

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«САМАРСКИЙ ГОСУДАРСТВЕННЫЙ АЭРОКОСМИЧЕСКИЙ  
УНИВЕРСИТЕТ ИМЕНИ АКАДЕМИКА С. П. КОРОЛЕВА  
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)»  
(СГАУ)

Факультет информатики  
Кафедра технической кибернетики

Расчётно-пояснительная записка к курсовой работе  
по дисциплине «Уравнения математической физики»

Тема: «АНАЛИТИЧЕСКОЕ РЕШЕНИЕ КРАЕВЫХ ЗАДАЧ  
МАТЕМАТИЧЕСКОЙ ФИЗИКИ»

Вариант № 5

Выполнил студент	Ионкин М. А.
Группа	6407
Руководитель работы	Быков Д. А.

САМАРА 2015

## ЗАДАНИЕ

Решить задачу теплопроводности на сфере радиуса  $R$ . Нагревание происходит за счет поглощения энергии излучения полусферой (см. рис. 1), причем количество поглощённой энергии определяется косинусом угла между нормалью и направлением распространения излучения.

При моделировании использовать следующую математическую модель:

$$\begin{cases} c \frac{\partial u}{\partial t} = K \Delta u + \beta f(\vec{r}), & \{x \in \mathbb{R}^3 \mid \rho(x,0) = R\}, t \geq 0; \\ u(\theta, \rho, t)|_{t=0} = 0. \end{cases} \quad (0.1)$$

где  $u(\theta, \varphi, t)$  — функция, описывающая температуру в некоторой точке  $(R, \theta, \varphi)$  в момент времени  $t$ ;

$f$  — функция, описывающая энергию, поглощенную единицей площади;

$c$  — положительная константа;

$K$  — положительная константа.

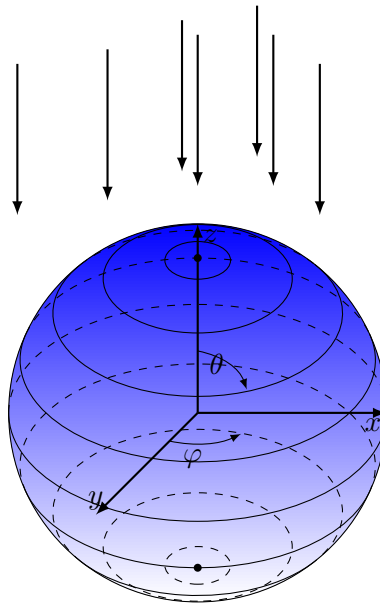


Рисунок 1 – К условиям задачи

В данной курсовой работе необходимо:

1. Используя метод разделения переменных, получить решение задачи математической физики в виде ряда Фурье.
2. Исследовать сходимость ряда. Получить оценку остатка ряда.

3. Разработать программу расчёта решения задачи с требуемой точностью. (Если необходимо, то использовать метод численного интегрирования для расчёта коэффициентов ряда. При этом следует контролировать погрешность численного интегрирования.)

4. Исследовать качество полученной аналитической оценки остатка ряда, используя вычислительный эксперимент.

## РЕФЕРАТ

Отчет 23 с, 5 рисунка, 58 уравнений, 7 использованных источников.

УРАВНЕНИЯ МАТЕМАТИЧЕСКОЙ ФИЗИКИ, УРАВНЕНИЕ ТЕПЛОПРОВОДНОСТИ, ПОЛИНОМЫ ЛЕЖАНДРА, РАЗЛОЖЕНИЕ В РЯД

Целью данной работы является решение задачи теплопроводности на шаре при заданных параметрах, более конкретно — разработка программы, выводящей график зависимости температуры от координат шара, времени и заданных констант.

Программа разрабатывалась в операционной системе *Ubuntu 15.10* с использованием сред *Scala IDE*, *Netbeans* и *IntelliJ Idea Community Edition*.

