том числе в системе таджикско-русского машинного перевода [3].

В результате ранее проведенных предварительных эмпирических исследований была произведена автоматизированная фильтрация словарных баз ТЯ с привлечением экспертов, в результате было выделено 746 простых глаголов таджикского языка в форме инфинитива, из которых 398 – производные отыменные глаголы. Из 348 непроизводных глаголов за счет исключения префиксов выделено 214 основ прошедшего времени [2].

На основе полученного результата сформирована база данных простых глаголов, которые условно разбиты на пять групп, табл. 1:

1. неправильные глаголы основы прошедшего времени (ОПВ);
2. неправильные глаголы основы настоящего времени (ОНВ);
3. правильные глаголы основы настоящего времени;
4. глаголы, образованные от имени существительного путем добавления глагольных постфиксов;
5. модальные глаголы.

Таблица 1. Пример условной классификации глаголов

Неправильные глаголы ОПВ	Неправильные глаголы ОНВ	Правильные глаголы ОНВ	Модальные глаголы
рафт	рав	бур	боист
пухт	паз	санҷ	шоист

На основе схемы глагольных парадигм ТЯ [1] получен исчерпывающий перечень глагольных постфиксов (всего 102 постфикса) и сформирована информационная база данных, в которой все постфиксы разбиты на четыре группы:

1. не присоединяемые к глаголу (соединимость с глаголом которых не установлена);
2. присоединяемые к основе глагола настоящего времени;
3. присоединяемые к основе глагола прошедшего времени;
4. присоединяемые к обеим основам глагола.

Также сформирована база данных префиксов: простых 13, сложных 35, всего 48 [2].

Так как простые глаголы таджикского языка состоят из одной основы (корня), к которой прибавляются известные префиксы и постфиксы, то их выявление обеспечивается применением процедуры POS-tagging (*частеречевого тэггера*), которая разбивает произвольную словоформу ТЯ на корни и аффиксы и ставит ей в соответствие информацию о части речи, представленную данной словоформой.

В ранее проведенных исследованиях в отношении глагольных конструкций ТЯ сформулированы следующие ограничительные утверждения, положенные в основу алгоритма распознавания ГК ТЯ [1, 2]:

– в составе ГК могут быть словоформы, представляющие простые глаголы, причастия и деепричастия, имена существительные, прилагательные, местоимения и предлоги;
– длина ГК ограничена шестью словоформами;
– словоформы ГК не могут быть разделены знаками пунктуации;
– предлог в ГК ТЯ может быть только первой словоформой;
– последней словоформой ГК может быть только простой глагол.

Чтобы добиться эффективной работы алгоритма распознавания глаголов [2] и повысить производительность экспериментального программного стенда для исследования глагольных конструкций таджикского языка, необходимо минимизировать количество обращений POS-tagger к базам данных. Для этого применяется сегментация предложений входного текста за счет учета знаков препинания, имен собственных, аббревиатур, наречий, союзов и частиц (неизменяемых частей речи ТЯ), оформленных в виде короткого стоп-словаря.

Экспериментальный программный стенд разрабатывается с использованием среды разработки Embarcadero® Delphi® 2010, информационные базы созданы на основе СУБД SQLite. Основное окно программного стенда содержит главное меню, панели инструментов и текстовые поля, отображающие входную и выходную информацию. Пункты главного меню и панелей инструментов обеспечивают операции файлового ввода-вывода, актуализации используемых баз данных (простых глагольных корней, префиксов и постфиксов) и основного функционала программы, рис. 1. Также стенд содержит строку состояния, в котором отображается информация о количестве глаголов, префиксов, постфиксов, ходе выполнения обработки и времени обработки.

Работа с программой осуществляется путем набивки или загрузки из файла исходных текстов ТЯ в первое текстовое поле («Шаг 1. Текст для обработки»). Здесь же доступны операции редактирования текста и подсчета количества слов в выделенном участке текста. После выполнения предварительных операций по подготовке входного текста, посредством пункта меню «Обработка» стенд выводит во второе текстовое поле («Шаг 2. Сегментация предложения и анализ») результаты

сегментации входного текста с учетом стоп-словаря и формальных правил сегментации. Далее к каждому сегменту последовательно применяется алгоритм распознавания глаголов ТЯ и в третье текстовое поле («Шаг 3. Выделение глагольных конструкций») выводятся только те сегменты, которые содержат в своем составе простые глаголы ТЯ.

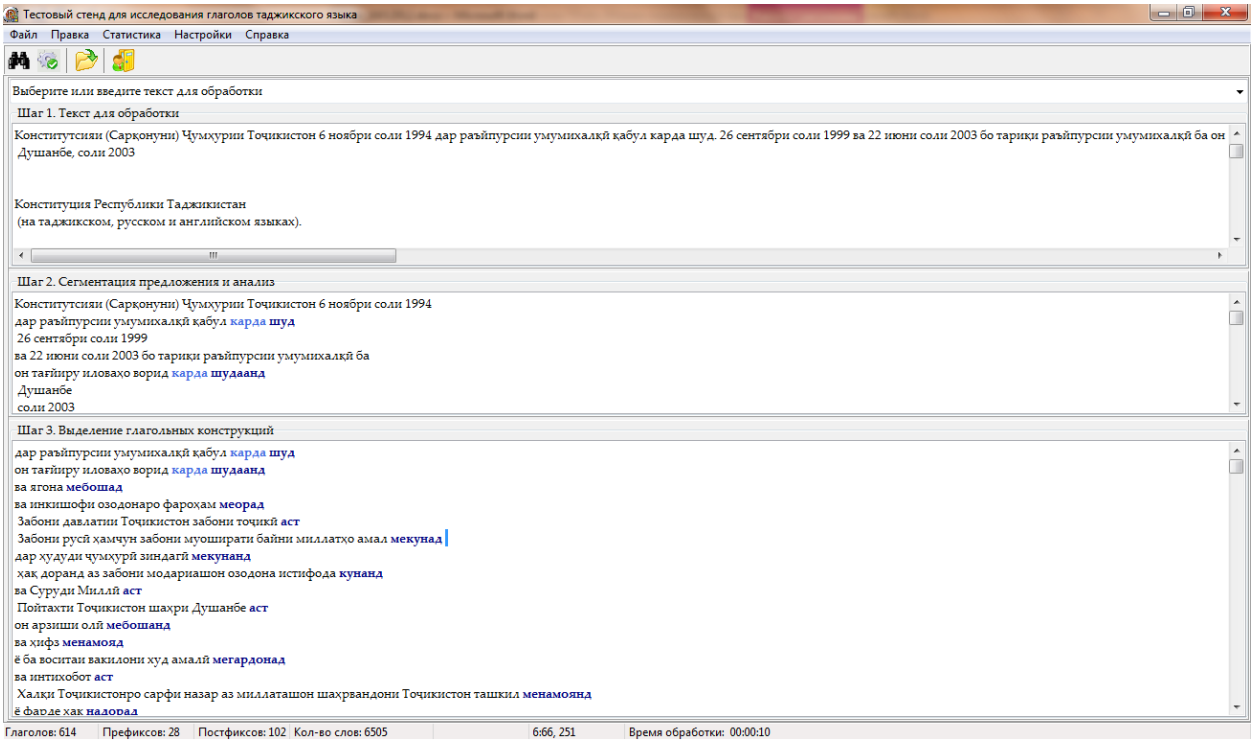


Рис. 1. Главное окно экспериментального программного стенда

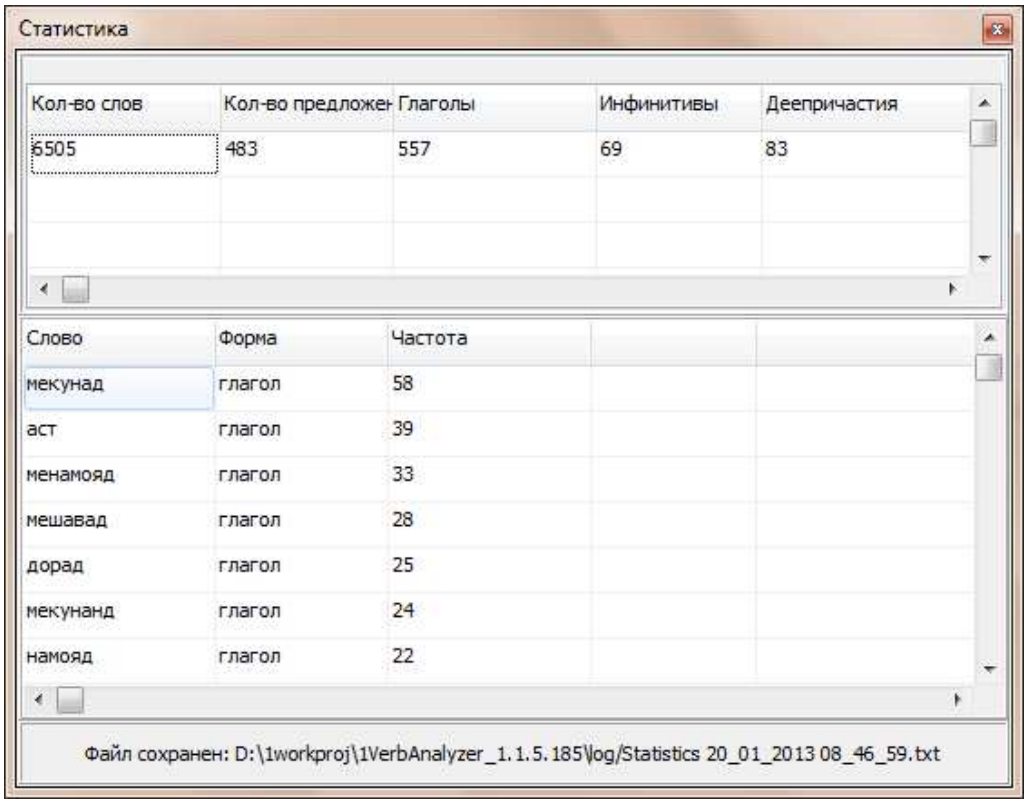


Рис. 2. Окно для отображения полученной статистики

После отработки основных процедур обработки текста становится доступна подробная статистика, рис. 2, которую можно также импортировать в лист MS Excel.

Таблица 2. Описание корпуса таджикских текстов

№	Наименование произведения	Жанр	Кол-во слов	Кол-во предложений	Размер файла (Кб)
1.	Тоҷикон	исторический	379782	43684	4872
2.	Нигоҳебатаърих	исторический	148401	8808	2060
3.	Тоҷикондар оинаитаърих аз Ориён то Сомониён	исторический	104682	5328	1397
4.	Куруши Кабир	исторический	39679	3125	513
5.	Китоби 15-солаги	исторический	18989	2455	916
6.	Шабисаввум	художественный	102162	9357	1244
7.	Садри Бухоро	художественный	116468	7575	1483
8.	Сорбон	художественный	58821	6409	727
9.	Хуросонаст ин ҷо	художественный	94235	4827	1188
10.	Суханварони Балх	художественный	61461	4024	770
11.	Акутагава	художественный	9594	493	114
12.	Қуръон	научный	139698	16013	1650
13.	Кодексимаъмурии ҶТ	нормативно-правовое	99558	4688	1411
14.	Кодексиграждании ҶТ	нормативно-правовой	57110	3517	806
15.	Кодексимеҳнати ҶТ	нормативно-правовой	31832	1482	449
16.	Конституцсияи ҶТ	нормативно-правовой	6505	483	91
17.	Назарияииқтисодӣ	учебный	153005	9679	2169
18.	Системаи Баланс	учебный	56189	4114	806
19.	Системаиидоракунӣ дар иқтисодиёт	учебный	53388	3114	734
20.	Асосҳои маркетинг	учебный	26127	1944	377
Итого			876128	57607	11629

Вспомогательное окно «Параметры» позволяет настроить опции работы программы, в частности, ведение журнала обработки (лог-файла), цветовую схему разметки текста (выделение цветом ГК и их элементов), форматы сохранения результатов и т.д.

Описываемый экспериментальный программный стенд находится в состоянии разработки, постоянно модернизируется и совершенствуется в соответствии с ходом

исследовательской работы, уточнения алгоритмов обработки текстов ТЯ и информационных баз.

Для обеспечения репрезентативной выборки исходного текстового материала для проведения исследований ГК ТЯ, был собран и описан специализированный корпус таджикских текстов, содержащий более 870 тыс. словоформ, в который вошли 20 произведений различных жанров, табл. 2. Тексты корпуса прошли предварительную обработку с целью удаления ошибок и приведения к однообразному виду, переведены в кодировку Юникод и сохранены в текстовом формате (*.txt).

Текущие возможности экспериментального стенда позволяют выделять в тексте простые глаголы, причастия (деепричастия) и инфинитивы (отглагольные имена). В результате обработки представленного корпуса ТЯ с помощью программного стенда были получены следующие промежуточные статистические данные, табл. 3.

Таблица 3. Промежуточные статистические результаты обработки корпуса ТЯ

№	Произведение	Глаголы	Инфинитивы	Причастия/ Деепричастия
1.	Тоҷикон	31268	2616	7520
2.	Нигоҳебатаърих	15132	864	3874
3.	Тоҷикондар оинаитаърих. Аз Ориён то Сомониён	8445	938	2708
4.	Куруши Кабир	4396	319	1202
5.	Китоби 15-солаги	1418	210	327
6.	Шабисаввум	12432	494	3664
7.	Садри Бухоро	13108	759	2301
8.	Сорбон	9072	470	2182
9.	Хуросонаст ин ҷо	10522	841	1449
10.	Суханварони Балх	6870	338	924
11.	Акутагава	1497	49	305
12.	Қуръон	23996	386	1495
13.	Кодексимаъмурии ҶТ	4253	3697	2040
14.	Кодексигражданиии ҶТ	4885	1265	1668
15.	Кодексимеҳнати ҶТ	2037	662	749
16.	Конституцсияи ҶТ	557	69	83
17.	Назарияиқтисодӣ	13033	1356	3807
18.	Системаи Баланс	3344	652	1432
19.	Системаиидоракунӣ дар иқтисодиёт	4198	701	1772
20.	Асосҳои маркетинг	2429	531	616
	Итого	126492	13737	28724

В частности, установлено, что простые глаголы составляют около 9% всех словоформ входных текстов, отглагольные имена – 2%, причастия (деепричастия) – 1%. Установлены частотные распределения встречаемости простых глаголов ТЯ для оптимизации информационных баз и сокращения временных затрат на работу основного алгоритма, рис. 3. Так, три глагола ТЯ («аст», «буд» и «мешавад») покрывают более 17,5% всех глаголов в исследованных текстах.

Предварительно полученные результаты дают возможность улучшить и модифицировать алгоритмы распознавания глагольных конструкций таджикского языка.

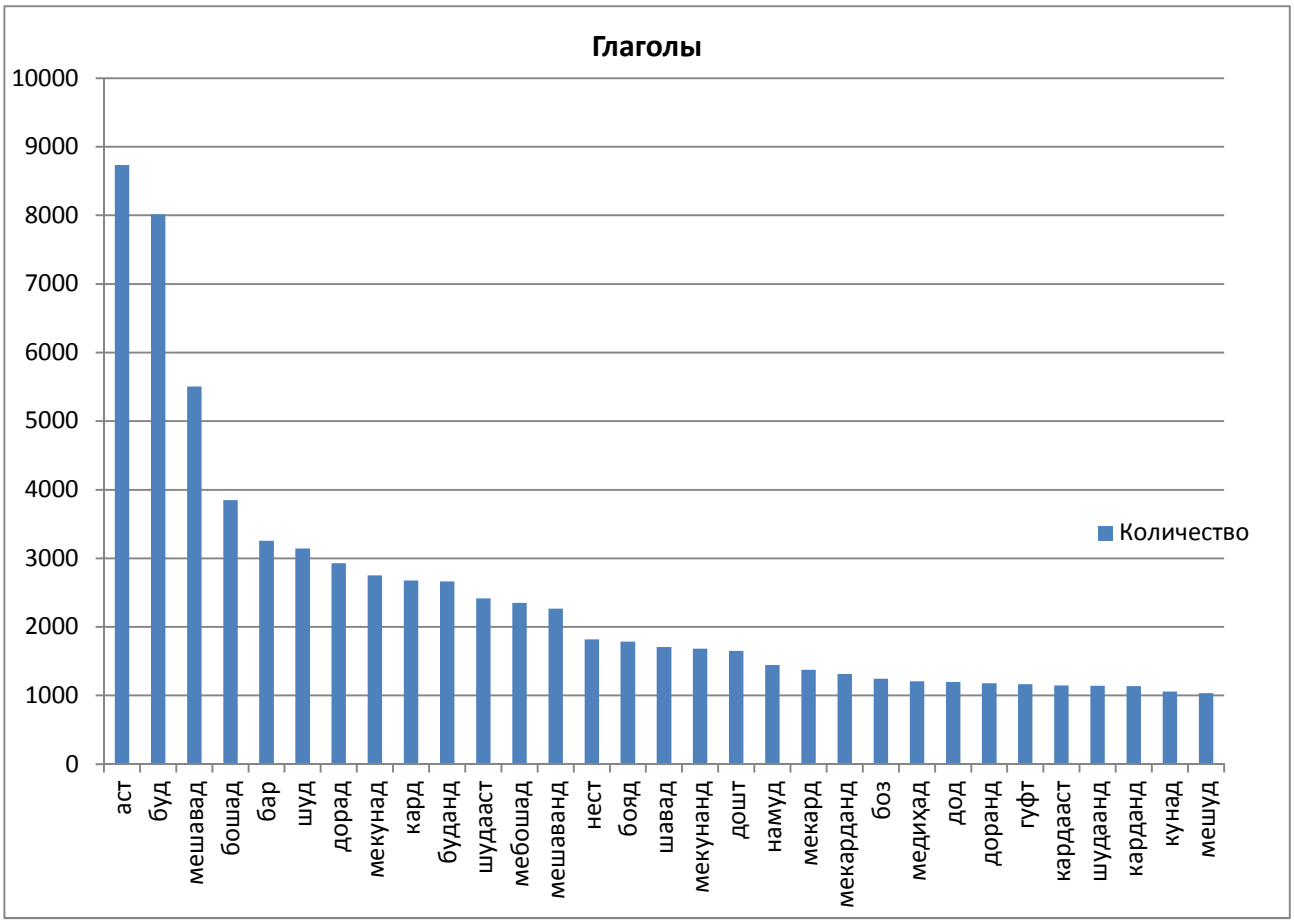


Рис. 3. Встречаемость различных глаголов в таджикских текстах

Список литературы

1. Собиров Д.Д, Гращенко Л.А., Усманов З.Д. Информационные основы автоматического распознавания глаголов таджикского языка // Известия Академии наук РеспубликиТаджикистан. Отд. физ.-мат., хим., геол. и техн. наук.- 2011. - №3. - С. 41-46.
2. Гращенко Л.А., Собиров Д.Д. Модели и алгоритм распознавания глаголов в предложениях таджикского языка // Доклады Академии наук Республики Таджикистан. - 2012. - том 55. - №2. - С. 120-125.
3. Гращенко Л.А., Клышинский Э.С., Тумковский С.Р., Усманов З.Д. Концептуальная модель системы русско-таджикского машинного перевода // Доклады Академии наук Республики Таджикистан. – 2011. - т. 54. - №4. - С. 279-285.

Комплекс средств автоматизации межъязыковых преобразований текстов для центрально-азиатского региона

Л.А. Гращенко
Институт математики Академии наук Республики Таджикистан
graschenko@mail.ru

Аннотация: В статье дается описание комплекса существующих и разрабатывающихся средств автоматизации для межъязыковых преобразований текстов между языками Центрально-азиатского региона по состоянию на конец 2012 года.