## СОДЕРЖАНИЕ

Введение . . . . .	6
1 Представление решения в виде ряда . . . . .	7
1.1 Упрощение модели . . . . .	7
1.2 Решение однородного уравнения . . . . .	7
1.3 Решение неоднородного уравнения . . . . .	9
1.4 Определение вида коэффициентов разложения $\{f_k\}_0^{+\infty}$ функции $f$ . . . . .	11
2 Исследование ряда . . . . .	14
2.1 Исследование сходимости ряда . . . . .	14
2.2 Определение остатка ряда . . . . .	15
3 Написание программы . . . . .	17
3.1 Описание использованных языков, приложений и пакетов . .	17
3.2 Описание кода . . . . .	17
3.3 Тестирование . . . . .	19
Заключение . . . . .	22

## ВВЕДЕНИЕ

В данной работе решается задача теплопроводности на сфере. Изменение температуры происходит за счет поглощения энергии верхней полусферой — аналогичная ситуация возникает, например, при нагреве Солнцем поверхности Земли.

В первом разделе исходная задача сводится к разложению искомой функции в ряд по ортогональным функциям — полиномам Лежандра. Это позволяет рассматривать счетное множество независимых уравнений. В конце раздела приводится явное разложение функции температуры, что позволяет найти решение в численном виде.

Во втором разделе проверяется сходимость ряда и оценивается его остаток в зависимости от количества слагаемых ряда. В результате, можно по заранее заданной точности определить число слагаемых, а значит — решить задачу с заданной точностью с использованием ЭВМ.

В третьем разделе приводится код программы, необходимый и достаточный для вычисления функции температуры. В нем же указывается ссылка на полный код проекта (а также на *jar*-архив), скомпилировав (запустив) который можно просматривать график решения. В конце раздела приводятся примеры графиков при заданных параметрах.

# 1 ПРЕДСТАВЛЕНИЕ РЕШЕНИЯ В ВИДЕ РЯДА

## 1.1 Упрощение модели

Оператор Лапласа в сферических координатах задается уравнением

$$\Delta u(r, \theta, \varphi, t) = \frac{1}{r^2} \frac{\partial}{\partial r} \left( r^2 \frac{\partial u}{\partial r} \right) + \frac{1}{r^2 \sin \theta} \frac{\partial}{\partial \theta} \left( \sin \theta \frac{\partial u}{\partial \theta} \right) + \frac{1}{r^2 \sin^2 \theta} \frac{\partial^2 u}{\partial \varphi^2}. \quad (1.1)$$

В нашем случае,  $r$  является константой ( $r \equiv R$ ). Также, можно выбрать базис (систему координат) так, чтобы функция  $u$  зависела лишь от одного пространственного параметра, а именно, от  $\theta$  (как это показано на рис. 1). Поэтому оператор Лапласа можно записать в следующем виде:

$$\Delta u(\theta, t) = \frac{1}{R^2 \sin \theta} \frac{\partial}{\partial \theta} \left( \sin \theta \frac{\partial u}{\partial \theta} \right). \quad (1.2)$$

Таким образом, из уравнений (0.1) и (1.2) получаем, что

$$c \frac{\partial u}{\partial t} = \frac{K}{R^2} \times \frac{1}{\sin \theta} \frac{\partial}{\partial \theta} \left( \sin \theta \frac{\partial u}{\partial \theta} \right) + \beta f(\vec{r}). \quad (1.3)$$

Введем теперь замену  $u(t, \theta) = X(\theta)T(t)$ :

$$cX(\theta) \frac{dT}{dt} = \frac{K}{R^2} T(t) \frac{1}{\sin \theta} \frac{d}{d\theta} \left( \sin \theta \frac{dX}{d\theta} \right) + \beta f(\vec{r}). \quad (1.4)$$

## 1.2 Решение однородного уравнения

Рассмотрим однородное уравнение (1.4):

$$cX(\theta) \frac{dT}{dt} = \frac{K}{R^2} T(t) \frac{1}{\sin \theta} \frac{d}{d\theta} \left( \sin \theta \frac{dX}{d\theta} \right), \quad (1.5)$$

или

$$c \frac{d \ln T}{dt} = \frac{K}{R^2} \times \frac{1}{X(\theta) \sin \theta} \frac{d}{d\theta} \left( \sin \theta \frac{dX}{d\theta} \right).$$

Согласно [1, стр. 50], т. к.  $\theta$  и  $t$  — независимые переменные, то

$$\frac{d \ln T}{dt} = \frac{K}{cR^2} \times \frac{1}{X(\theta) \sin \theta} \frac{d}{d\theta} \left( \sin \theta \frac{dX}{d\theta} \right) = -\mu,$$

где  $\mu$  — некоторая константа.

Следовательно, мы имеем два дифференциальных уравнения:

$$\frac{d \ln T}{dt} + \mu = 0, \quad (1.6)$$

$$\frac{d}{d\theta} \left( \sin \theta \frac{dX}{d\theta} \right) + \mu \frac{cR^2}{K} \sin \theta X = 0. \quad (1.7)$$

Введем замену  $x = \cos(\theta)$  и будем учитывать, что  $\theta \in (0; \pi)$ . Тогда

$$dx = -\sin(\theta) d\theta, \quad \frac{d}{d\theta} = -\sin(\theta) \frac{d}{dx} = -\sqrt{1-x^2} \frac{d}{dx},$$

и уравнение (1.7) преобразуется в

$$\frac{d}{dx} \left( -(1-x^2) \frac{dX}{dx} \right) - \mu \frac{cR^2}{K} X = 0,$$

или

$$(1-x^2) \frac{d^2 X}{dx^2} - 2x \frac{dX}{dx} + \mu \frac{cR^2}{K} X = 0. \quad (1.8)$$

Это есть уравнение гипергеометрического типа ([2, с. 12]). Введем новые функции:  $\sigma(x) = 1-x^2$ ,  $\tau(x) = -2x$ ,  $\lambda = \mu \frac{cR^2}{K}$ . Имеем:

$$\sigma(x) \frac{d^2 X}{dx^2} + \tau(x) \frac{dX}{dx} + \lambda X = 0. \quad (1.9)$$

Для этого уравнения строится класс наиболее простых решений — классические ортогональные полиномы, определяемые формулой Родрига [2, § 9 п. 2]:

$$y_n(x) = \frac{Y_n}{\rho(x)} \frac{d^n}{dx^n} [\sigma^n \rho(x)], \quad \forall n \in \mathbb{N}_0, \quad (1.10)$$

где  $\rho(x)$  — функция, удовлетворяющая уравнению  $(\sigma\rho)' = \tau\rho$ ;

$\{Y_n\}$  — некоторые константы.

При этом коэффициенты  $\lambda$  уравнения (1.9), связаны соотношением

$$\lambda + n\tau' + \frac{1}{2}n(n-1)\sigma'' = 0.$$

В нашем случае, для нахождения множества коэффициентов получаем уравнения

$$\lambda_n = n(1+n), \quad \forall n \in \mathbb{N}_0. \quad (1.11)$$

Согласно [3, часть II, § 1 п. 3], полиномы Лежандра  $\{P_n(x)\}$  являются собственными функциями оператора  $\frac{d}{dx} \left( (1-x^2) \frac{d}{dx} \right)$ , соответствующие (биективно по эквивалентному номеру) полученным собственным значени-



ям  $\{\lambda_n\}$ .

Используя полученные собственные функции  $\{P_n(x)\}$ , полная система которых одновременно является базисом для разложения других функций, мы можем записать функцию  $X$  через ряд:

$$X(x) = \sum_{k=0}^{+\infty} B_k P_k(x). \quad (1.12)$$

Возвращаясь к переменной  $\theta$ , получаем:

$$X(\theta) = \sum_{k=0}^{+\infty} B_k P_k(\cos \theta). \quad (1.13)$$

В то же время, по уравнению (1.6), с учетом полученных в системе (1.11) собственных чисел, можно определить функции

$$T_n(t) = e^{-\mu_n t} = e^{-n(1+n)\frac{K}{cR^2}t}, \quad \forall n \in \mathbb{N}_0, \quad (1.14)$$

причем каждый  $T_n$ ,  $n \in \mathbb{N}_0$ , будет соответствовать  $X_n$ . Решениями уравнения (1.6), вообще говоря, являются функции  $\{C_n T_n(t)\}$ , где  $\{C_n\}$  — некоторое множество констант.