Ключевые слова: комплекс средств автоматизации; машинный перевод; межъязыковое преобразование текстов; таджикский язык; персидский язык; узбекский язык; язык пушту

Введение

В 2012 году активно продолжала свою работу школа по компьютерной лингвистике Таджикистана под руководством академика АН РТ Зафара Джураевича Усманова (www.complingv.tj). Деятельность ряда исследователей из состава школы была направлена на развитие комплекса средств автоматизации (КСА) межъязыковых преобразований текстов для языков Центрально-азиатского региона (ЦАР), в соответствии с ранее предложенной стратегией [1]. Работа проходила по шести основным направлениям, рис. 1:

1. Разработка и совершенствование средств конверсии графических систем письма между фарси (Fa), дари (Da) и таджикским (Tj) языками;
2. Разработка и совершенствование средств внутриязыкового согласования текстов для таджикского языка;
3. Разработка системы машинного перевода (СМП) с языков фарси-дари (Fa-Da) на русский язык (Ru);
4. Исследование межморфемных отношений узбекского (Uz) и таджикского языка (Tj);
5. Исследование афганского языка (пушту, Af) и фарси (Fa);
6. Вспомогательные исследования в рамках разработки средств машинного перевода с таджикского на русский язык.

1-3 направления находятся в стадии разработки и совершенствования программных продуктов, а по 4-6 направлениям исследования носят подготовительный и перспективный характер. Ниже дадим характеристику текущих достижений в рамках каждого из перечисленных направлений.

Средства конверсии графических систем письма (Fa-Da ↔ Tj).

В рамках настоящего направления продолжалось развитие функционала таджикско-персидского конвертера графических систем письма и начата разработка обратного конвертера. К исходу 2012 года прототип персидско-таджикского конвертера обеспечивает конверсию с точностью порядка 65-75% на широкой выборке текстов, рис. 2. Объемы информационных баз составляют соответственно: корней - 700, стоп-слов - 72, префиксов - 146, постфиксов - 516.

Для повышения точности конверсии производится наполнение базы корней, а также совместно с А.Ю. Фоминым исследуются вопросы распознавания в персидском тексте изафетных конструкций [2].

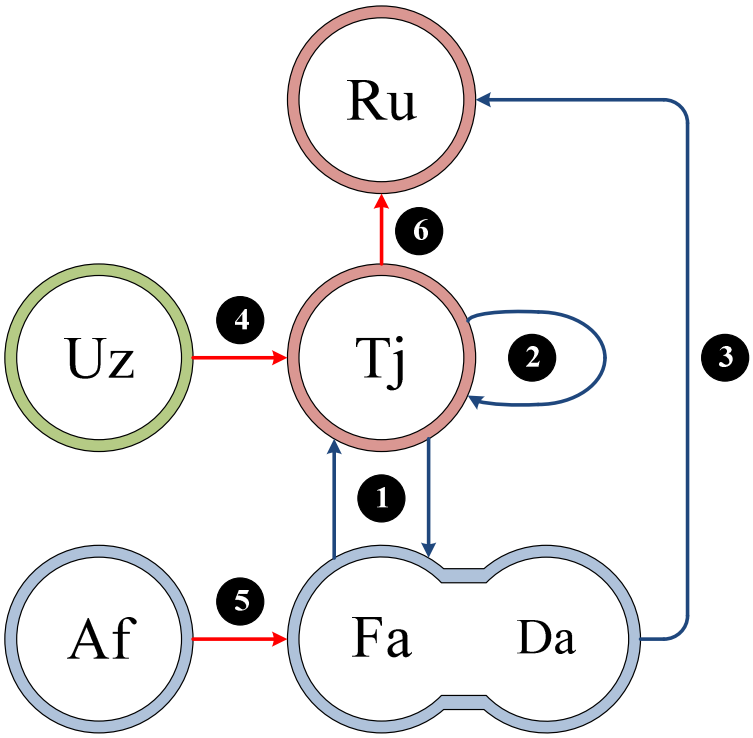


Рис. 1. Схема проводимых исследований по разработке КСА межъязыковых преобразований текстов для языков ЦАР

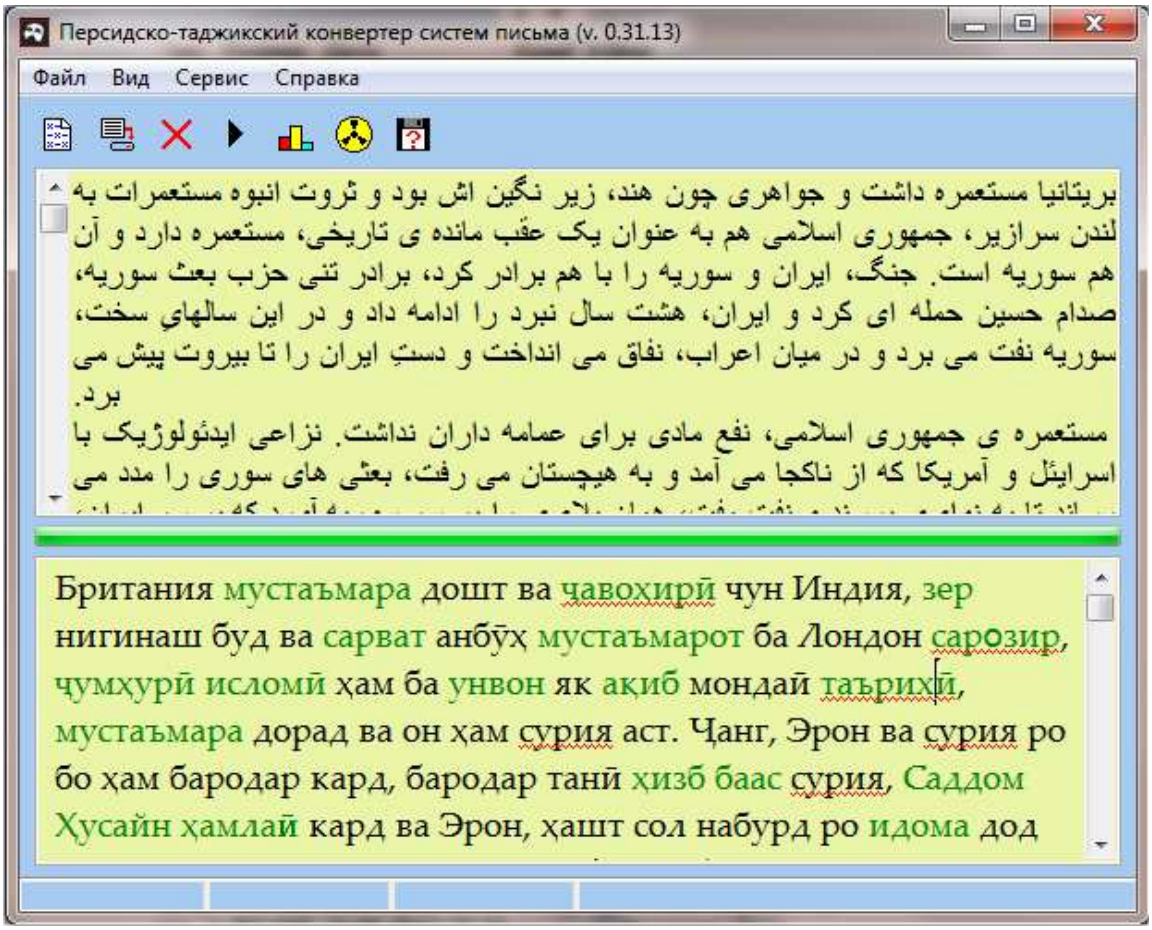


Рисунок 2. Экранная форма главного окна персо-таджикского конвертера систем письма

Инструментарий внутриязыкового согласования текстов (Tj ↔ Tj).

На основе ранее выполненных разработок мастера согласования таджикских шрифтов и автоматического корректора специфических букв таджикского языка, К.С. Бахтеевым начаты исследования по созданию унифицированного символьного препроцессора таджикских текстов. В настоящее время исследована проблематика предметной области и сформулирована последовательность этапов решения данной научной задачи. В результате должен быть создан отдельный модуль, реализующий начальный этап работы любой системы автоматической обработки текстов на таджикском языке, предшествующий этапу графематического анализа.

Система машинного перевода «Папирус» (Fa-Da → Ru).

Начата разработка системы машинного перевода «Папирус», рис. 3. К настоящему времени разработана и апробирована система графематического анализа персидского языка, набрана словарная база порядка 40 тыс. словоформ и база памяти переводов - около 9,5 тыс. выражений. Производится адаптация системы морфологического анализа, реализованная в персидско-таджикском конвертере систем письма, для реализации в данной СМП.

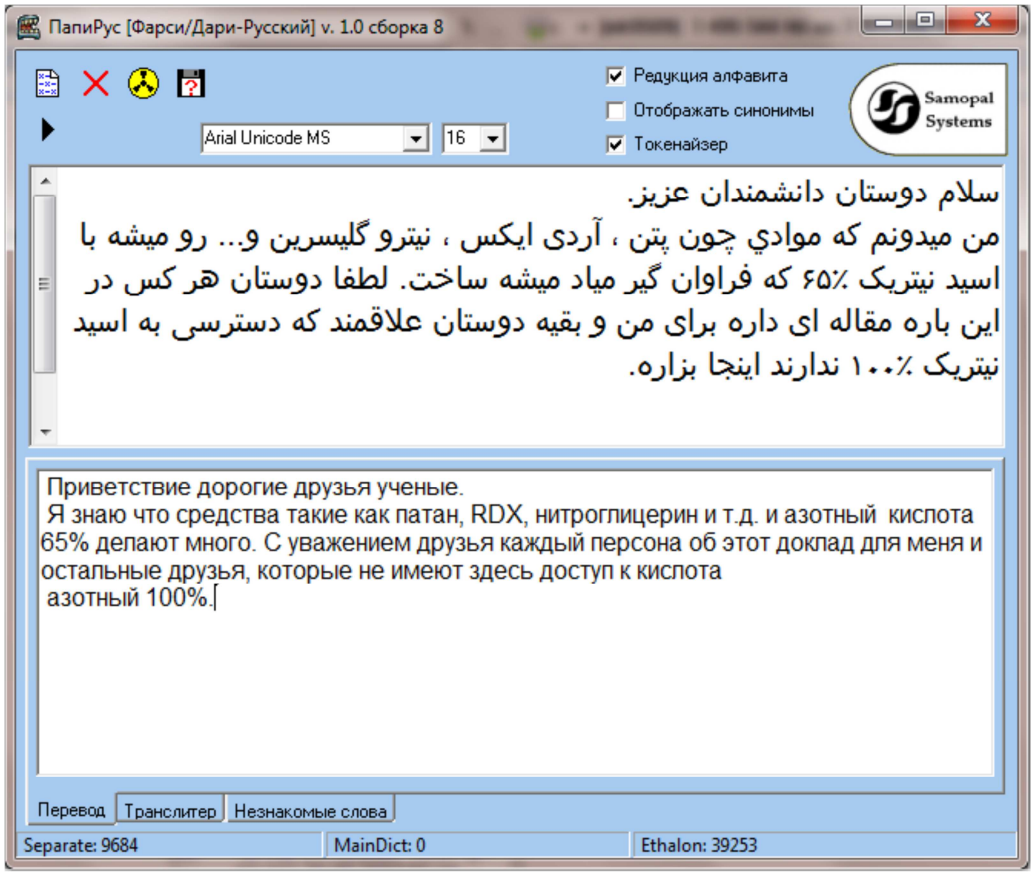


Рис. 3. Экранная форма главного окна СМП «Папирус»

Разработка морфемного преобразователя текстов (Uz → Tj).

Таджикский и узбекский языки, не являясь близкими, развивавшись в одном регионе в условиях конвергентной эволюции приобрели определенное сходство в части базовой лексики, в части общих арабских заимствований, во взаимопроникновении части лексики друг в друга, в количестве и сопоставимости грамматических категорий, в морфологическом строе. С учетом данного наблюдения

начата разработка средств автоматического установления межморфемных соответствий данной языковой пары для последующей реализации межъязыкового преобразователя.

Разработка морфемного преобразователя текстов (Af → Fa-Da).

А.Ю. Фоминым начаты исследования в направлении установления взаимосвязей персидско-таджикского языкового кластера и афганского языка (пушту) на предмет применимости методов языковой трансформации. Точкой соприкосновения указанных языков является пласт заимствованной арабской лексики, составляющая от 35 до 45% словарного фонда.

Разработка средств машинного перевода (Tj → Ru).

В рамках реализации СМП с таджикского языка на русский язык в соответствии с ранее представленной концепцией [3], Д.Д. Собировым проводились вспомогательные исследования автоматического распознавания глагольных конструкций таджикского зыка. Получена предварительная статистика распределения различных глаголов и элементов глагольных конструкций по таджикским текстам.

Список литературы

1. Гращенко Л.А. Анализ состояния и перспектив развития систем машинного перевода для стран Центральной Азии и Кавказа // Новые информационные технологии в автоматизированных системах: материалы 14 научно-практического семинара. - М.: МГИЭМ, 2011. - С. 92-106.

2. Гращенко Л.А. Обзор проблематики персидско-таджикской конверсии графических систем письма // Материалы международной научно-практической конференции «Подготовка научных кадров и специалистов новой формации в свете инновационного развития государств». - Душанбе: «Ирфон», 2010. - С. 363 - 365.

3. Гращенко Л.А., Клышинский Э.С., Тумковский С.Р., Усманов З.Д. Концептуальная модель систему русско-таджикского машинного перевода // Доклады АН РТ – том 54, №4 – 2011 г. – С. 279-285.

О множестве анаграмм и распознавании их элементов

З.Д.Усманов, В.Нормантас
Институт математики АН Республики Таджикистан
E-mail: zafar-usmanov@rambler.ru

Аннотация: Рассмотрены три способа кодирования слов. Выявлены их статистические закономерности и установлены мощности множеств анаграмм в текстовых корпусах английского, литовского, русского и таджикского языков, а также искусственного языка эсперанто. Предложены три модифицированных способа кодирования, реализующие с близкими к единице относительными частотами взаимно однозначные отображения между словами и их кодами.

Ключевые слова: слово - кодирование – анаграмма – распознавание - декодирование – статистика

Пусть L - какой-либо естественный язык с алфавитом A и $W = "\alpha_1\alpha_2 \ldots\alpha_n"$ - некоторое его слово длины n, состоящее из букв $\alpha_k \in A$, $k = \overline{1, n}$. Введём в рассмотрение цепочку $CW = "\alpha_{s1}\alpha_{s2} \ldots\alpha_{sn}"$, составленную из тех же самых букв, что и в W, но упорядоченных по алфавиту.

1. Как и в [1], рассмотрим следующий тип кодирования (отображения) слова W.

Определение 1. *Отображение* $F : W \to CW$ *назовём упорядоченным алфавитным кодированием (*$\alpha\beta$*- кодированием) слова* W*, а цепочку букв* CW *− его* $\alpha\beta$*- кодом..*

ПРИМЕР: $F : W = сорт \to CW = орст$.

Отображение F каждому слову W ставит в соответствие единственный образ CW, в то время как обратное отображение (декодирование) является, вообще говоря, не однозначным. Действительно, для цепочки $CW = орст$ прообразом является множество из четырех слов $\{W : W = рост, сорт, трос, торс\}$.

В настоящей работе конечное множество слов, составленных из одного и того же набора букв, будем называть *анаграммой*, а сами слова – *элементами* анаграмм[1].

Всякой анаграмме соответствует единственный образ, $\alpha\beta$- код. В свою очередь, для обратного отображения $F^{-1} : CW \to W$ образу всякой анаграммы сопоставляется не менее двух прообразов на множестве $\{W\}$, то есть декодирование образов анаграмм не однозначно.

Для того чтобы получить представление об “эффективности” $\alpha\beta$- кодирования, выражающейся в способности распознавания прообразов $\alpha\beta$- кодов, необходимо, как это сформулировано в [1], оценить мощность множества анаграмм в естественных языках. В работе [2] эта задача рассмотрена на примере английского, литовского, русского и таджикского языков, а также искусственного языка эсперанто. На основе обработки данных текстовых корпусов установлено, что относительные частоты встречаемости элементов анаграмм для отмеченных языков группируются вокруг

[1] Предложенное определение отличается от общепринятых. Так в большом энциклопедическом словаре под анаграммой понимается слово или словосочетание, образованное перестановкой букв другого слова или словосочетания, а в толковом словаре русского языка Д.Н.Ушакова анаграмма – это перестановка букв, посредством которой из одного слова составляется другое.

значения 0.5. Иначе говоря, почти каждое второе слово из корпуса текстов принадлежит множеству анаграмм. И несмотря на то, что $\alpha\beta$ - кодирование текстовых корпусов оказывается крайне удобным средством для выявления всевозможных элементов анаграмм, оно представляется неэффективным для декодирования.

2. В связи с этим рассмотрим две модификации F, обозначаемые через $F^{(f)}$ и $F^{(f,l)}$, [1]. Также, как и F, они задаются на множестве $\{W\}$ слов естественного языка L. Отображение $F^{(f)}$ наделяется следующими свойствами.

Определение 2. *Отображение $F^{(f)}$ слову W ставит в соответствие цепочку $\alpha_1 C(W/\alpha_1)$, в которой α_1 - первая буква в слове W и $C(W/\alpha_1)$ − $\alpha\beta$ - код цепочки W/α_1, т.е. слова W без первой буквы.*

Это отображение, в отличие от F, оставляет в слове W неизменной первую букву, т.е. α_1, и упорядочивает по алфавиту прочие буквы. Из общих соображений ясно, что декодирование $\alpha_1 C(W/\alpha_1) \to W$ в определенном смысле обладает лучшими свойствами, чем $CW \to W$.

Ещё один модифицированный способ кодирования представляется следующим образом.

Определение 3. $F^{(f,l)} : W \to \alpha_1 C(W/\{\alpha_1, \alpha_n\})\alpha_n$.

В нём α_1 - первая и α_n - последняя буквы слова W остаются неподвижными, а цепочка букв между ними, т.е. $W/\{\alpha_1, \alpha_n\}$, подвергается $\alpha\beta$ - кодированию.

ПРИМЕР. Обратимся к анаграмме $\{W$: W = *автор, втора, отвар, рвота, тавро, товар*$\}$. Отображение $F^{(f)}$ первые четыре элемента кодирует следующим образом − ***а**ворт, **в**аорт, **о**аврт, **р**авот*, оставляя неизменными первые буквы элементов анаграммы (отмечены жирным цветом) и располагая в алфавитном порядке прочие буквы. Этим кодам однозначно соответствуют первые четыре элемента анаграммы. Пятый и шестой элементы анаграммы кодируются одинаково – ***т**авро*.

Теперь рассмотрим применение отображение $F^{(f,l)}$ к той же анаграмме. В этом случае первая и последняя буквы (далее показаны жирными буквами) элементов анаграмм должны оставаться неизменными, а все другие буквы упорядочиваются по алфавиту. Результаты кодирования записываются в виде: ***а**вот**р**, **в**орт**а**, **о**авт**р**, **р**вот**а**, **т**авр**о**, **т**ав**о**р,* то есть все шесть слов рассматриваемой анаграммы получили собственные коды. Декодирование с помощью обратной функции также однозначно.

Итак, в сравнении с F отображения $F^{(f)}$ и $F^{(f,l)}$ несколько сложнее, зато наверняка успешнее в вопросах декодирования.