Таким образом, мы можем разложить функцию  $u(t, \theta) = X(\theta)T(t)$  в ряд:

$$u(t, \theta) = \sum_{k=0}^{+\infty} C_k T_k(t) B_k P_k(\cos \theta) = \sum_{k=0}^{+\infty} A_k T_k(t) P_k(\cos \theta). \quad (1.15)$$

### 1.3 Решение неоднородного уравнения

Пусть функция  $f$  уравнения (0.1) разлагается в ряд по полученным собственным функциям:

$$f(\theta) = \sum_{k=0}^{+\infty} f_k P_k(\cos \theta),$$

при этом коэффициенты  $\{f_k\}$  находятся из условия ортогональности полиномов Лежандра:

$$\begin{aligned} \int_{-1}^1 f(\theta) P_n(\cos \theta) d \cos \theta &= \sum_{k=0}^{+\infty} \int_{-1}^1 f_k P_k(\cos \theta) P_n(\cos \theta) d \cos \theta = \\ &= \|P_n\|^2 f_n, \quad \forall n \in \mathbb{N}_0. \end{aligned} \quad (1.16)$$

Таким образом,

$$f_n = \frac{1}{\|P_n\|^2} \int_{-1}^1 f(\theta) P_n(\cos \theta) d \cos \theta, \quad \forall n \in \mathbb{N}_0. \quad (1.17)$$

Будем находить функцию  $u$ , используя метод вариации произвольных постоянных  $\{A_k\}$ :  $\forall k \in \mathbb{N}_0$   $A_k = A_k(t)$ . В соответствии с уже полученным разложением (1.15), уравнение (1.3) можно переписать в виде

$$\begin{aligned} c \sum_{k=0}^{+\infty} P_k(\cos \theta) \frac{dA_k T_k}{dt} = \\ = \frac{K}{R^2} \times \frac{1}{\sin \theta} \frac{d}{d\theta} \left( \sin \theta \sum_{k=0}^{+\infty} A_k T_k \frac{dP_k(\cos \theta)}{d\theta} \right) + \beta \sum_{k=0}^{+\infty} f_k P_k(\cos \theta). \end{aligned} \quad (1.18)$$

Очевидно, что если

$$\begin{aligned} c P_k(\cos \theta) \frac{dA_k T_k}{dt} = A_k T_k \frac{K}{R^2} \times \frac{1}{\sin \theta} \frac{d}{d\theta} \left( \sin \theta \frac{dP_k(\cos \theta)}{d\theta} \right) + \\ + \beta f_k P_k(\cos \theta), \quad \forall k \in \mathbb{N}_0, \end{aligned} \quad (1.19)$$

то равенство в уравнении (1.18) будет выполняться.

Т.к.  $\{P_k\}$  — собственные функции оператора  $\frac{d}{dx} \left( (1-x^2) \frac{d}{dx} \right)$ , то

$$\frac{1}{\sin \theta} \frac{d}{d\theta} \left( \sin \theta \frac{dP_k(\cos \theta)}{d\theta} \right) = -\lambda_k P_k(\cos \theta),$$

и из системы (1.19) получаем

$$c P_k(\cos \theta) \frac{dA_k T_k}{dt} = -\lambda_k P_k(\cos \theta) A_k T_k \frac{K}{R^2} + \beta f_k P_k(\cos \theta), \quad \forall k \in \mathbb{N}_0,$$

или

$$\frac{dA_k T_k}{dt} = -k(k+1) \frac{K}{c R^2} A_k T_k + \beta \frac{f_k}{c}, \quad \forall k \in \mathbb{N}_0. \quad (1.20)$$

Решие уравнения  $y' = -ay + b$  при  $a \neq 0$  есть  $y = \frac{b}{a} + C e^{-at}$ , где  $C$  — некоторая константа. Поэтому решения системы (1.20) есть

$$A_k(t) T_k = \begin{cases} \frac{\beta \frac{f_k}{c}}{k(k+1) \frac{K}{c R^2}} + D_k T_k = \frac{\beta f_k}{k(k+1)} \times \frac{R^2}{K} + D_k T_k, & \forall k \in \mathbb{N}, \\ f_k \frac{\beta}{c} t + D_k, & k = 0, \end{cases} \quad (1.21)$$

где  $\{D_k\}$  — некоторое множество констант.