3. В таблице 1 приводятся результаты обработки статистической информации, характеризующие предварительные свойства введённых отображений. В столбце 1 дан перечень языков, которые явились предметом исследований (En – английский, Lt – литовский, Ru – русский, Tj – таджикский, Eo – эсперанто). В столбце 2 показаны размеры (в словах) корпусов текстов, подвергнутых обработке.

Обработка корпусной информации начиналась с построения *частотных словарей* - списка различных слов (без дубликатов) с частотами их встречаемости. Из этих данных в столбец 3 извлечены только числа различных слов.

Далее слова из частотного словаря кодировались тремя способами (столбец 4). Определяемое при этом число различных кодов заносилось в соответствующую строку столбца 5. На этом этапе для каждого языка были сформированы 3 база различных слов с их кодами.

Таблица 1

1. Язык	**2.** Размер корпуса (в словах)	**3.** Число различных слов	**4.** Тип кода	**5.** Число различных кодов	**6.** Число однозначных кодов	**7.** Число двусмысленных кодов	**8.** Отношение (6)/(5) в %	**9.** Отношение (7)/(5) в %
En	11252496	137732	F	119055	106841	12214	89.74	10.26
			$F^{(f)}$	130644	124366	6278	95.19	4.81
			$F^{(f,l)}$	135618	133570	2048	98.49	1.51
Lt	34165084	693995	F	605039	546254	58785	90.28	9.72
			$F^{(f)}$	654475	621487	32988	94.96	5.04
			$F^{(f,l)}$	675208	657925	17283	97.44	2.56
Ru	19175074	509031	F	462886	430517	32369	93.01	6.99
			$F^{(f)}$	488286	470336	17950	96.32	3.68
			$F^{(f,l)}$	500433	492360	8073	98.39	1.61
Tj	2323965	87181	F	80080	74512	5568	93.05	6.95
			$F^{(f)}$	84220	81501	2719	96.77	3.23
			$F^{(f,l)}$	85805	84499	1306	98.48	1.52
Eo	5080195	165570	F	147220	133851	13369	90.92	9.08
			$F^{(f)}$	158310	151885	6425	95.94	4.06
			$F^{(f,l)}$	162940	160407	2533	98.45	1.55

На следующем этапе различные коды разделялись на две группы. Одну из них составили однозначно декодируемые коды, число которых указано в столбце 6, а другую - коды анаграмм, названные двусмысленными (им соответствуют несколько прообразов), число которых записано в столбце 7. Следует оговорить, что для отображения F речь идет об анаграммах в смысле приведенного определения, а для отображений $F^{(f)}$ и $F^{(f,l)}$ – соответствующим образом модернизированных анаграммах.

В столбцах 8 и 9 данные столбцов 6 и 7 выражены в процентах по отношению к общему количеству различных кодов столбца 5. Из этих данных видно, что для рассматриваемых языков количество различных однозначно декодируемых кодов на порядок больше, чем суммарное количество различных кодов анаграмм. Отметим, что здесь имеется в виду перечень слов частотных словарей, причём без учёта частот их встречаемости. *Полученный результат*, отдавая приоритет однозначно декодируемым кодам, *создаёт, однако, искажённую картину о мощности множества слов, входящих в состав анаграмм*. Дальнейшее исследование, результаты которого представлены в таблице 2, подтверждает сделанный вывод.

Прежде чем переходить к рассмотрению этой таблицы, объясним смысл обозначений $\overline{F}$, $\overline{F}^{(f)}$ и $\overline{F}^{(f,l)}$.

4. Как отмечалось в п.п.1 и 2, отображения, $F^{(f)}$ и $F^{(f,l)}$ каждому слову приписывают единственный код, однако обратные отображения, в общем случае, не обеспечивают однозначного декодирования. Использование отображений $\overline{F}$, $\overline{F}^{(f)}$ и $\overline{F}^{(f,l)}$ - это, по существу, попытка устранения неоднозначности при декодировании анаграмм и распознавания порождающих их прообразов за счет использования дополнительных атрибутов, присоединяемых к $\alpha\beta$- кодированию.

Определение 4. *Отображения* $\overline{F}$, $\overline{F}^{(f)}$ *и* $\overline{F}^{(f,l)}$ *обладают следующими свойствами:*

− *они задаются на множестве* слов $\{W\}$ языка L**;**

− *совпадают, соответственно, с* F, $F^{(f)}$ и $F^{(f,l)}$ *при кодировании слов;*

− *их обратные отображения* $\left(\overline{F}\right)^{-1}$, $\left(\overline{F}^{(f)}\right)^{-1}$ *и* $\left(\overline{F}^{(f,l)}\right)^{-1}$ *на кодах, однозначно декодируемых, совпадают, соответственно, с* F^{-1}, $\left(F^{(f)}\right)^{-1}$, $\left(F^{(f,l)}\right)^{-1}$, *а на кодах "анаграмм" каждому из них ставят в соответствие единственное слово* W^*, *которое имеет максимальную частоту встречаемости в текстах в сравнении с другими словами с одинаковым кодом.*

ПРИМЕР. Положим, что в анаграмме $\{W:\ W =$ *казан, казна, наказ*$\}$ наибольшую частоту в корпусе текстов имеет слово *наказ,* затем *казна.* При отображении F ($= \overline{F}$) рассматриваемой анаграмме будет соответствовать код *аазкн,* которому отображение $\left(\overline{F}\right)^{-1}$ поставит в соответствие слово *наказ.*

Если же применить отображение $\overline{F}^{(f)}$($= F^{(f)}$), то первые два слова анаграммы получат одинаковый код ***к**аазн,* а третье слово – *код* ***н**аазк* (напомним, что при кодировании первые буквы в словах анаграмм фиксируются). Коду ***к**аазн* будет сопоставляться слово *казна,* у которого частота больше, чем у слово *казан.*

ЗАМЕЧАНИЕ. Предлагаемый в определении 4 метод распознавания единственного прообраза того или иного кода анаграммы носит вероятностный характер. Он не исключает возможности принятия ошибочного решения в случаях, когда при правильном декодировании на выходе должно появиться слово не с максимальной частотой. Статистические значения величин ошибок определяются для каждого из рассматриваемых языков путем соответствующей обработки текстовых корпусов.

5. Теперь перейдем к обсуждению результатов, представленных в таблице 2. В ней первые два столбца – те же, что и в таблице1. Третий столбец отмечает 3 типа используемых способов кодирования слов.

Относительно столбцов 4 – 7 следует сказать, что они по своей сути являются продолжением таблицы 1. С учётом того, что согласно определению 4 при кодировании слов отображения F и $\overline{F}$, $F^{(f)}$ и $\overline{F}^{(f)}$, $F^{(f,l)}$ и $\overline{F}^{(f,l)}$ совпадают, эти столбцы представляют информацию о частотах встречаемости однозначно декодируемых кодов и элементов анаграмм (двусмысленных кодов), причём столбцы 4 и 6 информируют об абсолютных значениях частот, а 5 и 7 – об относительных значениях, выраженных в процентах по отношению к общему количеству слов текстовых корпусов.

Из данных столбцов 4 – 7 и строк, привязанных к отображению $\overline{F}$, видно, что

для всех языков, за исключением русского, слова, являющиеся элементами анаграмм, составляют более половины общего количества слов текстовых корпусов (для русского языка - 45.69 %).

Итак, несмотря на то, что количество различных однозначно декодируемых слов оказалось на порядок больше количества различных слов, входящих состав анаграмм (см. данные столбцов 8 и 9 для строк F - отображения), частоты встречаемости рассматриваемых элементов в тестовых корпусах пяти языков оказались одного порядка.

Таблица 2

1. Язык	2. Размер корпуса (в словах)	3. Тип кода	4. Частоты однознач - ных кодов	5. Отно-шение (4)/(2) в %	6. Частоты двусмы-сленных кодов	7. Отно-шение (6)/(2) в %	8. Частоты слов, декоди-руемых по опр.4	9. Отно-шение (8)/(3) в %
Eng	11252496	$\overline{F}$	4738825	42.11	6513671	57.89	10961792	97.42
		$\overline{F}^{(f)}$	8282101	73.60	2970395	26.40	11179579	99.35
		$\overline{F}^{(f,l)}$	10830821	96.25	421675	3.75	11224361	99.75
Lt	34165084	$\overline{F}$	15637702	45.77	18527382	54.23	33199335	97.17
		$\overline{F}^{(f)}$	23737620	69.48	10427464	30.52	33834328	99.03
		$\overline{F}^{(f,l)}$	28999545	84.88	5165539	15.12	34028900	99.60
Ru	19175074	$\overline{F}$	10413280	54.31	8761794	45.69	18724384	97.65
		$\overline{F}^{(f)}$	14490319	75.57	4684755	24.43	19063618	99.42
		$\overline{F}^{(f,l)}$	16449834	85.79	2725240	14.21	19145587	99.85
Tj	2323965	$\overline{F}$	1152420	49.59	1171545	50.41	2280334	98.12
		$\overline{F}^{(f)}$	1759262	75.70	564703	24.30	2309336	99.37
		$\overline{F}^{(f,l)}$	2021532	86.99	302433	13.01	2316348	99.67
Eo	5080195	$\overline{F}$	1788681	35.21	3291514	64.79	4782380	94.14
		$\overline{F}^{(f)}$	4185715	82.39	894480	17.61	5033967	99.09
		$\overline{F}^{(f,l)}$	4834326	95.16	245869	4.84	5068691	99.77

Что касается данных столбцов 4 – 7 и строк, соответствующих отображениям $\overline{F}^{(f)}$ *и* $\overline{F}^{(f,l)}$, то в этом случае обнаруживается заметное превосходство частоты встречаемости однозначно декодируемых слов. А для английского языка и эсперанто

отображение $\overline{F}^{(f,l)}$ характеризуется особо высокими показателями декодирования, соответственно 96.25 % и 95.16 %.

Последние два столбца, 8 –й и 9-й, таблицы 2 выдают количественные показатели эффективности декодирования в соответствии с определением 4. Для всех пяти языков ошибки принятия неверных решений заключаются в пределах 1% при декодировании посредством $\left(\overline{F}^{(f)}\right)^{-1}$ и $\left(\overline{F}^{(f,l)}\right)^{-1}$ и не превосходит 3 % при декодировании с помощью $\overline{F}$ (для эсперанто - не более 6 %).

Отметим, что интерес к F - отображению, равно как и его модификациям $F^{(f)}$ и $F^{(f,l)}$, объясняется тем, что на множестве их цепочек-образов в сравнении с исходным множеством слов $\{W\}$ решение ряда задач обработки текстовой информации заметно упрощается.

Область исследований настоящей статьи граничит с работами [3] и [4], в которых обсуждается утверждение Г.Роулинсона (G.E.Rawlinson) о том, что случайное расположение букв в “середине” слов либо слабо влияет, либо не влияет совсем на способность квалифицированного читателя понимать текст.

Список литературы

1. Усманов З.Д. Об упорядоченном алфавитном кодировании слов естественных языков // ДАН Республики Таджикистан, 2012, т.55, № 7, с.

2. Усманов З.Д., Нормантас В. Статистические свойства $\alpha\beta$-кодирования слов естественных языков // ДАН Республики Таджикистан, 2012, т.55, № 8, с.

3. Rawlinson, G. E. (1976) The significance of letter position in word recognition. Unpublished PhD Thesis, Psychology Department, University of Nottingham, Nottingham UK.

4. Davis, M (2003) http://www.mrc-cbu.cam.ac.uk/people/matt.davis/Cmabrigde/

Создание «онтологии всего». Проблемы классификации и решения

Г.А. Волкова
МИЭМ НИУ ВШЭ, Кафедра ИТАС

Аннотация. В работе рассматриваются основные проблемы, возникающие при построении онтологий слабо формализуемых предметных областей и онтологий общего назначения. Обобщая мировой опыт создания онтологий общего назначения, можно выделить общие классификационные признаки, по которым должна производиться систематизация знаний о реальном мире. Наряду с ними, предлагается подход к формированию универсальной системы классификации понятий, основанный на анализе результатов синтаксического анализа больших корпусов текстов на естественном языке.

«Сложные онтологии»

Согласно общепринятому определению «предметной области» [14], под термином понимается совокупность всех предметов, свойства которых и отношения между которыми рассматриваются в научной теории. При наличии некой научной теории мы можем получить перечень понятий, которые являются ключевыми для некоторой предметной области. Для этого можно воспользоваться уже существующими терминологическими словарями, глоссариями, тезаурусами и проч. Таким образом, основываясь на существующей терминологии и отношениях, которые определены для рассматриваемой предметной области, при построении онтологии предметных областей задача классификации понятий частично решена еще на начальном этапе ее создания. Иначе дела обстоят с онтологиями верхнего уровня, которые содержат наиболее общие понятия всего реально мира, не относящиеся к строго ограниченному домену.

Стоит отметь, что термин «предметная область» для онтологии носит весьма условный характер. Существует принципиальная разница, например, между предметными областями «химические элементы» и «государственные закупки». В первом случае мы имеем дело с хорошо формализованной областью знаний: 118 химических элементов по состоянию на 2012 год, расположенных по определенным правилам в таблице Менделеева. Во втором – с совокупностью всех товаров и услуг, за которые платит государство. По сути, предметная область для онтологии – это способ выделения из всего реального мира только тех понятий и отношений, которые необходимы для решения некой задачи. Ограничение «то, что продается» относится скорее к свойству сущности, входящий в некоторый концепт, чем к самостоятельному классификационному признаку, и не может должным образом ограничить сущности реального мира. В связи с этим, онтология химических элементов, безусловно, - онтология предметной области с явным ограничением сущностей реального мира. В случае онтологии государственных закупок задача сводится в первую очередь не к созданию обширной таксономии понятий, а к построению онтологии верхнего уровня, которая будет представлять собой систему классификации понятий. И, пожалуй, самую обширную предметную область (домен) описывают «онтологии здравого смысла», целью создания которых служит концептуализация знаний обо всем реальном мире без каких либо ограничений.

Очевидно, что сложность построения онтологии в зависимости от домена заключается не в том, сколько понятий предметной области следует связать посредством таксономии, а насколько хорошо формализована эта предметная

область, легко ли будет определить, относится ли некоторое понятие к интересующей предметной области и целесообразно ли его внесение в создаваемую онтологию. Границы предметной области «химические элементы» вполне конкретны, и «детские качели» к ней не относятся. В случае государственных закупок вопрос не настолько тривиален, т.к. ясно, что качели можно купить, но неизвестно, покупает ли их государство в текущий период времени и будет ли покупать в дальнейшем.

Мы подошли к следующей проблеме, возникающей при проектировании онтологий слабо формализованных предметных областей, а именно, необходимой функциональной полноте и минимальности создаваемой онтологии. Для предметной области с четко определенными границами возможно построение полноценной онтологии верхнего уровня, поскольку уже на этапе создания известно, как будет развиваться данная предметная область, поэтому необходимо предусмотреть только возможность вертикального расширения онтологии, т.е. пополнение ее подчиненными концептами и экземплярами. В случае государственных закупок, или любой другой предметной области, для которой сложно предсказать расширение, целесообразно создание наиболее универсальной онтологии верхнего уровня, которая дала бы возможность помимо вертикального расширения осуществлять также и горизонтальное, т.е. необходимо предусмотреть возможность расширения самой предметной области, а не пополнение существующей новыми сущностями.

Для многих предметных областей актуально создание аппарата автоматического пополнения онтологии из открытых источников информации. В зависимости от условий дальнейшего использования онтологии это может быть Интернет, или же внутренний репозиторий документов предприятия. В случае пополнения из Интернет актуально создание многоязычной онтологии, ориентированной на представление знаний на нескольких языках. Сложность создания таких онтологий заключается в различии понятийных систем естественных языков, с которыми предполагается работать.[19]

Помимо характеристики предметной области, онтологии могут отличаться по количеству используемых элементов и типов отношений. Так, Э. Хови называет «терминологическими» онтологии, включающие сущности, явления, свойства, связи предметной области и объединяющие их структурные отношения, а «настоящими онтологиями» только онтологии, содержащие дефиниционные отношения, отношения дополнительной информации, взаимосвязи между отношениями и аксиомы [5]. В зависимости от количества включаемых элементов трудоемкость создания онтологий для одной и той же предметной области может разительно отличатся.

Проблемы построения онтологий общего назначения

Термин «онтология верхнего уровня» может трактоваться по-разному: зачастую под ним понимается некая онтология, содержащая наиболее общие понятия предметной области и создаваемая на первоначальном этапе для их классификации. Так, в своей методологии построения онтологий Усколд и Кинг указывают на необходимость создания метаонтологии для описания таких фундаментальных концептов, как сущности, отношения и исполнители [12]. Методология создавалась для построения бизнес-онтологий, в связи с чем не может быть универсальной. Так, введение метапонятия «исполнитель» оправдано только для бизнес-онтологий, поскольку связано с теорией активностей [17].

Существует и другой подход, при котором под «онтологией верхнего уровня» понимается систематизация знаний о реальном мире безотносительно к конкретного

домену. Такие онтологии называют «онтологиями общего назначения» или «здравого смысла». Возвращаясь к «онтологии госзакупок», следует отметить, что по сути такая онтология может быть также отнесена к онтологиям общего назначения, поскольку границы предметной области слабо определены.

При построении онтологий общего назначения существует несколько основных проблем, связанных с ее предполагаемой обширностью.

- Не существует очевидного способа классификации понятий реального мира и выделения базисных понятий. К тому же, любая классификация, в силу того, что она определяется только лишь мнением разработчиков онтологии, может быть оспорена. Наиболее распространенным подходом к классификации понятий в онтологиях верхнего уровня является использование философского понятийного аппарата, а именно метафизики [15]. Данный факт обусловлен относительной универсальностью философских категорий и их общепринятостью, что гарантирует онтологии, как минимум, критерий ясности. Но и такой подход ставится под сомнение [10].
- Не существует объективных способов оценки создаваемой онтологии. Поскольку для онтологии общего назначения основным критерием является ее универсальность, классические приемы оценки полноты, такие как формирование «вопросов компетенции» [16], являются задачей соизмеримой по сложности с построением «онтологии всего, что есть на свете».
- Нет общепринятой точки зрения на то, какова может быть глубина создаваемой онтологии, т.е. нет критерия, по которому некоторое понятие может присутствовать в онтологии общего назначения, а некоторое уже должно быть отнесено к онтологии определенной предметной области. Например, пирамида знаний TOVE [3] состоит из уровня общих знаний (активности, процессы, ресурсы, время, причины) и уровня, специфичного для бизнеса (цена, качество, оргструктура и др.). В этой связи неясно, можно ли считать всю пирамиду TOVE онтологией общего назначения, или же оправдано называть ею только уровень общих знаний.
- К онтологиям общего назначения не применим экспертный подход (ad hoc), т.к. не существует критериев, по которым можно было бы выделить «экспертов здравого смысла».
- Существует сложность в корреляции слов естественного языка и интеллектуальных понятий, поскольку зачастую определение смысла слова зависит от контекста, в том числе и социального.

Принципы классификации в онтологиях общего назначения - дорогу осилит идущий

Тем не менее, на данный момент уже создано около полутора десятков онтологий верхнего уровня, которые обширно используются на практике [20]. Одна из них – онтология BFO [1] – создавалась как верхний уровень для доменных онтологий в сфере науки. В ее основе лежит концепция разделения онтологии на статическую и динамическую части. При этом онтология претендует на спецификацию знаний о самой реальности, а не представлений о реальности. Такой подход позволяет описывать элементы онтологии на разных уровнях детализации и в разных контекстах, т.е. один и тот же объект может рассматриваться как целостный или как совокупность других объектов, и в то же время, он может быть статическим и динамическим в зависимости от выбранного «угла зрения». Универсальность представления знаний подтверждается и тем фактом, что материальные объекты онтологии не обязательно должны иметь четкие физические границы, например,

возможно существование не только понятия «человек», как целостного объекта социальных отношений и человека как сущности, включающей в себя все органы, но и «нижней левой части человеческого тела».

Онтология SUMO [11] принадлежит IEEE и преследует цель интеграции существующих онтологий в единую структуру, которая имела бы статус универсального стандарта [9]. Для связи SUMO с отраслевыми онтологиями разработана онтология среднего уровня MILO (Mid-Level Ontology). Как и многие другие онтологии верхнего уровня [15] в онтологии реализована категоризация понятий на основе философского понятийного аппарата: вершиной является понятие «Сущность», сущности разделяются на физические и абстрактные, и так далее. Но как говорилось выше, классификация понятий SUMO может быть легко оспорена: совершено не ясно, почему понятие «еда» находится на столь высоко уровне иерархии, но не введен классификационный признак «съедобное/несъедобное», как это было сделано в случае «мокрое/сухое». Также вызывает сомнение наличие в иерархии физических объектов двух понятий «Группа», одного в ветви «Коллекция», а другого в ветви «Агент». Интересен подход создателей к ограничению количества понятий онтологии верхнего уровня: установлен порог в 1000 понятий, если возникает необходимость введения нового метапонятия, то понятие нижнего уровня переносится в соответствующую онтологию предметной области.

В онтологию YAMATO [7] помимо разделения на «физическое» и «абстрактное», введено понятие «полу-абстрактное». Классификация понятий верхнего уровня основана на двух классификационных признаках: «необходимости места существования» и «необходимости времени существования». Так, абстрактным сущностям для существования нет необходимости иметь ни места, ни времени, физические сущности обязаны иметь и то, и другое, а полу-абстрактные только время. Создатели онтологии утверждают, что не существует таких сущностей, которым необходимо было бы только место [8].

Самый верхний концепт «Сущность» в онтологии DOLCE [13] подразделяется на «Статическое», «Динамическое», «Качество» и «Абстрактное». Интересно, что в онтологии DOLCE не введен наряду с понятием «абстрактное» метаконцепт «физическое», хотя классификационный признак «физическое/абстрактное» в онтологии присутствует явно, поскольку введена пара понятий «Физическое качество» - «Абстрактное статическое». В то же время «Статическое» подразделяется на «Физическое статическое» и «Нефизическое статическое». Можно сделать вывод, что в онтологии DOLCE «Абстрактное» не является явной противоположностью «Физического», что не характерно для вышеописанных онтологий верхнего уровня. Существуют и другие спорные моменты [21]

Рассмотренные выше онтологии создавались на основе композиционной гипотезы, т.е. вводился минимальный набор базовых понятий, и предполагалось, что все другие понятия можно рассматривать как композицию понятий, уже присутствующих в онтологии. Очевидно, что при таком подходе онтология удовлетворяет критерию минимальности.

Существует также другой подход к построению онтологий общего назначения, ориентированный на полноту онтологии, т.е. занесение максимально возможного количества понятий. Такая точка зрения свойственна, например, создателям WordNet [6]. При создании WordNet были использованы следующие принципы: любая компонента естественного языка могла рассматриваться отдельно от всех остальных и быть обособленным предметом исследования, и утверждалось, что существует

такое формальное описание слов, которое может быть применено к большинству слов языка. Базовой словарной единицей в WordNet является не отдельное слово, а так называемый синонимический ряд («синсеты»), объединяющий слова со схожим значением и по сути своей являющимися узлами семантической сети. Существует русскоязычная версия WordNet, где переведено около половины синсетов, но их описания остались на английском [22].

Другим примером может служить ConceptNet [2], представляющий собой в отличие от WordNet, многоязычную онтологию, охватывающую английский, китайский, португальский, японский, датский, венгерский, французский, испанский и другие языки. Примечательно, что ConceptNet содержит концепты и на русском языке. Аналогичным ресурсом является FREEBASE [4], коллаборативная база знаний, основанная на т.н. фолксономии[1]. FREEBASE содержит около 23 миллионов сущностей на английском языке, и по классификации понятий ориентирована на Интернет-сообщество, ее пополняющее.

Разница в подходах к построению обусловлена целью создания онтологии. Онтологии BFO, SUMO и др. создавались в первую очередь для систематизации данных, в то время как для WordNet и ConceptNet основополагающей идеей было создание словарей очень большой величины для использования в приложениях автоматической обработки текстов.

Классификация понятий в слабо формализуемых предметных областях

Одной из основных задач при построении онтологии общего назначения является первичная классификация объектов реального мира, которая служит базисом для наполнения онтологии. Поэтому для построения таких онтологий необходимо в первую очередь разработать систему классификационных признаков. Обобщая опыт создания подобных онтологий, можно выделить наиболее универсальные классификационные признаки:

1. По признаку существования в материальном мире – на «Материальное» и «Нематериальное»;
2. По степени абстракции – на «Абстрактное» и «Физическое»;[2]
3. По композиционному признаку – на «Единичное» и «Множество»;
4. По возможности декомпозиции – на «Целостное» и «Составное»;
5. По признаку зависимости от других элементов онтологии – на «Обособленное» и «Зависимое»;
6. По возможности совершения собственной деятельности – на «Объект» и «Агент» («Субъект»);
7. По признаку нахождения в покое и равновесии – на «Статическое» и «Динамическое»;
8. По продолжительности – на «Мгновенье» и «Временной интервал».

К наиболее универсальным понятиям можно отнести «сущность», «роль», «отношение», «процесс», «событие», «качество», «количество», «причина», «следствие», «изменение» и др. При этом понятие «Сущность» должно обозначать все, что представлено в создаваемой онтологии и является вершиной таксономии.

---

[1] Неологизм, обозначающий практику совместной категоризации посредством произвольно выбираемых ключевых слов.

[2] Как отмечалось ранее, признак не может считаться универсальным.

Фактически, приведенный выше перечень классификационных признаков определяет свойства самого верхнего концепта «Сущность». Подобный метод построения онтологии на основе многоаспектной классификации понятий подробно описывается в [21]. Согласно данному методу для каждого базового обособленного концепта онтологии формируется перечень свойств с перечислением их возможных значений. Свойства в дальнейшем выступают в качестве классификационных признаков, и каждое допустимое значение свойства задает подчиненное понятие. Таким образом, онтология естественным образом пополняется за счет введения новых свойств объектов.

С другой стороны, данный подход имеет существенный недостаток: в иерархию могут быть включены любые наименования признаков, в том числе и числовые, поэтому узлы таксономии не могут рассматриваться как базовые понятия реального мира. Например, для сущности «млекопитающее» может быть введено свойство «количество конечностей, используемых при перемещении». Тогда для присоединения к таксономии понятия «человек» необходимо определить подчиненный класс «млекопитающее, у которого количество конечностей, используемых при перемещении, равно двум». Для онтологии общего назначения введение подобного класса не приемлемо, поскольку он не является универсальным.

Система классификации на основе синтаксического разбора текстов

Первым этапом метода многоаспектной классификации является определение вершины таксономии, для которой выделяются первичные классификационные признаки. Но возможен и более абстрактный подход, при котором изначально формируется система классификации, и уже на ее основе выделяются понятия.

Для построения системы классификации предлагается основываться на результатах синтаксического разбора текстов на естественном языке. Поскольку стоит задача построения онтологии общего назначения, в качестве исходных данных целесообразно использовать как можно больше текстовых материалов, относящихся к как можно большему количеству предметных областей. В связи с этим для опытной проверки данного подхода были использованы результаты синтаксического анализа корпуса текстов на основе библиотеки Мошкова (более 600 млн. слов) и корпуса новостных текстов различной направленности (Лента.ру, РИА Новости, РБК и другие) общим объемом более 400 млн. слов [18].

Результаты кластеризации представляли собой группы терминов, выделенных автоматически на основе анализа исходных данных: { АВТОМОБИЛИЗАЦИЯ; ХИМИЗАЦИЯ }, { АГРЕССИЯ; ВТОРЖЕНИЕ; НАПАДЕНИЕ; НАСИЛИЕ; НАСТУПЛЕНИЕ; НАШЕСТВИЕ; СТОЛКНОВЕНИЕ; СТЫЧКА; СХВАТКА; УБИЙСТВО; ШТУРМ } и т.п. При построении системы классификации для каждого кластера определялся один классификационный признак, объединяющий все его термины. Например, для кластера {АВТОМОБИЛИЗАЦИЯ; ХИМИЗАЦИЯ} основным объединяющим свойством можно считать внедрение новой технологии, {АГРЕССИЯ; ВТОРЖЕНИЕ; НАПАДЕНИЕ; НАСИЛИЕ; НАСТУПЛЕНИЕ; НАШЕСТВИЕ; СТОЛКНОВЕНИЕ; СТЫЧКА; СХВАТКА; УБИЙСТВО; ШТУРМ} представляет собой некий процесс, носящий разрушительный характер.

Для формирования полноценной системы классификации каждому объединяющему свойству необходимо привести в соответствие свойство-антогонист. Методом обобщения для кластера { АВТОМОБИЛИЗАЦИЯ; ХИМИЗАЦИЯ } мы получаем классификационный признак «Усовершенствование/Застой», для кластера {

АГРЕССИЯ; ВТОРЖЕНИЕ; НАПАДЕНИЕ; НАСИЛИЕ; НАСТУПЛЕНИЕ; НАШЕСТВИЕ; СТОЛКНОВЕНИЕ; СТЫЧКА; СХВАТКА; УБИЙСТВО; ШТУРМ } – «Разрушительное/Созидательное». Таким образом, мы получили достаточно универсальные классификационные признаки для построения онтологии общего назначения, каждый из которых может рассматриваться как базовое понятие реального мира.

Выводы

Создание системы классификации понятий является приоритетной задачей при построении онтологий для слабо формализуемых предметных областей. При наполнении таких онтологий следует отталкиваться от наиболее общих классификационных признаков, которые сами могут быть рассмотрены как базовые понятия реального мира. Для достижения последовательности представления и полноты онтологии предлагается формировать систему классификации понятий на первом этапе создания онтологии, и наполнять ее понятиями согласно выделенным классификационным признаком.

Система классификации также как и сама онтология должна обладать как можно большей универсальностью. Для того чтобы классификация была более объективной, можно использовать результаты синтаксического анализа больших корпусов текстов различной тематики.

Список литературы

1. Basic Formal Ontology, BFO [Электронный ресурс]. — Режим доступа: http://www.ifomis.org/bfo/
2. ConceptNet [Электронный ресурс]. — Режим доступа: http://conceptnet5.media.mit.edu/
3. Fox M.S., Chionglo J.C., Fadel F.G. A Common-Sense Model of the Enterprise, in 2nd IE Research Conference Proceedings, May 1993, Los Angeles, CA, 1993.
4. Freebase [Электронный ресурс]. — Режим доступа: http://www.freebase.com/
5. Hovy E. A Standard for Large Ontologies [Электронный ресурс]. — Режим доступа: http://www.isi.edu/nsf/papers/hovy2.htm.
6. Miller G. A., Beckwith R., Fellbaum C., Gross D., Miller K.J. Introduction to WordNet: an on-line lexical database. // International Journal of Lexicography 3 (4), 1990, pp. 235 - 244.
7. Mizoguchi R. YAMATO: Yet Another Advanced Top-level Ontology. Draft as of March 17 2010 [Электронный ресурс]. — Режим доступа: http://www.ei.sanken.osaka-u.ac.jp/hozo/onto_library/upperOnto.htm.
8. Mizoguchi R. YAMATO: Yet Another More Advanced Top-level Ontology / The Institute of Scientific and Industrial Research Osaka University 8-1 Mihogaoka, Ibaraki, Osaka 567-0047, Japan
9. Niles, I., Pease, A. 2001. Towards a Standard Upper Ontology. // Proceedings of the 2nd International Conference on Formal Ontology in Information Systems (FOIS-2001); под ред. Welty C., Smith B – Ogunquit (USA) - 2001
10. Nirenburg S., Raskin V.Ontological Semantics. Cambridge, MA, 2004.
11. Ontology portal [Электронный ресурс]. — Режим доступа: http://www.ontologyportal.org/.
12. Uschold M., King M., Moralee S., Zorgios Y. The Enterprise Ontology. Enterprise Project Deliverable: MID 3.1, Version 1.1, 1995.

13. WonderWeb project [Электронный ресурс]. — Режим доступа: http://www.loa.istc.cnr.it/DOLCE.html.

14. Большой энциклопедический словарь [Электронный ресурс]. — Режим доступа: http://www.vedu.ru/BigEncDic/.

15. Волкова Г.А. Обзор методологий и методов построения онтологий с чистого листа // Материалы 3-й международной научно-практической конференции «Модель подготовки специалистов новой формации, адаптированных к инновационному развитию отраслей» (Душанбе, РТ, 2-3 ноября 2012 г.).

16. Гурьянова, М. А. Онтологическое моделирование экономики предприятий и отраслей современной России: Часть 2. Мировые исследования и разработки: аналитический обзор: препринт WP7/2011/08 (ч. 2) [Текст] / М.А. Гурьянова, И.В. Ефименко, В.Ф. Хорошевский; Нац. исслед. ун-т «Высшая школа экономики». – М.: Изд. дом Высшей школы экономики, 2011. – 88 с.

17. Ефименко, И. В., Хорошевский, В. Ф. Онтологическое моделирование экономики предприятий и отраслей современной России: Часть 1. Онтологическое моделирование: подходы, модели, методы, средства, решения: препринт WP7/2011/08 (ч. 1) [Текст] / И. В. Ефименко, В. Ф. Хорошевский; Нац. исслед. ун-т «Высшая школа экономики». – М.: Изд. дом Высшей школы экономики, 2011. – 76 с.

18. Клышинский Э.С., Кочеткова Н.А. Метод автоматической генерации модели управления глаголов русского языка / Тринадцатая национальная конференция по искусственному интеллекту с международным участием КИИ-2012 (16-20 октября 2012 г., г. Белгород, Россия). Том 2 [Электронный ресурс]. — Режим доступа: http://publications.hse.ru/chapters/66264822/

19. Митрофанова О.А., Константинова Н.С. Онтологии как системы хранения знаний / Всероссийский конкурсный отбор обзорно-аналитических статей по приоритетному направлению "Информационно-телекоммуникационные системы", 2008. - 54 с.

20. Пивоварова Л.М. Онтологии верхнего уровня (обзор) // Компьютерная лингвистика и развитие семантического поиска в Интернете: Труды научного семинара XIII Всероссийской объединенной конференции «Интернет и современное общество». Санкт-Петербург, 19 – 22 октября 2010 г. / Под ред. В.Ш. Рубашкина. — СПб., 2010. — 94 с.

21. Рубашкин В. Ш., Пивоварова Л. М. Методология наполнения онтологий — практика без теории? // Труды Второго Симпозиума «Онтологическое моделирование», г. Казань, 11–12 октября 2010 г. Ред. Калиниченко Л. А. — М: ИПИ РАН, 2011.

22. Русский Wordnet [Электронный ресурс]. — Режим доступа: http://www.wordnet.ru/

Статистические языковые методы. Коллокации и коллигации

Кочеткова Н.А.
МИЭМ НИУ ВШЭ, ФИТиВТ

В последнее время наблюдается накопление массивов специализированных текстовых документов. В ходе своего существования предприятия формируют архивы документации колоссального объема. Эти данные требуют не только хранения, но и соответствующей обработки. А так как значительная часть документов представляет собой текстовое описание то, для выполнения этих задач требуется использование методов автоматической обработки текстов на естественном языке.

Анализ текстов на естественном языке

Этапы анализа текста на естественном языке практически не зависят от выбранного языка.

Обычно различают следующие этапы анализа текста :

1. графематический анализ — выделение структурных единиц из входного текста;
2. морфологический анализ — определение морфологических характеристик каждого слова — часть речи, падеж, склонение, спряжение и т.д.;
3. синтаксический анализ — построение синтаксического представления предложения;
4. семантический анализ — построение аргументно-предикатной структуры высказываний или другого вид семантического представления предложения.

На каждом из этих этапов возникают различного рода неоднозначности, и из-за невозможности кодирования всех явных и неявных знаний о языке, для решения многих задач обработки текста используются статистические языковые методы.

Статистические языковые методы

Статистические языковые методы рассматривают естественный язык как случайный процесс, что позволяет переопределить многие задачи, связанные с обработкой естественного языка, в более строгом математическом смысле.

Для формального описания статистической модели языка используется модель скрытых Марковских цепей. Самая распространенная версия этой модели - модель n-грамм, соответствующая скрытой Марковской цепи n-1 порядка, где вероятность P(w) появления какого-то символа последовательности $w=w_1,w_2,..w_t$ длины T - это первое разложение в соответствии с теоремой Байеса.

Определенная таким образом модель n-грамм позволяет предсказать появление символов в некотором ограниченном наборе, основываясь на контексте из n-1 предшествующего элемента. С увеличением длины контекста возникают серьезные трудности при вычислениях такой модели, и применение ее на практике требует огромных затрат памяти.

Коллокации

Коллокации - это частный случай модели n-грамм. Коллокации – это комбинация двух или более слов, имеющих тенденцию к совместной встречаемости.

Так как основное внимание уделяется частоте совместной встречаемости, то коллокации в корпусной лингвистике могут быть определены как статистически устойчивые словосочетания. За последние годы появилось большое число

исследований и разработок, посвященных коллокациям, затрагивающих как теоретические аспекты статистического подхода к данному понятию, так и практические методы выявления и применения коллокаций.

Извлечение коллокаций — довольно простой процесс – необходимо лишь разбить весь текст на последовательности заданной длины и подсчитать частоту каждой уникальной последовательности. Коллокации нашли широкое применение в системах машинного перевода, а так же часто используются для снятия неоднозначностей и поиска ошибок. И хотя точность полученных таким образом инструментов не велика, они существенно сокращают объем ручной обработки.

Также коллокации достаточно успешно применяются в таких задачах, как выделение терминологии и семантических предпочтений. Однако стоит отметить, что в большинстве работ, помимо самого словаря коллокаций вводятся разного рода дополнения: от фильтрации по словарям до фильтрации по частеречным и синтаксическим шаблонам. Естественно любые привнесенные лингвистические правила существенно улучшают качество получаемых инструментов.

Также существует множество «мер устойчивости», которые позволяют выделять те или иные «интересные» коллокации. К наиболее распространенным относятся MI и T-score. MI — позволяет выделять наиболее редкие и своеобразные коллокации и подходит для выделения терминологии, имен собственных и прочих конструкций, в которых частота составляющих коллокацию слов в тексте вне этой коллокации ничтожно мала. T-score — напротив позволяет найти наиболее распространенные частые обороты. Эти (и другие) меры сводятся к простым преобразованиям частотных характеристик коллокации и каждого из коллокатов, и дают существенно меньшую точность выделения, чем введение фильтров, основанных на лингвистических правилах.

Кроме того, стоит отметить, что хотя речь чаще всего идет о коллокациях вообще, на практике обычно используют биграммы, гораздо реже триграммы, а работ по четырехграммам и более применительно к задачам автоматизации нет. Так для четырехграмм до сих пор нет ни одной общепринятой меры устойчивости.