Тогда из уравнения (1.15) следует, что  $u(\theta, t)$  есть

$$u(\theta, t) = \left( f_k \frac{\beta}{c} t + D_0 \right) P_0(\cos \theta) + \sum_{k=1}^{+\infty} \left( \frac{R^2}{K} \times \frac{\beta f_k}{k(k+1)} + D_k T_k(t) \right) P_k(\cos \theta). \quad (1.22)$$

Применим однородное условие (0.1):

$$D_k = \begin{cases} -\frac{R^2}{K} \times \frac{\beta f_k}{k(k+1)}, & \forall k \in \mathbb{N}, \\ 0, & k = 0. \end{cases} \quad (1.23)$$

С учетом этого условия, окончательно получаем, что

$$u(\theta, t) = \beta \frac{R^2}{K} \sum_{k=1}^{+\infty} \frac{f_k}{k(k+1)} \left( 1 - e^{-k(1+k) \frac{K}{cR^2} t} \right) P_k(\cos \theta) + f_0 \frac{\beta}{c} P_0(\cos \theta) t. \quad (1.24)$$

#### 1.4 Определение вида коэффициентов разложения $\{f_k\}_0^{+\infty}$ функции $f$

Согласно формулировки задания, функция  $f(\theta)$  будет определяться из уравнения

$$f(\theta) = \eta(\cos(\theta)) \cos \theta, \quad (1.25)$$

где  $\eta(x)$  — функция Хевисайда.

Определим теперь по формуле (1.17) коэффициенты  $\{f_k\}$ :

$$\begin{aligned} f_k &= \frac{1}{\|P_k\|^2} \int_0^1 \cos \theta P_k(\cos \theta) d \cos \theta \\ &= \frac{1}{\|P_k\|^2} \int_0^1 x P_k(x) dx, \quad \forall k \in \mathbb{N}_0. \end{aligned} \quad (1.26)$$

Норма  $\|P_k\|$ , для всех  $k \in \mathbb{N}_0$ , согласно [3, часть II, § 1 п. 5] определяется из уравнения

$$\|P_k\| = \sqrt{\frac{2}{2n+1}}. \quad (1.27)$$

Согласно [4, ур-е 26], для всех целых  $n > 1$

$$\int_0^1 x P_n(x) dx =$$

$$= \frac{n(2n+3)P_{n-2}(0) - (2n+1)P_n(0) - (n+1)(2n-1)P_{n+2}(0)}{(2n-1)(2n+1)(2n+3)}. \quad (1.28)$$

Для  $n = 0$  имеем:

$$\int_0^1 xP_n(x) dx = \int_0^1 x dx = \frac{1}{2}, \quad (1.29)$$

а для  $n = 1$ :

$$\int_0^1 xP_n(x) dx = \int_0^1 x^2 dx = \frac{1}{3}. \quad (1.30)$$

Согласно [4, ур-е 18], для нечетных  $n$   $P_n(0) = 0$  и для целых  $n$   $P_{2n}(0)$  можно найти из уравнения

$$P_{2n}(0) = \frac{(-1)^n(2n)!}{2^{2n}(n!)^2}. \quad (1.31)$$

Будем определять последующие коэффициенты рекурсивно:

$$P_{2n+2}(0) = -P_{2n}(0) \frac{(2n+1)(2n+2)}{4(n+1)^2} = -P_{2n}(0) \frac{2n+1}{2(n+1)}, \quad (1.32)$$

начиная с  $P_0(0) = 1$ . Соответственно, для  $P_{2n-2}(0)$  имеем:

$$P_{2n-2}(0) = -P_{2n}(0) \frac{4n^2}{2n(2n-1)} = -P_{2n}(0) \frac{2n}{2n-1}. \quad (1.33)$$

Подставим полученные значения в формулу (1.28):

$$\begin{aligned} \int_0^1 xP_{2n}(x) dx &= \\ &= \frac{2n(4n+3)P_{2n-2}(0) - (4n+1)P_{2n}(0) - (2n+1)(4n-1)P_{2n+2}(0)}{(4n-1)(4n+1)(4n+3)} = \\ &= -P_{2n}(0) \frac{2n(4n+3)\frac{2n}{2n-1} + (4n+1) - (2n+1)(4n-1)\frac{2n+1}{2(n+1)}}{(4n-1)(4n+1)(4n+3)} = \\ &= -P_{2n}(0) \frac{(4n+3)(4n-1)(4n+1)}{2(4n-1)(4n+1)(4n+3)(2n-1)(n+1)} = \\ &= -P_{2n}(0) \frac{1}{2(2n-1)(n+1)}. \end{aligned} \quad (1.34)$$

Найдем теперь коэффициенты разложения функции  $f$ , используя (1.26), (1.27) и (1.34).