Словари коллокаций существенно зависят от корпуса, из которого они извлечены, и потому трудно сравнимы между собой. Еще одна трудность – пороговые значения, которые зависят от размера корпуса. Чем выше значение порога, тем больше редких коллокаций не входят в словарь, и главным становится вопрос: «как не потерять верные, но редкие события?» Коллокации бывают как контактными — когда слова идут в тексте одно за другим, а бывают и неконтактными — когда из последовательности выбираются скажем 1 и 3 слово, то есть некоторое количество слов последовательности может пропускаться. Однако извлечение неконтактных коллокаций требует больше затрат, в связи с чем неконтактные коллокации используются гораздо реже.

Коллигации

Еще одной статистической моделью являются коллигации. Если коллокации дают представление о поведении слов и словоформ, то коллигации — о поведении грамматических групп. В отличие от чисто лексического характера коллокаций, коллигации имеют смешанный лексико-грамматический характер. Коллигация - это комбинация лексических и грамматических характеристик, имеющих тенденцию к совместной встречаемости. Зачастую в качестве таких характеристик выбирают части речи или более фразовые элементы. Обязательным условием коллигации является то,

что в качестве одного из коллигатов должна выступать словоформа или лемма. Коллигации являются очень удобным промежуточным звеном между фразовыми шаблонами и коллокациями, представляя собой смесь лексических единиц, меток частей речи и фразовых категорий.

Очевидно, что извлечение коллигаций гораздо сложнее извлечения коллокаций. Сначала необходимо разметить части речи, на основе этого выделить фразовые элементы, выбрать из всех вариантов разбора корректные, и установить предпочтительность лексемы или параметра. Никакого систематического описания и массового применения коллигации еще не получили. Наиболее распространены словари коллигаций, в которых первый коллигат представлен словоформой, а второй коллигат - всеми возможными грамматическими параметрами слова, следующего за первым, со статистикой.

Основными недостатками статической модели коллокаций/коллигаций можно считать:

а) заведомо неверное предположение о независимости вероятности появления очередного слова/параметра от более длинной истории

и

б) колоссальные объёмы требующихся обучающих данных.

Однако, как словари коллокаций, так и словари коллигаций, а также словари частот могут быть крайне полезны в сочетании с другими методами, так как они дают общее представление о поведении слов в тексте.

Рассмотрим подробнее применение словарей коллокаций на разных этапах анализа текстов.

Статистический синтаксический анализ

В работе [5] описан синтаксический анализ на основе словаря n-грамм с использованием грамматики основанной на зависимостях. Вместо обобщения синтаксических правил для сегментации последовательности слов на две клаузы используется информация о частях речи, получаемая из n-грамм. Вводится специальная часть речи - сегментационное слово, которое соответствует начальному или конечному символу клаузы. Это позволяет выделять структуру клаузы, и впоследствии сделать процесс разделения автоматическим. Сегментационные слова для каждой клаузы извлекались из небольшого размеченного вручную корпуса. Экспериментальные результаты для предложений на японском языке показали, что синтаксический анализатор использующий описанные выше дополнения достигает 72,2% полноты, что является примерно тем же уровнем производительности, что и у синтаксического анализатора на основе вероятностных контекстно-свободных грамматик с введенными человеком лингвистическими правилами.

Статистические методы в семантическом анализе

Неоднозначность смысла слова распространена во всех естественных языках. Правильный смысл многозначного слова может быть выбран на основе контекста, в котором оно встречается, и, соответственно, проблема разрешения семантической неоднозначности слова может быть определена как задача автоматического назначения многозначному слову наиболее подходящего смысла исходя из контекста. Одним из наиболее частых статистических способов выделения семантической связности слов являются коллокации.

Так в работе [2] используются две идеи:

- слова имеют характерные семантические профили - просодии
- сила связи между словами может быть измерена количественно.

Объединив эти две идеи можно получить сравнительные семантические профили слов, которые показывают частоту и характерные коллокаты заданного слова, и, следовательно, выявляют семантические отношения между коллокатами.

На данных корпуса размером в 120 млн. слов было показано, что лемма ПРИЧИНА (CAUSE) встречается преимущественно в «неприятных» коллокациях, таких как *причина проблемы (cause trouble)* и *причина смерти (cause death).* Подробное изучение этой леммы используется для пояснения количественных методов исследования коллокаций. Было проведено краткое сравнение семантических особенностей для связанных лемм, например, ПРИЧИНА (REASON) и СЛЕДСТВИЕ (CONSEQUENCE). И результаты показали такие отношения между леммами и семантическими категориями, которые в настоящее время не захвачены ни одним из словарей или грамматик. Использование извлеченных из корпуса коллокаций гарантирует надежность получаемых семантических профилей, из-за повторяемости их на миллионах слов.

Аналогичный метод использования коллокаций для семантических целей представлен в статье [1].

Принцип семантического предпочтения зависит от отношений между набором часто встречающихся коллокатов и некоторыми общими семантическими особенностями.

Эта статья представляет описание семантических профилей, созданных на основе этого принципа с использованием метода, основанного на корпусной лингвистике, включающего в себя четыре этапа: вычисление, сжатие данных, семантический анализ и интерпретация.

Полученные в результате профили и формат их представления делают их легко поддающимися использованию переводчиками. Во-первых, они практически обоснованы и являются представителями того корпуса, из которого они получены. Во-вторых, характер значения, который можно найти в таких профилях, не ограничен лексическим или словарным значением, но включает в себя прагматическую или энциклопедическую информацию, а также ту, которая не могла бы быть предсказана из словарного определения.

Кроме того, можно показать, что частота, с которой некоторые значения многозначных слов, найденных в корпусе, предлагает способ, позволяющий сделать выводы о типе текста или жанре. Проверка числа этих профилей подтверждает предположение Стаббса (2001) о том, что такие профили могут быть использованы в качестве доказательств семантических полей.

Семантические профили, таким образом, могут предложить аккуратное и доступное средство для переводчиков, или, если они применяются в параллельных корпусах то, как часть базы данных для автоматического перевода.

Заключение

До сих пор не существует методов, позволяющих полностью снимать разного рода неоднозначности анализа. Так на уровне слов существует омонимия - ситуация, когда одной словоформе можно приписать несколько нормальных форм. При синтаксическом разборе оказывается, что связи между словами можно назначить более чем одним способом. Может оказаться слишком много вариантов соединения слов, перебор которых займет катастрофически много времени и сложность задачи

будет расти экспоненциально с ростом количества слов. Проблема с семантической неоднозначностью стоит еще острее. Большинство методов опираются лишь на словарные статьи. И хотя за последние десять лет, известность статистических приложений по обработке естественного языка, которые используют статистические, а не только основанные на правилах анализаторы, увеличилась очень значительно поиски более лучших решений продолжаются. Корпусный подход и мощности современной вычислительной техники дают уверенность, что лексические описания в будущем будут предоставлять более точную и исчерпывающую информацию о словах, и дадут доступ к неявным, скрытым моделям языка.

Список литературы

1. Allan Lauder Collocation, Semantic Preference and Translation: Semantic Preference as a Reference Source for Translation, in Proceedings of the The Regional Conference of the International Association of Forensic Linguists 2010, 2010.
2. Michael Stubbs Collocations and semantic profiles: on the cause of the trouble with quantitative studies, in Functions of Language, 2, 1, 1995.
3. Serge Sharoff, Joakim Nivre, The proper place of men and machines in language technology: Processing Russian without any linguistic knowledge. in Proceedings of the Dialog 2011, 2011.
4. Yue Zhang and Joakim Nivre. Transition-based Dependency Parsing with Rich Non-local Features. Proceedings of ACL 2011, 2011.
5. Nobuo Inui; Yoshiyuki Kotani. Robust N-gram Based Syntactic Analysis Using Segmentation Words. Proceedings of the 15th Pacific Asia Conference on Language, Information and Computation, 2001.
6. Большакова Е.И., Клышинский Э.С., Ландэ Д.В., Носков А.А., Пескова О.В., Ягунова Е.В. Автоматическая обработка текстов на естественном языке и компьютерная лингвистика: учебное пособие, М.: МИЭМ, 2011.

Анализ методики построения новостного анонса по теме «электронное правительство и электронные услуги» и время жизни информационного повода

А.В. Чижик
СПбГУ, физический факультет, кафедра квантовых магнитных явлений,
факультет искусств, кафедра информационных систем в искусстве и гуманитарных науках
chizhik.spinus@gmail.com

Аннотация. В статье описаны результаты анализа коллекции новостных анонсов за годовой период по теме «электронное правительство и электронные услуги». Выявлены основные особенности жанра. Приведено описание методов сбора коллекции.

Введение

Глобальная сеть Интернет, появившаяся для облегчения поиска и консолидации информации в научных кругах, в последние десятилетия благодаря развитию технологий веб-программирования стала использоваться как один из основных коммуникационных каналов между людьми [1]. Общей тенденцией текстового материала в Интернет-изданиях является стремление к краткости и точности изложения. Можно выделить три основных объекта любого Интернет-СМИ: заголовок, анонс и раскрываемая после клика на него статья. Наибольший интерес для исследования представляет специфический жанр анонса, который существует исключительно в дискурсе Интернет-СМИ. Его особенность заключается в основной функции – мотивировании читателя перейти по ссылке и продолжить чтение статьи. С точки зрения самостоятельности анонса как необходимой для Интернета формы компактной передачи информации можно выделить следующие свойства:

1. автономность: новостной анонс в дискурсе Интернет-СМИ строится таким образом, чтобы читатель мог получить необходимую информацию о событии еще не переходя к тексту основной статьи;
2. разноаспектность: в зависимости от типа Интернет-СМИ и от его статуса одна и та же новость будет освещаться в рамках анонса по-разному;
3. ограниченное время жизни.

Ниже представлены результаты работы по созданию и анализу коллекции новостных анонсов по теме «электронное правительство и электронные услуги» за период с 1 января 2012 г. по 28 февраля 2013 г. (68661 слово). Сбор материала проходил в рамках сотрудничества с Центром технологий электронного правительства НИУ ИТМО. Основные порталы, чьи новостные анонсы вошли в коллекцию: Российская газета, Минкомсвязь России, CNews, ТАСС-Телеком, Фонд информационной демократии, ИТАР-ТАСС, Экспертный центр электронного государства, МИНСВЯЗЬ России, Коммерсантъ, ComNews, PC Week, Ведомости.

Сбор коллекции

Сбор коллекции происходил в три этапа. На первом этапе были использованы возможности сервисов по обработке новостей (Google News и Яндекс.Новости), т.к. в качестве основной задачи здесь можно выделить кластеризацию новостей по тематике и выделение заголовка новостного кластера, что дало неплохой результат для первичного отсева новостных потоков.

Затем встал вопрос об оценки качества получаемых новостных кластеров: некоторые события (особенно крупные, например, кластер новостей, связанных с выборами президента) имели достаточно сложную внутреннюю структуру, и целую систему рерайтов, которые также включаются сервисами в кластер [2].

Для более точной обработки новостных лент на интересующую предметную область была использована программа, извлекающая информацию из web-страницы, используя иерархическую структуру тэгов. Алгоритм программы основан на рассмотрении документа в виде дерева разбора согласно стандартной объектной модели DOM [3], что дает возможность характеризовать структурные свойства документа и осуществить поиск нужной информации (использовался ручной способ описания правил распознавания нужной информации). Схематично процесс сбора информации можно представить в следующем виде (рис. 1).

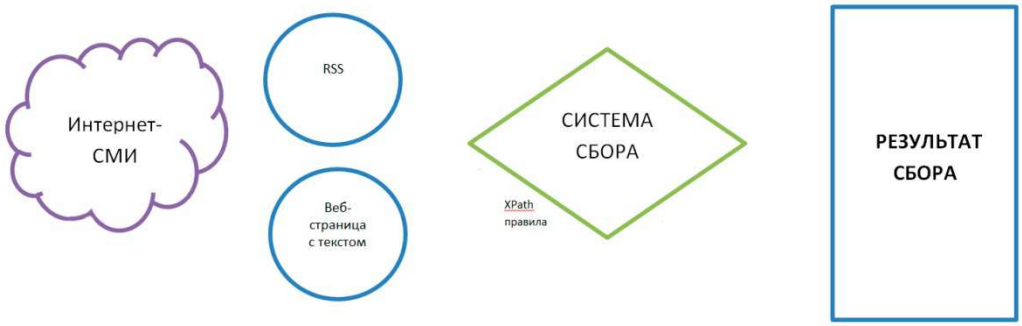


Рис. 1. Схема метода извлечения

Следует отметить, что это стандартный алгоритм для решения подобных задач, в данном случае процесс описания правил для извлечения информации был облегчен за счет того, что для исследования были выбраны только заголовки и анонсы, а как дополнительная информация в предварительный файл записывались дата публикации и название СМИ. Сбор информации происходил два раза в неделю. Итак, были решены основные задачи: получение информации из HTML-страницы, структурирование информации и сохранение информации в формате .txt. С помощью программы, уточнившей условия выборки новостей, коллекция приобрела наиболее специализированный вид (отсеялись темы из смежных областей).

На третьем этапе полученный автоматическим путем файл еженедельно проверялся аналитиком и пополнялся новостями, которые по каким-либо причинам не прошли систему фильтров, однако имели отношения к теме электронного правительства.

Поиск коллокаций

Полученную коллекцию было решено исследовать на предмет устойчивых конструкций, характерных для жанра анонса по заданной тематике. Выбор коллокаций, наиболее часто встречающихся в коллекции, дал большой шум, т.к. содержал биграммы, являющиеся парными функциональными словами. Однако, при выставлении фильтра определяющего лингвистические паттерны, которые могут быть смысловой единицей предложения (определение части речи слова), результаты приобрели годный для последующей обработки вид. Обозначив абсолютную частоту встречаемости коллокации как L(w1, w2), в таблице 1 показан ранжированный список высокочастотных биграмм после применения фильтра.

Таблица 1. Двадцать одна наиболее встречаемая коллокация в новостных анонсах по теме «электронное правительство»

L(w1, w2)	w1	w2	Tag Pattern
11487	российской	федерации	AN
7261	информационных	технологий	AN
5412	информационной	демократии	AN
3301	Николай	Никифоров	NN
3191	мероприятий	по	NP
2691	государственных	органов	AN
2567	государственной	власти	AN
2308	в	сети	PN
2171	электронных	карт	AN
2092	электронных	заявлений	AN
2021	экспертно-общественное	обсуждение	AN
1973	фонд	информационной	NA
1903	стандарта	открытости	NN
1839	универсальный	электронный	AN
1801	субъектах	российской	NA
1766	стратегии	кибербезопасности	NN
1703	ФНС	России	NN
1210	социальные	сети	AN
1004	Совете	Федерации	NN
988	региональной	информатизации	AN
970	открытого	правительства	AN

По выборке первых двадцати одной часто встречаемой коллокации очевидна закономерность, что для биграмм по заданной теме, во-первых, характерны конструкции вида AN, где в качестве прилагательного типично появление слов «информационный», «электронный» и «государственный». Во-вторых, в качестве конструкций NN встречаются словосочетания, входящие в семиотический круг данного сообщества (н-р, «Николай Никифоров» – министр связи и массовых коммуникаций Российской Федерации, а «стандарт открытости» – одно из главных направлений обсуждений и разработок экспертно-общественного сообщества и органов исполнительной власти в рамках программ, связанных с информационным обществом и электронным правительством, особенно в течение года предшествующего вступлению в ВТО).

Запуск программы на подсчет частотности отдельных слов (с учетом вариативности словоформ) дает следующие тринадцать первых результатов в порядке убывания: электронный, правительство, государственный, услуга, система, Россия, госуслуга, внедрение, портал, Российская, РФ, Ростелеком, органы, Федерация. Если отсеять шум в виде слов «Россия», «РФ», «Российская Федерация» и пр. (характерных для новостных текстов любой общественно значимой для нашей страны темы), получается облако слов, используемых для написания любого анонса по теме электронного правительства [4].

Этот анализ косвенно показал, что задавать программе вычисления функции дисперсии для выявления N-грамм повлечет избыточность кода, т.к. для нахождения таковых, руководствуясь правилами русского языка, можно задать окно с допуском на одно слово и этого будет достаточно. При этом прогон программы выдает

тридцать одну неявную коллокацию вида «внедрение электронного правительства»: к биграммам из таб.1 прибавляется одно слово из списка тринадцати часто встречаемых отдельных слов, однако частота встречаемости такого рода конструкций намного меньше (общая частота для всех подобных коллокаций – 539).

Время жизни информационного повода

При сборе коллекции новостных анонсов изначально была поставлена задача отсева одинаковых новостей, что осуществлялось путем сравнения заголовков на предмет полного совпадения и по информации, а также статьи анализировались на наличие ссылки на первоисточник. Однако эта задача выявила две подзадачи исследования: 1) определить время жизни информационного повода по заданной тематике (количество дней, когда новостной поток пополняется новыми фактами о событие); 2) отследить количество рерайтов новости и время затухания перепечатки новости.

Был разработан следующий алгоритм:

1. получение новой новости от информационного агентства (эта новость – глава класса новостей);
2. получение новых новостей от других источников: классификация новостей, как относящихся к этой новости или не относящихся;
3. нахождение такого дня, в котором, ничего связанного с новостью не было найдено (проверка еще несколько дней после такого дня);
4. подсчет количества дней, определение длины.

Предварительное отслеживание новостного потока за месяц показало, что существуют два типа новостей: новости длинного цикла и новости короткого цикла. Новость длинного цикла может на протяжении периода от нескольких недель до нескольких месяцев оказывать сильное влияние деятельность людей, соприкасающихся с тематикой электронного правительства. Примером такой новости может служить информационный поток об обеспечении Ростелекомом online-трансляции выборов президента 2012 г. Обозначенный контекст новости является по времени серединой развития события: в Интернет-СМИ эта тема продолжала быть актуальной (в виде самобытных материалов) около трех месяцев с момента первого сообщения (включив новости об объявлении тендера и заканчивая отзывами о трансляции). Рерайты новостей длинного цикла затухают примерно в тот же день, что и заканчиваются информационные поводы для продолжения освещаемой новости, при этом время возникновения рерайтов также практически совпадает с появлением первой новости (это обусловлено тем, что обычно первая информация по новости длинного цикла является официальным релизом или заявлением органов государственной власти, что обеспечивает достаточно быстрое ее распространение). Новость короткого цикла обычно имеет время жизни около трех дней, рерайты в этом случае идут с опозданием от часа до двух дней.

Заключение

В данной статье приведены предварительные результаты исследования, показавшие характерное представление информации в новостном анонсе по теме электронного правительства и электронных услуг. По статистическим методам исследования лингвистических особенностей текста можно сделать вывод о том, что новостной анонс по данной тематике нацелен на привлечение внимания людей к проблематике развития информационного общества, т.к. для этого жанра характерно

использование устойчивых конструкций, вычленение которых сразу дает представление об информации в статье. Исследование временного цикла жизни информационного повода показало, что данную информацию можно применять для аналитического обзора социальной значимости новости.

Список литературы

1. Кастельс М. Информационная эпоха: экономика, общество и культура / Пер. с англ. под науч. ред. О. И. Шкаратана. – М., 2000. – С. 65-70.

2. Андреев С.Н. Многомерный подход: кластерный и дискретный анализ лингвистических данных // Многомерный анализ в лингвистике. – Смоленск, 2002. – С. 3-19.

3. Жамбю М. Иерархический кластер-анализ и соответствия. - М.: Финансы и статистика, 1988.

4. Трошина Н.Н. Стилистические параметры текстов массовой коммуникации и реализация коммуникативной стратегии субъекта речевого воздействия // Речевое воздействие в сфере массовой коммуникации. – М., 1990.