Из того, что для целых  $k \in \mathbb{N}$   $P_{2k+1}(0) = 0$ , следует, что  $f_{2k+1} = 0$  для

всех  $k \in \mathbb{N}$ . Для  $n = 0$  имеем:

$$f_0 = \frac{1}{2} \int_0^1 x \, dx = \frac{1}{4}, \quad (1.35)$$

для  $n = 1$ :

$$f_1 = \frac{3}{2} \int_0^1 x^2 \, dx = \frac{1}{2}, \quad (1.36)$$

для остальных  $n > 0$ :

$$\begin{aligned} f_{2n} &= -P_{2n}(0) \frac{2 \cdot 2n + 1}{2} \times \frac{1}{2(2n - 1)(n + 1)} = \\ &= -P_{2n}(0) \frac{4n + 1}{4(n + 1)(2n - 1)}. \end{aligned} \quad (1.37)$$

Напомним, что (для  $n \in \mathbb{N}_0$ )

$$P_{2n+2}(0) = -P_{2n}(0) \frac{2n + 1}{2(n + 1)}, \quad (1.38)$$

или (для  $n \in \mathbb{N}$ )

$$P_{2n}(0) = -P_{2n-2}(0) \frac{2n - 1}{2n}, \quad (1.39)$$

причем  $P_0(0) = 1$ .

Подставляя (1.39) в формулу (1.37), для  $n \in \mathbb{N}$  получаем:

$$\begin{aligned} f_{2n} &= P_{2n-2}(0) \frac{2n - 1}{2n} \times \frac{4n + 1}{4(n + 1)(2n - 1)} = \\ &= P_{2n-2}(0) \frac{4n + 1}{8n(n + 1)}. \end{aligned} \quad (1.40)$$

Подставим, наконец, полученные значения в ряд (1.24):

$$\begin{aligned} u(\theta, t) &= \beta \frac{R^2}{K} \sum_{k=1}^{+\infty} \frac{f_k}{k(k+1)} \left( 1 - e^{-k(1+k)\frac{K}{cR^2}t} \right) P_k(\cos \theta) + \\ &\quad + f_0 \frac{\beta}{c} P_0(\cos \theta) t = \\ &= \beta \frac{R^2}{K} \sum_{k=1}^{+\infty} \frac{(4k+1)P_{2k-2}(0)}{16k^2(2k+1)(k+1)} \left( 1 - e^{-2k(1+2k)\frac{K}{cR^2}t} \right) P_{2k}(\cos \theta) + \\ &\quad + \frac{\beta}{4c} t + \beta \frac{R^2}{4K} \left( 1 - e^{-2\frac{K}{cR^2}t} \right) \cos \theta. \end{aligned} \quad (1.41)$$

## 2 ИССЛЕДОВАНИЕ РЯДА

### 2.1 Исследование сходимости ряда

Пусть задана функция

$$u(x) = \sum_{k=0}^{+\infty} \hat{u}_k P_k(x), \quad x \in [-1; 1], \quad (2.1)$$

определено пространство

$$L_2[-1, 1] = \{u \mid \|u\| < \infty\},$$

где  $\|u\| = \sqrt{(u, u)}$ ,  $(u, v) = \int_{-1}^1 u(x)v(x) dx$ .