Кластеризация семантически связанных слов из неразмеченного текста на русском языке

Э.С. Клышинский, Н.А. Кочеткова
МИЭМ НИУ ВШЭ, eklyshinsky@hse.ru

Аннотация. В работе рассматривается метод кластеризации слов, связанных между собой некоторым семантическим признаком или набором признаков. Кластеризация основана на анализе синтаксически связанных сочетаний слов предварительно выделенных из неразмеченного текста. Точность кластеризации составила около 80%. Данный проект выполнен при частичной финансовой поддержке гранта РГНФ № 12-04-00060.
Ключевые слова: поверхностный синтаксический анализ, кластеризация, семантически связанные группы слов.

Введение

В ходе работ по созданию онтологий разработчикам приходится решать задачу объединения семантически связанных слов в группы. Степень сходства связываемых слов зависит от конкретной задачи и обычно определяется вручную. Средства автоматизации данного процесса начали создаваться достаточно давно. Так, например, используется извлечение информации из существующих словарей [1] или извлечений терминов из текста [2, 3]. В данной работе мы применили для поиска семантически связанных слов информацию о синтаксической сочетаемости слов.

Синтаксическая сочетаемость слова в тексте определяется валентностями данного слова, то есть его способностью вступать в синтаксические связи с другими словами [4, 5]. Согласно Кацнельсону, валентностями обладают как глаголы, так и существительные. В соответствии с принятым определением мы можем считать, что валентностью также обладают, например, прилагательные, к которым могут присоединяться определенные наречия.

База данных или словарь валентностей слов является серьезной помощью при синтаксическом анализе или построении онтологий. На данный момент делались попытки создать подобные словари, однако их объем всё еще недостаточен для проведения качественного синтаксического анализа. Так в работе [6] приводится описание валентностей для более чем 8000 глаголов.

Валентности можно разделить на актанты и сирконстанты [7, 8]. Здесь мы будем считать, что актант является обязательным участником действия, хотя и может быть выделен из контекста, а не записан в предложении явно. Сирконстанты, в свою очередь, являются необязательными участниками действия. Таким образом, актанты должны встречаться с данным действием чаще. Кроме того, они чаще должны занимать позицию рядом с глаголом.

Для глаголов в русском языке валентности могут выражаться (а) падежом, (б) предлогом и падежом и, наконец, (в) предлогом, падежом и семантическими характеристиками главного и зависимого слов. Для существительных валентности указываются (г) согласованием грамматических и семантических характеристик главного и зависимого слов. Главные слова можно разделить по количеству подчиняющихся им актантов. Кроме того, главные слова различаются между собой набором подчиняемых актантов. В соответствии с (в) и (г) мы можем ожидать, что часть подчиненных слов у данного главного слова будет образовывать семантическую группу. Мы пошли дальше и сформулировали противоположное предположение, что набор зависимых слов может определить семантическую группу

главного слова. При этом можно использовать сами зависимые слова, а не наши представления об их семантическом сходстве.

Логично предположить, что для глаголов следует принимать в рассмотрение только актанты, так как сирконстанты могут присоединяться к гораздо более широкому классу слов. Само разделение зависимых слов на актанты и сирконстанты является отдельной серьезной задачей. Однако здесь можно считать, что актанты должны встречаться чаще, чем сирконстанты. В случае рассмотрения пар существительное + прилагательное зависимое слово, судя по всему, будет являться сирконстантом, так как обязательность прилагательного означает термин или коллокацию.

Таким образом, дальнейшая работа основывается на двух предположениях, что семантически сходные слова обладают сходным употреблением зависимых слов, а также что в случае рассмотрения глаголов наиболее часто встречающиеся употребления зависимых слов скорее всего относятся к актантам, а не сирконстантам.

Предлагаемый метод

Предлагаемый метод выделения групп семантически связанных слов с использованием информации о валентностях заключается в следующем. На вход метода подается база синтаксически связанных групп слов, которая была получена нами в ходе предыдущих работ [9, 10]. База данных хранит информацию о группах вида существительное + прилагательное, глагол + существительное или глагол + предлог + существительное, а также абсолютную частоту встречаемости данных групп в проанализированном корпусе текстов. Все слова приведены к своим начальным формам.

Заметим, что эти данные не являются информацией о валентностях слов, а только реализацией этих валентностей в тексте. Нашим третьим предположением является предположение о том, что с некоторым словом с заданными семантическими характеристиками чаще всего в роли определенных валентностей будут встречаться одни и те же слова. Если данное предположение окажется верно, нам не нужно будет получать информацию о валентностях слов в явном виде, а достаточно будет использовать описанные выше данные.

Для отсечения шумов отбрасываются все сочетания с частотой встречаемости ниже заданной. Кроме того, выбираются только те главные слова (и соответствующие им сочетания), у которых мощность множества зависимых слов превышает некоторое пороговое значение. Это необходимо, чтобы отсеять шум в извлекаемых из коллекции сочетаниях. Пороговое значение частоты встречаемости позволяет избавиться от случайно попавших в базу сочетаний, число различных сочетаний гарантирует нам достаточную статистику для сравнений.

Для каждого главного слова может быть составлен вектор зависимых слов с приписанными к ним частотами встречаемости пары «главное слово – зависимое слово» в корпусе. В качестве меры сходства векторов была выбрана косинусная мера.

$$\mathrm{Cos}(\mathbf{x}, \mathbf{y}) = \frac{\sum_{i=1}^{n} \mathbf{x}_i * \mathbf{y}_i}{\sqrt{\sum_{i=1}^{n} \mathbf{x}_i * \mathbf{x}_i *} \sqrt{\sum_{i=1}^{n} \mathbf{y}_i * \mathbf{y}_i}}, \qquad (1)$$

где $\mathbf{x}_i$ и $\mathbf{y}_i$ – i-е элементы сравниваемых вектора частот встречаемости. Здесь считается, что вектора выровнены по употребляемым словам и их длины совпадают и равны максимальной длине одного из двух векторов. Частота отсутствующих элементов равна 0.

Мера сходства Дайса в данном случае работает недостаточно хорошо. Это связано с тем, что для двух главных слов может быть найдено два одинаковых подчиненных слова с существенно отличающейся частотой встречаемости. У одного главного слова это будет часто встречающееся выражение, тогда как у другого слова случайный шум. Мера Дайса не учитывает подобные нюансы и оба слова будут засчитаны с одинаковым весом.

Заметим также, что вместо абсолютной частоты встречаемости слов в корпусе следует брать ее логарифм. Это связано с тем, что для некоторых пар главных слов имеются устойчивые словосочетания с одним и тем же зависимым словом, частота встречаемости которых на несколько порядков превышает встречаемость остальных комбинаций. В качестве примера можно привести сочетания «Новый Орлеан» и «Новая Зеландия», которые чаще всего встречаются именно в таком сочетании. При этом Зеландия также может быть «далекой», «датской», «северной», «южной» и т.д., а Орлеан «французским», «чудесным», «шумным» и также «южным». В целом пересечение по лексике является незначительным. Однако в данном случае косинусная мера оказывается очень чувствительна к дисбалансу частот и практически не учитывает наличие других слов. Логарифмирование частот позволяет сгладить эту разницу и существенно снизить процент ошибок кластеризации.

На следующем шаге по вычисленной мере сходства проводится кластеризация главных слов. В кластеризации принимают участие только те пары главных слов, значение косинусной меры для которых превышает определенное значение. Тем самым мы сокращаем набор анализируемых данных и устраняем появление слабо связанных кластеров в конце работа алгоритма кластеризации.

В связи с тем, что количество кластеров на выходе метода не известно, предлагается использовать один из агломеративных методов. Кластеризация проводится до тех пор, пока мера сходства между объединяемыми кластерами больше заданного значения.

Для построения онтологии больше подходит двухпроходный метод кластеризации. На первом этапе строится плоская кластеризация кластеров, после чего полученные кластеры объединяются между собой иерархическим методом. Данный метод позволяет построить костяк онтологии, выделив не только группы семантически связанных слов, но и связать полученные кластеры между собой на более абстрактном уровне.

Результаты экспериментов и выводы

Для проверки предложенных подходов были проведены эксперименты с текстами на русском языке. Нами был взят рассматривавшийся в наших предыдущих работах [11] корпус объемом 7,2 млрд словоупотреблений. Корпус состоял из нескольких подкорпусов беллетристики (Библиотека Мошкова – 688 млн словоупотреблений, lib.rus.ec – более 6 млрд), новостных текстов за 1999 – 2010 гг. (самые значимые РИА Новости – 186 млн, Независимая газета – 97 млн, Лента.ру – 41 млн, общий объем – 511 млн), научных текстов (авторефераты, диссертации, статьи из сборников конференций и журналов, объемом несколько десятков миллионов словоупотреблений, общий объем – 51 млн словоупотреблений).

Из данного корпуса было выделено более 23 млн. уникальных связок вида «глагол+предлог+существительное» без учета формы существительного и глагола. Также было выделено более 5,5 млн сочетаний «существительное+прилагательное»

также приведенных к нормальной форме. Для каждого сочетания была подсчитана встречаемость на всём указанном корпусе.

Для русского языка использовались сочетания существительных с прилагательными и глагола с существительными. Для английского языка использовались только связи между существительными и прилагательными. Результаты экспериментов показали, что высказанная гипотеза была корректна. Так, для русского языка было выделено около 4200 существительных и столько же глаголов, то есть полнота метода составила около 5% для существительных и около 20% для глаголов (в используемом морфологическом словаре имелось 83000 существительных и 26400 глаголов). Около 85% кластеров были оценены как удачные, то есть все слова в них могли рассматриваться как относящиеся к одной семантической группе (например, «абсурд, бессмыслица, нелепица, околесица», «авианосец, дредноут, крейсер, линкор, миноносец, эсминец»). Еще около 5% кластеров были помечены как сомнительные (например, «аромат запах привкус», «беспринципность ненадежность несамостоятельность»). Остальные примерно 10% кластеров оказались неудачными (например, «антисанитария, белиберда, галиматья, неприличие, несогласованность»). Однако большое количество кластеров следовало бы объединить между собой в более крупные группы (например, кластеры, содержащие названия государств, одежды, еды, материалов и т.д. – всего порядка 10% слов).

Таким образом, для приведенных примеров f_1-мера составит всего 0,095 для существительных и 0,32 для глаголов. Однако если считать, что в выдаче участвовали все слова, то f_1-мера будет равна 0,92. Если вести расчет слов, не попавших в свой кластер (порядка 20%, в качестве правильного кластера брался кластер максимального размера), f_1-мера составит 0,88. Последняя цифра значительно превосходит другие результаты, полученные в данной области. Так, например, в работе [12] было получено значение f-меры, равное 0,44. Кластеризации подвергались 200 глаголов, разбитые в золотом стандарте на 17 классов. В работе [13] на тех же данных было получена точность 64% (при 100% покрытии значение f-меры должно было составить 0,78). Таким образом, можно сказать, что полученные результаты не только превосходят опубликованные ранее для английского языка, но их превышают их по объему рассматриваемой лексики.

Таким образом, высказанные нами предположения оказались в той или иной сетпени верны. Однозначно можно утверждать, что предположение о том, что слова со сходным набором семантических параметров встречаются с примерно одинаковым набором слов в одних и тех же валентностях, оказалось верным для большого списка слов. Справедливости рад следует заметить, что данный список может варьироваться в пределах группы, так как сходные слова разделяются предложенным методом на несколько кластеров. Кроме того, метод оказался применим лишь для 5-20% рассматриваемой лексики. Для остальных слов всё ещё не хватает статистики или точности работы метода.

Полученное разбиение слов может быть использовано в качестве основы для построения онтологии предметной области. Следует заметить, что эксперименты с текстами, выбранными для одной узкой предметной области, не проводились, то есть метод был проверен для онтологии общей направленности.

Кроме того, в качестве исходных данных использовались слова, приведенные к нормальной форме. В этом случае из фраз «ехать на юг» и «ехать на юге страны» будет выделено одно и то же сочетание, тогда как они будут выражать разные

валентности. Более того, оно успешно сравнится с сочетанием «опаздывать на поезд», содержащим третий тип валентности. Вероятно, имеется возможность повысить качество кластеризации, если брать в качестве исходных данных еще и информацию о падеже зависимого существительного. Однако данное предположение должно быть проверено в ходе будущих экспериментов.

Список литературы

1. Рубашкин В.Ш., Бочаров В.В., Пивоварова Л.М., Чуприн Б.Ю. Опыт автоматизированного пополнения онтологий с использованием машиночитаемых словарей // Компьютерная лингвистика и интеллектуальные технологии: По материалам ежегодной Международной конференции «Диалог» (Бекасово, 26-30 мая 2010 г.). Вып. 9 (16). - М.: Изд-во РГГУ, 2010. - сс. 413-418
2. Ефремова Н., Большакова Е., Носков А., Антонов В. Терминологический анализ текста на основе лексико-синтаксических шаблонов // Компьютерная лингвистика и интеллектуальные технологии: По материалам ежегодной Международной конференции «Диалог 2010» - М.: Изд-во РГГУ, 2010. С.124-130.
3. Пивоварова Л., Ягунова Е. Извлечение и классификация терминологических коллокаций на материале лингвистических научных текстов // Терминология и знание: Материалы II Международного Симпозиума – М., 2010.
4. Тестелец, Я.Г. Введение в общий синтаксис / Я.Г. Тестелец. — М.: Издательство РГГУ, 2001. — 800 с.
5. Большой энциклопедический словарь: Языкознание / Гл. ред. В.Н. Яцева. М, Большая российская энциклопедия, 1998. — 685 с.
6. Русские глагольные предложения: Экспериментальный синтаксический словарь / Под общ. ред. Л. Г. Бабенко. М.: Флинта: Наука, 2002. — 462 с.
7. Плунгян В.А., Рахилина Е.В. Сирконстанты в толковании предиката? // Z. Saloni (red.). Metody formalne w opisie języków słowiańskich. Warszawa, 1990, s.201-210
8. Храковский В.С. Понятие сирконстанта и его статус // Семиотика и информатика. – М., 1998. – Вып. 36. – с. 141-153.
9. Клышинский Э.С., Кочеткова Н.А., Литвинов М.И., Максимов В.Ю. Метод разрешения частеречной омонимии на основе применения корпуса синтаксической сочетаемости слов в русском языке // Научно-техническая информация. Сер. 2: Информационные системы и процессы. №1 2011 г., сс. 31-35
10. Гурбанов Т.П., Клышинский Э.С. Параллельный алгоритм составления словаря глагольного управления для новостных текстов на английском языке // Сб. трудов 15 научно-практического семинара «Новые информационные технологии», М., 2011.
11. Клышинский Э.С., Кочеткова Н.А. Метод автоматической генерации модели управления глаголов русского языка // В кн.: Тринадцатая национальная конференция по искусственному интеллекту с международным участием КИИ-2012. Том 1. Белгород: БГТУ им. В.Г. Шухова, 2012. С. 227-235.
12. Lippincott T., Séaghdha D., Korhonen A. Learning Syntactic Verb Frames Using Graphical Models // In Proc. of 50th Annual Meeting of Association for Computational Linguistics 2012, pp. 420-429
13. Schulte im Walde, S. (2000). Clustering verbs semantically according to their alternation behavior // In Proc. of the 18th International Conference on Computational Linguistics (COLING2000), Saarbrücken, Germany, pp. 747–753.

Машинная лингвистика: от перевода со словарём к нелинейным динамическим системам

Л.Л. Волкова
Кафедра «Информационные технологии и автоматизированные системы» Московского института электроники и математики Научно-исследовательского университета «Высшая школа экономики»
liliyavolkova@itas.miem.edu.ru

Аннотация. В статье дан краткий обзор ключевых этапов развития машинной лингвистики в разрезе анализа и синтеза текста. Выделены проблемы работы с языком, являющиеся фундаментальными ограничениями, отделяющими существующий уровень развития отрасли от качественно нового. Рассмотрены перспективные теории, предлагающие новый подход к рассмотрению языка и открывающие возможность заглянуть за барьер машинной лингвистики.
Ключевые слова: машинная лингвистика, анализ текста, автоматическая обработка текста, динамические системы, синергетика, фракталы, тесный мир

Машинная лингвистика, беря начало в пятидесятые с первых опытов машинного перевода, бурно развивается, производя новые методики автоматической обработки текстов, в тесной связи с аппаратами дискретной математики, дискриминантного анализа, задач оптимизации. Однако отрасль вплотную подходит к барьеру, для перехода через который нужно создать качественно новые системы, позволяющие иначе взглянуть на язык. Речь о переходе на более высокий по сравнению, в частности, с синтаксическими правилами уровень, на котором строение языка будет рассматриваться с точки зрения более общих законов, с естественно-языкового ракурса. Это хорошо подмечено в [5]: «Проблема не в том, что имеющиеся теории находят неверные ответы на поставленные ими вопросы, а в том, что они неверно ставят сами вопросы. Нужен подход, который мог бы дать разумный ответ на вопрос: «Как организован язык для выражения значения?», а не на вопрос: «Как организованы синтаксические структуры, если их рассматривать изолированно?»

Предыстория машинной лингвистики состоит из трудов составителей словарей и сводов правил для каждого языка, представляющих собой входные данные для многих методов обработки языков и текстов. Этот краеугольный камень заложен в фундамент отрасли, эклектичное строение которой кропотливо возводится и перестраивается инженерами и по сей день при самом деятельном участии лингвистов. Первые отголоски надвигающейся эпохи машинного перевода раздались ещё в XVII в., когда Лейбниц и Декарт выдвинули предположение о существовании некоего кода, соединяющего между собой слова разных языков [6]. Впервые мысль собственно о возможности машинного перевода высказал Чарльз Бэббидж, разработавший в 1836-1848 гг. проект механической цифровой аналитической машины: он предложил использовать память объемом в тысячу 50-разрядных десятичных чисел (по 50 зубчатых колес в каждом регистре) для хранения словарей [7]. Начало же машинной лингвистике было положено в 50-е гг. XX века с возникновением задачи машинного перевода. Так с создания подстрочников, сначала с крайне ограниченным словарём и минимальным количеством правил, в дальнейшем дополнявшихся и обрабатывавшимися постоянно совершенствующимися методиками в США [8], Советском Союзе [9], а также Великобритании, Канаде, Франции и Германии, началась машинная лингвистика.

При изучении языка и его формализации возникло разделение языков на формальные и естественные. Данные термины можно объяснить соответственно как созданные человеком и без его участия [3]. В зависимости от полноты информации о языке различают системы с ограниченным естественным языком (ОЕЯ) и естественно-языковые (ЕЯ) системы. Системы с ОЕЯ обладают знанием об ограниченном количестве лингвистических конструкций, работают с понятиями только данной предметной области. Как следствие, в них разбирается только некоторое подмножество предложений естественного языка. ЕЯ-системы лишены подобных недостатков, однако чёткой демаркационной линии пока проведено не было. Справедливо считается, что на данный момент созданы лишь ОЕЯ-системы, вплотную приближающиеся к решению ЕЯ-задач [1].

При употреблении термина «формальный язык» подразумевается изучение и описание искусственных языков, созданных людьми для специальных целей, к примеру, языков программирования. Но, как и естественные языки, они подчиняются грамматическим правилам и имеют неоднозначности [2]. Изучение формальных языков с математической точки зрения – предмет дискретной математики и теории формальных языков. Тогда как развитие классической («непрерывной») математики было обусловлено в первую очередь решением задач естествознания, главным образом физики, «дискретная» математика развивалась в связи с изучением законов и правил человеческого мышления, что и обусловило её применение в тех областях техники, которые так или иначе связаны с моделированием мышления. Мышление реализует себя в первую очередь в языке.

Теория формальных языков (ТФЯ) опирается на аппарат, базирующийся на понятии алгебраической структуры кольца [2], близкий к аппарату линейной алгебры, и на теорию графов. Многие задачи теории языков (например, задача определения языка конечного или магазинного автомата) сводятся к задаче о путях во взвешенных (размеченных) ориентированных графах, где множество меток имеет алгебраическую структуру полукольца. ТФЯ включает в себя теорию конечных автоматов и регулярных языков, а также контекстно-свободных языков, последние являются основой многих информационных технологий. Фундаментальным здесь является понятие магазинного автомата – механизма распознавания в классе контекстно-свободных языков, служащего математической основой технологий разработки синтаксических анализаторов для языков программирования. А от синтаксического анализатора недалеко до семантики формальных языков.

Согласно [2], можно выделить 3 аспекта изучения языка.

1) Синтаксис языка – рассмотрение языка как множества слов, которые являются последовательностями букв – символов фиксированного алфавита, коими могут быть как буква алфавита формального или естественного языка (таким буквам соответствуют слова в привычном понимании), так и предложения ЕЯ (или программы на языке программирования) в случае постулирования буквы как целого слова (лексемы). При этом не каждая последовательность букв будет словом, или лексемой, языка: корректность здесь обеспечивается синтаксисом языка, т.е. системой правил. Поскольку каждое слово характеризуется определённой структурой, специфичной для данного языка, необходимы механизмы перечисления, или порождения, слов и механизмы проверки принадлежности слова языку. Эти механизмы в первую очередь изучает ТФЯ.

2) Семантика языка предполагает сопоставление словам языка некого смысла. К примеру, математическая формула несёт в себе не только корректность записи, но и

некое значение. При соблюдении корректного синтаксиса можно создать случайный, некорректный смысл. Математическое понятие теории «смысла» появились сравнительно недавно.

3) Прагматика языка связана с теми целями, которые ставит перед собой человек, при этом они могут не совпадать с пространством семантических смыслов текста (к примеру, получение за речь некой суммы денег). Это дисциплина социально-философская, затрагивающая целеполагающую деятельность личности. Теорией формальных языков она не рассматривается.

Первой задачей обработки текста является определение его структуры. Только после перевода текста на естественном языке в его формальное представление, пригодное для машинной обработки [11], текст можно подвергать дальнейшей обработке и более глубокому анализу, выявлению закономерностей и пр. Здесь возникает вопрос о том, какие есть уровни изучения текста. По В.А. Звегинцеву, можно выделить следующие уровни изучения текста: дискурс (связанный текст); предложение; словосочетание; слово; морфема; слог; фонема; дифференциальный признак [12]. В системах автоматической обработки текста используются в основном первые пять уровней. Слоги иногда используются при представлении информации в морфологических словарях.

Установление структуры текста, или анализ в структурном смысле, включает в себя начальные, базовые этапы обработки документации [1]:

1) графематический анализ;
2) морфологический анализ;
3) предсинтаксический анализ;
4) синтаксический анализ;
5) постсинтаксический анализ;
6) семантический анализ.

Каждый этап представляет собой анализ определённого уровня абстракций текста: графематический – анализ графем, т.е. синтаксических и структурных единиц, морфологический – определение морфологических атрибутов и нормальных форм (т.е. исходной формы каждого слова, от которой образована данная словоформа), синтаксический – определение ролей слов и их связей между собой (существует множество лингвистических критериев, по которым могут быть выделены связи между отдельными словами в предложении [29]), семантический – анализ смысла текста; предсинтаксический и постсинтаксический анализ призваны выполнить сопутствующие синтаксическому анализу задачи, к примеру, снять неоднозначность (в частности, омонимию, некоторые механизмы снятия синтаксической и морфологической омонимии см. в [10, 32, 30, 28]), объединить или разъединить лексические единицы, в том числе разнесённые в предложении, и таким образом повысить качество синтаксического анализа. Современные системы проводят анализ текста поэтапно таким образом, что на вход каждого следующего этапа поступают результаты обработки предыдущего этапа. Самое высокое качество анализа текста на естественном языке получается при полном анализе текста.

Анализ, или разбор, текста – это прямое направление обработки текста. С ним тесно связано обратное направление – синтез текста, чьи этапы обратны этапам анализа. Анализ текста используется более всего: текст перед практически любой обработкой требует разбора – для извлечения информации (выделение ключевых слов и маркеров, определение эмоциональной окраски, заполнение фреймов по содержанию и др.), классификации и кластеризации, выявления характерных

особенностей текстов. Синтез же (следуя за анализом) используется в таких целях, как машинный перевод (здесь особенности языка учитываются в процессе анализа и синтеза, а внутреннее представление текста строится унифицированным и языконезависимым, чтобы синтезировать текст по правилам конкретных грамматики и синтаксиса) и генерация отклика диалоговых и рекомендательных систем.

Начальные этапы обработки текстов реализуются при помощи различных подходов и методик (к примеру, деревья, а также разработанный сотрудником IBM Джоном Бэкусом метаязык для описания языков программирования, по сути, мало чем отличающийся от контекстно-свободных грамматик [34, 35], который, будучи уточнён датским учёным Питером Науром, был назван формой Бэкуса-Наура, или бэкусовской нормальной формой [37, 38]). И анализ, и синтез текстов требуют повышения качества, которое достигается зачастую разработкой методик ускорения и оптимизации работы предсинтаксического и постсинтаксического анализа (к примеру, [41, 42]). Впрочем, все этапы начальной обработки текста открыты для доработки и переработки. На этапе морфологического анализа в некоторых случаях для повышения производительности системы выделяется только неизменяемая часть токена (стемматизация) [39, 31, 33], однако для систем с полным анализом текста такой информации недостаточно, поэтому в таких системах на этапе морфологии выделяются все возможные лексические характеристики каждого токена [36].

Машинный перевод сегодня не сводится к простой замене слов их иностранными аналогами, как это было когда-то. Существуют набирающие популярность за счёт сравнительной дешевизны улучшения результата статистические системы и системы, основанные на анализе текста. Впрочем, эти системы зачастую используют на разных этапах функционал друг друга. Текст документа разбирается «на запчасти», и на основании выделения зависимостей и подчинения в нём осуществляется перевод. К примеру, отечественные ученые из института Прикладной Математики им. Келдыша разработали свою нотацию для системы автоматического перевода «Crosslator 2», которая является расширением форм Бэкуса-Наура и позволяет учитывать согласование слов в предложении [41, 43, 40]. В данной нотации слово в предложении на естественном языке представляет собой входной символ грамматики, в котором указана нормальная форма слова и набор морфологических атрибутов [45, 44]. То есть осуществляется переход от естественного языка к его формальному представлению, анализ, решение прикладной задачи и синтез результата.

Существенным является доступ к базам корпусов языков и размеченных текстов для использования в программных средствах и машинного обучения. В этом смысле время работает на лингвистов, поскольку появляются новые размеченные корпуса текстов, дающие возможность для проведения исследований, тренировки и обучения методов [21].

В настоящее время большой удельный вес имеет статистический подход. Обучившись на внушительной выборке данных, порой меньшими затратами достигается сопоставимый со структурными, проникающими в суть конструкции языка методами результат. Уже не редкость системы с высоким качеством (хотя оценки разнятся на разных корпусах, как отмечено, к примеру, в [22]) и точностью порядка 80 % и даже 90 % (в частности, это отражено в результатах РОМИП – ежегодного Российского семинара по Оценке Методов Информационного Поиска [23]), и борьба подчас идёт за доли процентов.

Однако перспективы с точки зрения развития языка такой «плоский» подход не имеет, поскольку язык – материя гибкая и развивающаяся, что отражается в статистике лишь со временем, уже де факто. Будущее за системами де юро, синтезирующими знание, в противоположность решению насущных задач здесь и сейчас, достаточному, но не удовлетворительному с точки зрения фундаментальной науки. Статистические методы, вносящие огромный вклад в машинное обучение и выявление числовых закономерностей в текстах, скорее инструмент познания, но синтезирующей функции в своей базе они не несут.

С большим трудом преодолеваются, чаще, увы, статистически, такие «граничные условия», как снятие неоднозначностей (большой процент ошибок разбора текстов приходится на омонимию), непрозрачность в связи с всё ещё недостаточным развитием семантического анализа связей в текстах, ошибки в данных (Errare humanum est, или как обрабатывать слова с четырьмя ошибками в трёх буквах), наконец. Неизвестные слова – новые, заимствованные, имена собственные, окказионализмы. Даже в статистическую систему, способную, предположим, собрать в корпусе текстов информацию об употреблении нового термина или, более того, становящегося популярным окказионализма – слова или даже модели управления (к примеру, непереходный глагол с переходным управлением: «его ушли с работы»), – нужно вложить знания о механизмах работы языка. А это уже рассмотрение по существу, где сбор статистики лишь вспомогательный инструмент (при этом не опорный для увечной разработки).

Пусть в ходе конвергентной эволюции язык подчас становится похож на другой, не родственный ему вид, – развитие и внутренние законы всё же специфичны для каждого конкретного языка. И именно умение синтезировать новые правила и, сообразуясь с ними, знания – та черта, которая отличит отечественные и зарубежные системы, сиюминутно функционирующие по принципу «авось» (см. известный пример про трёх белых котят и одного политкорректно чёрного), от подлинных демиургов машинной лингвистики, чей приход мы с нетерпением ждём и по мере сил стараемся приблизить.

А чтобы разработать таких демиургов (др.-греч. Δημιουργός – мастер, ремесленник, творец), то есть хорошие системы из гильдии машинной лингвистики, усвоившие секреты ремесла, ноу-хау языка, а именно модели формирования языковых конструкций и их генерации, нужно потрудиться над умением применять, агрегировать и синтезировать системы на правилах (если не сказать «правильные» системы) функционирования языка и тем самым подняться на новую ступень на пути от систем с ограниченным естественным языком к естественно-языковым системам, лишённым многих ограничений [1] именно благодаря большей свободе синтеза. Тогда мы, повернув вспять ход этимологической истории термина «демиург», начнём новый виток спирали: обратно от концепции всемогущего творца к именованию класса ремесленников, как это и было в античном обществе.

Впрочем, предсказуемость языковых конструкций будет ограничена, какими бы оптимистичными ни были взгляды неофитов, энтузиастов и экспертов по искусственному интеллекту. Язык, возможно, является **нелинейной динамической системой**.

Как отмечено в [4], до 60-х гг. предполагалось, что есть два класса процессов. Первые описываются динамическими системами, в которых будущее однозначно определяется прошлым, вторые – системами, в которых будущее не зависит от прошлого. В первых, детерминированных, предполагалась полная предсказуемость,

во вторых – полностью случайные события. В 70-е гг. был выделен третий класс процессов: формально они описываются динамическими системами, но их поведение может быть предсказано только на небольшой промежуток времени. Ограниченность горизонта прогноза связана с чувствительностью к начальным данным (малые изменения могут повлечь лавинообразные изменения и, в конечном итоге, катастрофу) и с непериодическим движением в детерминированных системах, т.е. динамическим хаосом. Траектории движения частиц расходятся со скоростью, определяемой так называемым ляпуновским показателем. Но те же явления можно отметить и в историческом движении языка и его частиц.

Язык развивается вместе с человеком, он неразрывно связан с коллективным опытом и сознанием [24]. Под воздействием миграций, войн, научно-технического прогресса (в частности, появления радио, телевидения, компьютеров, интернета) языки видоизменяются, приобретая новые черты [25]. И некоторые факторы, к примеру, переселение народов [26], могут вызвать лавинообразные изменения в языке, будь то орфография или семантика. Даже символьные обозначения, как, например, путь от египетских иероглифов и их наследников, финикийской клинописи, к символьной записи иврита, от которой произошло арабское письмо, и греческому алфавиту, от которого, в свою очередь, произошли кириллица с одной стороны и латиница с другой. Подобные явления происходят в пространстве смыслов слов, моделей управления и, далее, построения речи. А каждое изменение входных данных влечёт всё большую разницу между выходными результатами, то есть языками и правилами в них [26].

Один из основных признаков системы состоит в её структурированности, в целесообразности связей между её элементами [3]. Структурные взаимосвязи в тексте, результате функционирования сложной системы языка, описываются, в первую очередь, синтаксисом и семантикой. На примере их сложности и подчас лавинообразного роста вариантов можно увидеть их сложность и нелинейность. Рассмотрим ряд приложений автоматической обработки текстов. В частности, рассмотрим систему, производящую по словоформе поиск нормальной формы слова. В системах, в которых поиск организован в упорядоченном по частотности дереве (по первой букве по убывающей частоте, по второй букве и т.д.), скорость нахождения допустимых словоформ выше, чем при поиске в словаре, а главное, в них высвечивается такая проблема, как увеличение допустимых конструкций на порядки при увеличении длины строки (при генерации всех возможных цепочек из алфавита рост количества допустимых цепочек, если грамматика не учитывает правил отбраковки некорректных для человеческого языка цепочек, будет экспоненциальным). А поскольку язык развивается, такой словарь словоформ никогда не будет полным. Взять хотя бы приставки и суффиксы, которые приводят к катастрофическому разрастанию и без того немалого «ядра» языка в виде стемов до бесконечного количества словоформ (будем считать это на данный момент машинной бесконечностью, поскольку система, владеющая всеми порождающими правилами для корректных цепочек языка, не существует). При этом если говорить только о рассмотрении грамматически верного текста, то на существующих базах правил ещё можно получать хорошие результаты. А теперь возьмём тексты с ошибками. Здесь требуются методы предположения об ошибке. Пусть мы допускаем 1-3 ошибки. А если их больше, речь может идти не только о неграмотности (здесь можно было бы положить, что настолько неграмотные тексты разбору не подлежат), но и о словах, не наличествующих в словаре. К ним могут относиться и новые термины, и

малоупотребимые слова, а если речь идёт не о сухих текстах с минимумом лексики, а о художественной литературе или публицистике, то и объём словаря для полного разбора требуется больший, и редких слов больше, а окказионализмы и подобные им явления в словаре всё равно не фиксируются. Следовательно, разбор требует иного, не словарного подхода, а скорее словообразовательного, далее – образующего предложения, семантические структуры и т.д.

Словообразование само по себе представляет собой нелинейную систему с изменяющимися и достаточно свободно соединяющимися структурными элементами, то же касается и синтаксиса. Опущенные члены предложения существенно усложняют разбор, а вкупе с растущим за счёт неоднозначностей (число которых дополнительно увеличивается за счёт отсутствующих узлов дерева разбора) количеством претендентов на корректный вариант разбора задача синтаксического анализа усложняется дополнительно, что показывает, насколько несовершенны ещё методы разбора, при всех существенных достижениях по улучшению качества разбора.

Многое в строении и истории развития естественных языков свидетельствует о том, что человеческие языки являются носителями черт нелинейных динамических систем. Возможно, развитие теории синергетики и прогноза [4] претворит в реальность пророчество Анри Пуанкаре о том, что в будущем можно будет предсказывать новые физические явления, исходя из общей математической структуры описывающих эти явления уравнений. Возможно, и развитие человеческих языков тоже можно будет предсказать и полностью описать математической моделью. Дело за экспериментами и фундаментальными исследованиями.

Новые теории в машинной лингвистике

Как отмечается в [3], одним из направлений перестройки в высшем образовании и систематизации науки является преодоление недостатков узкой специализации, усиление междисциплинарных связей, развитие диалектического видения мира, системного мышления.

Анализ текстов на сегодняшний день в некотором роде представляет собой работу с фреймовой моделью [27]: предложения и языковые единицы рассматриваются по некоторому ограниченному множеству шаблонов в рамках стереотипных ситуаций. В слоты фрейма (возможно, содержащие ссылки на другие фреймы) записываются извлечённые данные о структуре слов (выделяются начальные формы и заполняется список параметров в зависимости от части речи), предложений (синтаксические роли и связи), содержания (переход к семантике; сейчас решаются такие задачи, как извлечение информации из текстов и заполнения слотов фрейма: к примеру, для близкой к некому рассматриваемому шаблону статьи о государстве из текста выделяются сведения о площади, численности населения, климате и т.д.). Однако возможность развития системы правил ограничена самой шаблонностью такого подхода. «Кругозор» такой системы расширяется разовым пополнением базы правил, и это всё же система с ограниченным естественным языком.

Существуют новые для машинной лингвистики математические теории, перспективные для познания устройства языка, в частности, теория фракталов и теория тесного мира.

Фракталы

Теория фракталов, как правило, рассматривается как подход к статистическому исследованию, который позволяет получать важные характеристики информационных потоков, не вдаваясь в детальный анализ их внутренней структуры и связей. Одним из основных свойств фракталов является самоподобие (скейлинг). Как показано в работах С.А. Иванова [20], для последовательности сообщений тематических информационных потоков в соответствии со скейлинговым принципом количество сообщений, резонансов на события реального мира пропорционально некоторой степени количества источников информации (кластеров) и итерационно продолжается в течение определенного времени. Точно так же, как и в традиционных научных коммуникациях, растущее множество сообщений в Интернет по одной тематике во времени представляет собой динамическую кластерную систему, возникающую в результате итерационных процессов. Этот процесс объясняется републикациями, прямой или совместной цитируемостью, различными публикациями – отражениями одних и тех же событий реального мира, прямыми ссылками и т.д. Кроме того, для большинства тематических информационных потоков наблюдается увеличение их объемов, причем на коротких временных интервалах рост линейный, а на длительных – экспоненциальный.

Фрактальная размерность в кластерной системе, соответствующей тематическим информационным потокам, показывает степень заполнения информационного пространства сообщений в течение определенного времени. Между количеством сообщений и кластеров проявляется свойство сохранения внутренней структуры множества при изменении масштабов его внешнего рассмотрения. По мнению С.А. Иванова, все основные законы научной коммуникации, такие как законы Парето, Лотки, Бредфорда, Зипфа, могут быть обобщены именно в рамках теории стохастических фракталов.

Исследование информационного потока в его развитии даёт возможность проследить изменения в корпусах текстов и в языке. При проведении информационно-аналитических исследований на основе обработки информационного потока, формируемого в Интернет [17], особо актуальной оказывается задача автоматического извлечения из текстов фактографической информации [18]. При этом ввиду значительных объемов и динамики информационных потоков контент-анализ осуществляется сегодня с использованием современных информационно-аналитических систем. По наблюдению Д.В. Ландэ [15], следует признать, что изначальные парадигмы поисковых систем и контент-анализа, сформированные десятилетия тому назад, уже не отвечают реальной ситуации. Один из подходов к решению задачи извлечения фактов из текстовых документов и выявления их взаимосвязей базируется на технологии контент-мониторинга, который можно рассматривать как непрерывный во времени содержательный анализ информационных потоков с целью получения необходимых качественных и количественных информационных срезов. Именно непрерывная аналитическая обработка сообщений является самой характерной чертой этого подхода, который позволяет извлекать факты из тестов, выявлять новые понятия, формировать разнообразные статистические отчеты. Названные задачи сегодня охватываются двумя основными технологиями – извлечением фактографической информации из текстов (Information Extraction [18]) и глубинным анализом текстов (Text Mining [19]).

Однако, основываясь на структуре текста, можно предположить, что фрактальные свойства присущи и внутреннему устройству текста. Самоподобие в

текстах на естественном языке не формализовано, и это требует глубинного исследования в тесной связи с правилами построения текста.