Пусть также определены частичные суммы

$$u_N(x) = \sum_{k=0}^N \hat{u}_k P_k(x), \quad x \in [-1; 1], \quad N \in \mathbb{N}. \quad (2.2)$$

Тогда, согласно [5] (восходящей к [6]), если  $u \in H^{2q}[-1, 1]$ , то для  $q \geq 0$

$$\sqrt{\sum_{k=N+1}^{+\infty} \|P_k\|^2 \hat{u}_k^2} = \|u - u_N\| \leq CN^{-2q} \|u\|_{H^{2q}[-1, 1]}, \quad (2.3)$$

где  $H^p[-1, 1]$  — пространство Соболева:

$$H^p[-1, 1] = \left\{ u \in L^2[-1, 1] \mid \|u\|_{H^p[-1, 1]}^2 = \sum_{k=0}^p \|u^{(k)}\|^2 < \infty \right\};$$

$C$  — некоторая константа, согласно [6, Th. 2.1] равная  $2^{-d/2+1}$  ( $d$  — размерность  $x$ , в [5]  $d = 1$ ).

Исходя из этого, если сама  $u$  имеет конечную норму, и если нормы ее производных по  $x$  вплоть до  $p$ -го порядка ( $p > 0$ ) — конечны, то ряд сходится в  $L_2$ .

В нашем случае, при ограниченном максимуме

$$\max_{n \in \mathbb{N}} \left| \int_{-1}^1 f(\theta) P_n(\cos \theta) d \cos \theta \right| < \infty \quad (2.4)$$

норма  $\|u\|$  тоже ограничена, т.к. сходится ряд  $\sum_{k=1}^{+\infty} k^{-2}$ . Можно показать, что сумма норм ее производных также конечна. Однако, основная задача — проверка сходимости в  $L_1$ . Оценим остаток ряда, используя явное выражение функции  $f$ .

## 2.2 Определение остатка ряда

Вспомним полученное выражение для функции  $u(t, \theta)$ :

$$u(\theta, t) = \beta \frac{R^2}{K} \sum_{k=1}^{+\infty} \frac{(4k+1)P_{2k-2}(0)}{16k^2(2k+1)(k+1)} \left(1 - e^{-2k(1+2k)\frac{K}{cR^2}t}\right) P_{2k}(\cos \theta) + \frac{\beta}{4c}t + \beta \frac{R^2}{4K} \left(1 - e^{-2\frac{K}{cR^2}t}\right) \cos \theta. \quad (2.5)$$

Пусть для всех  $n \in \mathbb{N}$

$$I_n = \sum_{k=n}^{+\infty} \frac{4k+1}{16k^2(2k+1)(k+1)} \quad (2.6)$$

и

$$G_n = \beta \frac{R^2}{K} \sum_{k=1}^{+\infty} \frac{(4k+1)P_{2k-2}(0)}{16k^2(2k+1)(k+1)} \left(1 - e^{-2k(1+2k)\frac{K}{cR^2}t}\right) P_{2k}(\cos \theta). \quad (2.7)$$

Заметим, что  $|G_n|$  является оценкой  $n$ -го остатка ряда (2.5), точнее — погрешность метода в вычислениях функции  $u$  при использовании  $n$  членов ряда (2.5) не превосходит по модулю  $|G_n|$ .

Функции  $\{P_n(\cos \theta)\}$  являются ограниченными на множестве  $[-1; 1]$  (см. [3, часть II, § 1 п. 7]):

$$|P_n(x)| \leq 1, \quad \forall n \in \mathbb{N}_0. \quad (2.8)$$

Учитывая это, а также то, что для всех  $k \in \mathbb{N}$   $e^{-2k(1+2k)\frac{K}{cR^2}t} \leq 1$ , получаем оценки для  $\{G_n\}$ :

$$G_n \leq \beta \frac{R^2}{K} I_n \cdot 1 \cdot 1 = \beta \frac{R^2}{K} I_n, \quad \forall n \in \mathbb{N}. \quad (2.9)$$

Оценим сумму  $I_n$ :

$$\begin{aligned} I_n &< \sum_{k=n}^{+\infty} \frac{4k+2}{16k^2(2k+1)(k+1)} \\ &= \sum_{k=n}^{+\infty} \frac{1}{8k^2(k+1)} \\ &< \sum_{k=n}^{+\infty} \frac{1}{8k^3}. \end{aligned} \quad (2.10)$$

Рассмотрим функцию  $h(\xi) = 1/8\xi^{-3}$  и последовательность  $w_n = 1/8n^{-3}$ . Очевидно, что  $h(\xi)$  определена на интервале  $[1; +\infty)$ , и, кроме того, убыв-