Теория тесного мира

Проведённые недавно испанскими и американскими учёными исследования показали, что свойства тесного мира (англ. small world), наблюдаемые во многих биологических и социальных системах, присущи и текстам [46]. Более того, построение тесного мира высвечивает такие свойства текста, как самоподобие и стойкие тенденции связности слов [47-49].

Слова в человеческом языке взаимодействуют в предложениях не случайным образом: они связаны. Человек способен построить огромное количество предложений из ограниченного количества структурных единиц, процесс построения крайне быстр и всё ещё недостаточно изучен, чтобы создавать достаточно близкие к человеческой способности прототипы генерирующих речь систем. Совместное употребление слов в предложениях отражает организацию языка и, таким образом, может способствовать его описанию в терминах графа взаимодействий слов. Такие графы, как показано в [46], отражают два важных свойства, открытых для различных по происхождению систем:

1) эффект малого мира (в частности, среднее расстояние между двумя словами, или среднее минимальное количество дуг (связей) на пути от одного слова к другому, оценивается в пределах от 2 до 3, хотя человеческий мозг способен хранить тысячи и тысячи таких связей);

2) новое слово (вершина) при расширении графа будет связано с существующим словом с вероятностью, пропорциональной количеству связей последнего с другими словами.

На основании исследований в [46] показано, что существуют доселе неизвестные особенности организации языка, которые могут отражать эволюцию и социальную историю языков и истоки их гибкости и комбинаторной природы. Таким образом, развитие данного направления исследований перспективно для углубления знаний о структуре языка.

Свойство самоподобия информационного пространства (см. выше) [13-15], может быть более подробно изучено по графам тесного мира. Выявление самоподобия в текстах и информационных потоках позволит повысить эффективность анализа текстов, выявления зависимостей и извлечения информации, что не только полезно для решения прикладных задач компьютерной лингвистики, но и видится существенным подспорьем в установлении общих законов языка и построения речи. Таким образом, как отмечено в [50], теория тесного мира позволит выявлять и изучать особенности построения текста, а также его семантическую структуру.

В заключение стоит сказать, что рассмотрение языка как нелинейной динамической системы (см. ранее) также может внести весомый вклад в машинную лингвистику. Основываясь на прочном фундаменте проведённых исследований языка и текстов и разумно используя новые для машинной лингвистики взгляды на естественно-языковой мир, найдя тот кардинально отличающийся подход к рассмотрению использования аппарата языка, мы, в лучших традициях Карла Сагана, совершим переход от последовательности простых чисел к объёмному восприятию информации, дающей нам ключ к удивительным путешествиям к центру Знания о Языке.

Список литературы

1. Автоматическая обработка текстов на естественном языке и компьютерная лингвистика : учеб. пособие / Большакова Е.И., Клышинский Э.С., Ландэ Д.В., Носков А.А., Пескова О.В., Ягунова Е.В. — М.: МИЭМ, 2011. — 272 с.

2. Белоусов А.И., Ткачев С.Б. Дискретная математика: Учеб. для вузов / Под ред. В.С. Зарубина, А.П. Крищенко. — 4-е изд., исправл. — М.: Изд-во МГТУ им. Н.Э. Баумана, 2006. — 744 с. (Сер. Математика в техническом университете; Вып. XIX).

3. Перегудов Ф.И., Тарасенко Ф.П. Введение в системный анализ: Учеб. пособие для вузов. — М.: Высш. шк., 1989. — 367 с.

4. Малинецкий Г.Г. Синергетика и прогноз. // Будущее прикладной математики. Лекции для молодых исследователей / Под ред. Г.Г. Малинецкого. — М.: Едиториал ЦРСС, 2005. — 512 с. — С. 374-403.

5. Виноград Т. Программа, понимающая естественный язык. — М.: Мир, 1976. — 296 с.

6. Hutchins, J. Machine Translation: past, present, future. — Chichester: Ellis Horwood, 1986.

7. Апокин И.А., Майстров Л.Е., Эдлин И.С. Чарльз Бэббидж. — М.: Наука, 1981. — 128 с.

8. MacDonald N. Language translation by machine - a report of the first successful trial. // Computers and automation. 1954, v. 3, № 2, p. 6-10.

9. Кулагина О.С. Исследования по машинному переводу. — М.: Наука, 1979. — 320 с.

10. Мельчук И.А. Опыт теории лингвистических моделей «Смысл→Текст». — М.: Школа «Языки русской культуры», 1999. — I-XXII, 346 с.

11. Крысин Л.П. Лингвистический процессор для сложных информационных систем. — М.: Наука. — 1992.

12. Попов Э.В. Общение с ЭВМ на естественном языке. — М.: Наука, 1982. — 360 с.

13. Ландэ Д.В. Фрактальные свойства тематических информационных потоков из Интернет. // Регистрация, хранение и обработка данных. — Киев, 2006. — Т. 8, No 2. — С. 93 - 99.

14. Додонов А.Г., Ландэ Д. В. Самоподобие массивов сетевых публикаций по компьютерной вирусологии. // Реєстрація, зберігання і обробка даних, 2007, Т. 9, — N 2. — С. 53-60.

15. Ландэ Д. Фракталы и кластеры в информационном пространстве. // Корпоративные системы, №6'2005. — С. 35-39.

16. Ландэ Д.В. Выявление понятий и их взаимосвязей в рамках технологии контент-мониторинга. // Реєстрація, зберігання і обробка даних, 2006, - Т. 8, - N 4.

17. Ландэ Д.В. Основы интеграции информационных потоков. — К.: Інжиніринг, 2006. — 240 с.

18. Ralph Grishman. Information extraction: Techniques and Challenges. In Information Extraction (International Summer School SCIE-97) // Springer-Verlag. — 1997.

19. Гершензон Л. М., Ножов И. М., Панкратов Д. В. Система извлечения и поиска структурированной информации из больших текстовых массивов СМИ. Архитектурные и лингвистические особенности. Труды Международного семинара «Диалог’2005» (Звенигород, 1–6 июня 2005 г.). — М.: Наука, 2005.

20. Иванов С.А. Стохастические фракталы в Информатике. Научно-техническая информация. — Сер. 2. — 2002. — № 8. — С. 7–18.

21. Баранов А.Н. Введение в прикладную лингвистику. — М.: Издательство ЛКИ, 2007. — 360 с.

22. Sharoff S., Nivre J. The proper place of men and machines in language technology. Processing Russian without any linguistic knowledge. // Компьютерная лингвистика и интеллектуальные технологии: По материалам ежегодной Международной конференции «Диалог» (Бекасово, 25 - 29 мая 2011 г.). Вып. 10 (17). — М.: Изд-во РГГУ, 2011.

23. Маслов М. Ю., Пяллинг А.А. КС-классификатор и дорожка классификации веб-сайтов РОМИП'2010. // Труды РОМИП 2010. Под ред. И.С. Некрестьянова. — Казань, 2010. — 200 с. — С. 80-97.

24. Абаев В. И. О термине «естественный язык». // Вопросы языкознания. — М., 1976, № 4. — С. 77-80.

25. Практическая транскрипция личных имен в языках народов мира / [отв. ред. Э.С. Клышинский]; Ин-т прикладной математики им. М.В. Келдыша РАН. — М.: Наука, 2010. — 679 с.

26. Плунгян В.А. Почему языки такие разные. — М.: АСТ-ПРЕСС КНИГА, 2010. — 272 с.

27. Минский М. Фреймы для представления знаний. — М.: Энергия, 1979. — 152 с.

28. Ножов И.М. Реализация автоматической синтаксической сегментации русского предложения. Диссертация на соискание ученой степени кандидата технических наук, М.: РГГУ, 2003.

29. Тестелец Я.Г. Введение в общий синтаксис. М.: РГГУ, 2001 г. 798 с.

30. Neumann G., Brauny C., Piskorski J. A Divide-and-Conquer Strategy for Shallow Parsing of German Free Texts // ANLC '00 Proceedings of the sixth conference on Applied natural language processing, 2000, pp. 239-246.

31. Porter M.F. An algorithm for suffix stripping // Program, 1980, Vol. 14, №3, pp. 130-137.

32. Przepiorkowski A., Buczynsk A. Shallow Parsing and Disambiguation Engine. [Proceedings of 3rd Language & Technology Conference, 2007]

33. Segalovich I. A fast morphological algorithm with unknown word guessing induced by a dictionary for a web search engine // [Proceedings MLMTA, 2003]

34. Рассел С., Норвиг П. Искусственный интеллект. Современный подход. Москва — С-т Петербург — Киев: Вильямс, 2006. — 1093 с.

35. Формальные модели анализа и распознавания языковых структур. [Сайт Международной конференций по компьютерной лингвистике «Диалог»].

36. Черненьков Д.М., Клышинский Э.С. Формальный метод пополнения словарей морфологического анализа с использованием несловарной лексики // Вестник компьютерных и информационных технологий, №3, 2011, сс. 22-28

37. Backus J.W. The syntax and semantics of the proposed international algebraic language of the Zurich ACM-GAMM Conference // Proceedings of the International Conference on Information Processing, UNESCO, 1959, pp. 125–132.

38. Knuth D.E. Backus Normal Form vs. Backus Naur Form // Communications of the ACM, 1964, Vol. 7, Issue 12, pp. 735–736.

39. Lovins J.B. Development of a stemming algorithm // Mechanical Translation and Computational Linguistics 11, 1968, pp. 22–31.

40. Востриков А.В., Клышинский Э.С., Морозов С.Н., Манушкин Е.С., Максимов В.Ю. Исследование метода автоматической генерации правил фрагментарного анализа // Сб. тезисов международной конференции MegaLing'08, Партенит, 2008

41. Галактионов В.А., Мусатов А.М., Мансурова О.Ю., Ёлкин С.В., Клышинский Э.С., Максимов В.Ю., Аминева С.Н., Жирнов Р.В., Игашов С.Ю., Мусаева Т.Н. Система машинного перевода «Кросслятор 2.0» и анализ ее функциональности для задачи трансляции знаний // Препринт ИПМ им. М.В.Келдыша РАН. Москва, 2007.

42. Гладкий А.В. Синтаксические структуры естественного языка, Изд. 2 — М.: ЛКИ, 2007. С. 12-15.

43. Жирнов Р.В., Клышинский Э.С., Максимов В.Ю. Модуль фрагментарного анализа в составе системы машинного перевода. Crosslator 2.0 // Вестник ВИНИТИ, 2005 г. НТИ. Серия 2. №8 С. 31-33

44. Клышинский Э.С., Манушкин Е.С. Математическая модель порождения правил синтаксической сегментации // Сб. трудов второй Всероссийской конференции «Знания – Онтологии – Теории», Новосибирск, 2009, Том 2., С. 182-186.

45. Манушкин Е.С., Клышинский Э.С. Метод автоматического порождения правил синтаксической сегментации для задач анализа текстов на естественном языке // Информационные технологии и вычислительные системы, 4, 2009 г., С. 57-66.

46. The small world of human language. Ramon Ferrer i Cancho and Ricard V. Solé. Proceedings of the Royal Society, Series B, Vol. 268, No. 1482, pp. 2261-2266, 2001.

47. Amaral, L. A. N., Scala, A., Barthélémy, M. & Stanley, H. E. Classes of behaviour of small-world networks. Proc. Natl Acad. Sci. USA 97, p. 11149-11152, 2000.

48. Watts, D. J. & Strogatz, S. H. Collective dynamics of “small-world” networks. Nature 393, p. 440-442, 1998.

49. Albert, R., Jeong, H. & Barabási, A.-L. Error and attack tolerance of complex networks. Nature, 406, p. 378-381, 2000.

50. Волкова Л.Л. Приложения теории тесного мира в компьютерной лингвистике. // Сборник трудов третьей Международной научно-практической конференции «Модель подготовки специалистов новой формации, адаптированных к инновационному развитию отраслей». — Душанбе, 2012.

Оглавление

Компьютерная графика

Программная система для анализа и оптимизации вычислительных свойств гибридных разностных схем Burgers2 *А.Е. Бондарев, А.В. Бондаренко, В.А. Галактионов, Т.Н.Михайлова, И.Г. Рыжова* …….. 3
Интеграция модуля компьютерной графики в систему CATIA *Б.Х. Барладян, Н.Б. Дерябин, Л.З.Шапиро* …… 11
Задача позиционирования камеры заднего вида *Е.Д. Бирюков* …… 20

Способы реализации механизма скользящего стерео-курсора в реальном времени *А.Е. Афанасьева, А.В. Игнатенко*……... 25
Определение индекса преломления прозрачного материала с помощью визуализации *В.В. Афанасьев, А.В. Игнатенко, И.О. Тисевич* …… 31
GMLePublish: web-система оценки алгоритмов тональной компрессии HDR-изображений *М.А. Матросов, А.В. Игнатенко* ……... 37
Субъективная оценка алгоритмов тональной компрессии HDR-изображений для образцов ювелирной промышленности с помощью системы GMLePublish *М.А. Матросов, А.В. Игнатенко* ……..……. 43
Пересегментация изображений для применения в задаче трекинга объектов, меняющих форму *К.С. Зипа, С.С.Жулин, А.А. Ильин* …… 51
Доказательство подлинности фотоизображений встраиванием цифровых водяных знаков *Ю.А. Белобокова, Э.С. Клышинский* …… 58

Моделирование процессов

Модель искусственного интеллекта на реляционных множествах *А.П. Гусев* …. 64
Использование генетических алгоритмов в задачах анализа данных и построения систем правил *В.И. Солодовников* …….. 72
Кластеризация пользователей системы дистанционного образования на основе модели нейронной сети ART1 *М.А.Гаврилова, О.Ю.Ерёмин* ……. 78
Исследование многомерных данных в задачах многопараметрической оптимизации *А.Е. Бондарев, В.А. Галактионов* …….. 84
Проблемы построения имитационных моделей для проектирования технологических процессов *А.Ф. Зародов, Н.Г. Ульянов* …….. 93

Автоматизация проектных исследований надежности и стойкости аппаратуры

Методика оценки накопленной дозы функционального узла БРЭА КА при воздействии низкоинтенсивных излучений *М.А. Артюхова, С.Н. Полесский* … 95

Анализ построения локальной вычислительной сети по критерию надёжности на базе «тонких» и «толстых» клиентов *А.А. Гавриленко, С.Н. Полесский* ……... 101
Электронный журнал учета отказов радиоэлектронных средств *Цыганов П.А.* 113
Метод оценки безотказности структурно-сложных РЭУ *А.Н. Тихменев* ……. 116
Модуль расчета показателей безотказности резервированных групп для ПК АСОНИКА-К *А.М. Авхименко, С.Н. Колесников* …… 124

Исследование модели интенсивности отказов пружин скручивания *Монахов М.А., Фокин В.М., Лушпа И.Л.* ……. 128

Сравнительный анализ методов оценки надежности полупроводниковых интегральных микросхем *В.В. Жаднов* …… 132
Сервисная система SaaS-модели программного обеспечения расчетов надежности РЭА В.В. Жаднов …….. 138

Системы автоматизированного проектирования

Моделирование термоэлемента Пельтье для процессов оптимизации систем обеспечения тепловых режимов радиоэлектронных средств *И.Ю.Громов, А.М. Кожевников* …… 147
Метод оптимизации системы радиатор-кулер обеспечения тепловых режимов электрорадиоэлементов *А.М. Кожевников, И.Ю.Громов* …… 153
Математическое моделирование синхронного спирального излучателя *С.С. Захарова* …….. 156
Моделирование диаграммы направленности излучателей, расположенных над металлическим экраном С.С. Захарова …… 163
Эффективная поверхность рассеяния, применимость и возможность ее определения для сложных целей *Т.А. Ясова* …… 170
Неопределенность результатов виртуальной сертификации РЭС по эмиссии излучаемых радиопомех *Н.В. Лемешко* …… 175
Структура виртуального проекта в системе автоматизации проектирования радиоэлектронных средств с функцией виртуальной сертификации по эмиссии излучаемых радиопомех *Н.В. Лемешко* ……... 181
Разработка структуры программных средств системы автоматизации проектирования с функцией виртуальной сертификации радиоэлектронных устройств по эмиссии излучаемых радиопомех *Н.В. Лемешко, С.С. Захарова* 186

Разработка программного обеспечения

Система человеко-машинного взаимодействия на базе акселерометра *Аксёнов К.В.* …….. 191
Особенности построения графического интерфейса для ИПС Серелекс на платформе Windows 8 *А. Романов, П. Романов* ……. 195
Автоматизированное создание тестового регрессионного набора на основе документации процесса тестирования программного обеспечения *А.С. Мартюков* …… 198
Автоматизированное тестирование программных комплексов оптического моделирования *Е.Ю. Денисов* …….. 202
Автоматизированное тестирование интерактивных программных комплексов *Е.Ю. Денисов, И.А. Калугина* …….. 207
Управление требованиями к программному обеспечению для поддержки контроля достижения целей *В.О. Каковский, Т.Н. Заболотняя* …….. 211
Программирование голосом, как необходимость *А.А. Ильский* …….. 221
Проектирование интерфейса и модель пользователя *Иванов С.Ю.* …… 227
Алгоритм равномерного распределения вычислительной нагрузки на кластере методом деревьев нитей *А.П. Гусев* …….. 231

Машинная лингвистика

Автоматическая классификация текстов на основе заголовка рубрики *О.В. Гуляев, Н.В. Лукашевич* …… 238
Кластеризация оценочных слов по тональности на основе Марковских цепей *И.И. Четвёркин* …… 245
Автоматизация оценки благозвучия текстов *К.В. Иванов* …… 253
Синтаксический анализ музыкальных текстов *И. Голубева, А.Ю. Филиппович* 257
Применение методов машинного перевода для анализа древнерусских музыкальных рукописей *М. Даньшина* …… 263
Извлечение переводного словаря значимых словосочетаний из параллельных текстов с использованием методов дистрибутивной семантики *Ю.И. Морозова* …… 268
О синтезаторе таджикской речи по тексту *Х.А. Худойбердиев* …… 273
Программный стенд для исследования глагольных конструкций таджикского языка *Д.Д.Собиров* …… 277
Комплекс средств автоматизации межъязыковых преобразований текстов для центрально-азиатского региона *Л.А. Гращенко* …… 283
О множестве анаграмм и распознавании их элементов *З.Д.Усманов, В.Нормантас* …… 287
Создание «онтологии всего». Проблемы классификации и решения *Г.А. Волкова* …… 293
Статистические языковые методы. Коллокации и коллигации *Н.А. Кочеткова* 301
Анализ методики построения новостного анонса по теме «электронное правительство и электронные услуги» и время жизни информационного повода *А.В. Чижик* …… 306
Кластеризация семантически связанных слов из неразмеченного текста на русском языке *Э.С. Клышинский, Н.А. Кочеткова* …… 311
Машинная лингвистика: от перевода со словарём к нелинейным динамическим системам *Л.Л. Волкова* …… 316
Оглавление …… 329

Под редакцией
ТУМКОВСКОГО Сергея Ростиславовича

Материалы Шестнадцатого научно-практического семинара
“Новые информационные технологии в автоматизированных системах”

ISBN 978-5-94506-335-8

9 785945 063358

Изд. лиц. ИД № 06117 от 23 октября 2001 г. Подписано в печать .03.13.
Формат 60x84/16 . Бумага типографская № 2. Печать - ризография.
Усл.печ. л. 20,75 Уч.-изд. л. 18,7. Тираж 100 экз. Заказ 45 Изд. № .
Московский государственный институт электроники и математики
109028 Москва, Б. Трехсвятительский пер., 3/12.