вает на нем. Заметим также, что

$$w_{k+1} \leq h(\xi) \leq w_k, \quad k \in \mathbb{N}, k \leq x \leq k+1. \quad (2.11)$$

Поэтому справедливы следующие неравенства (см. [7, стр. 347]):

$$w_{k+1} \leq \int_k^{k+1} h(\xi) d\xi \leq w_k, \quad k \in \mathbb{N}, k \leq x \leq k+1. \quad (2.12)$$

Суммируя по  $k$  от  $n_0 \in \mathbb{N}$  до бесконечности, получаем:

$$\sum_{k=n_0}^{+\infty} w_{k+1} \leq \sum_{k=n_0}^{+\infty} \int_k^{k+1} h(\xi) d\xi = \int_{n_0}^{+\infty} h(\xi) d\xi. \quad (2.13)$$

Вычислим теперь интеграл  $\int_{n_0}^{+\infty} h(\xi) d\xi$ :

$$\int_{n_0}^{+\infty} h(\xi) d\xi = \frac{1}{8} \int_{n_0}^{+\infty} \xi^{-3} d\xi = \frac{1}{8} \left( 0 - \left( -\frac{1}{2} n_0^{-2} \right) \right) = \frac{1}{16} n_0^{-2}. \quad (2.14)$$

Таким образом, из формул (2.10), (2.13) и (2.14) мы можем оценить  $I_{n_0}$ :

$$I_{n_0} < \frac{1}{16} n_0^{-2}, \quad n_0 \in \mathbb{N}. \quad (2.15)$$

Наконец, из системы (2.9) получаем, что

$$G_{n_0} < \frac{\beta R^2}{16K} n_0^{-2}, \quad n_0 \in \mathbb{N}. \quad (2.16)$$

Учитывая пояснения к формуле (2.7), получаем, что для достижения точности  $\varepsilon$  вычисления функции  $u$  достаточно использовать

$$n_0 = \left\lceil \sqrt{\frac{\beta R^2}{16K\varepsilon}} \right\rceil$$

членов ряда (2.5).



## 3 НАПИСАНИЕ ПРОГРАММЫ

### 3.1 Описание использованных языков, приложений и пакетов

Для написания программы были использованы языки *Java* и *Scala*. Код на языке *Scala* 2.11 был написан в среде *Scala IDE build of Eclipse SDK*, версия IDE — 4.1.0. Графический интерфейс был написан в среде *NetBeans 8.02* на языке *Java* 8. Итоговая версия была собрана с использованием *IntelliJ IDEA 15.0*.

При написании интерфейса были использованы пакеты *jfreechart-1.0.19*, *jcommon-1.0.23* и *jfreeSVG-2.0*.

Программа может быть запущена на компьютерах с установленным *JDK 8*.

### 3.2 Описание кода

**3.2.1** Объект для описания исходных данных представлен на следующем листинге.

```
1 package solution
2 object Data{
3     var c: Double = 2.0
4     var K: Double = 0.01
5     var R: Double = 2.0
6     var beta: Double = 0.1
7     // @param eps: accuracy:  $|u - u^*|, u^* = \sum_0^N \{\dots\}$ 
8     var eps: Double = 1e-7
9
10    def R2 = R*R
11    def kDelcR2 = K/(c*R2)
12    def bR2DelK = beta*R2/K
13
14    // Long.Max =  $2^{63}$ . I can use  $n0^4$ ,  $\Rightarrow n0 < (2^{63})^{0.25} =$ 
15    // 55108.
16    val maxN0 = 40000
17    /** counts of  $\{f_{2k}\}, \{P_{2k}(0)\}$  */
18    /**  $n_0 = \left\lceil \sqrt{\frac{\beta R^2}{16K\varepsilon}} \right\rceil$  */
19    def n0 = math.min(math.sqrt(bR2DelK/(16*eps)),
20        (maxN0>>1)).toInt
21
22    /**  $P_{2n}(0) = -P_{2n-2}(0)\frac{2n-1}{2n}, P_0(0) = 1$  */
```

```

21 val P2k = new Array[Double](n0)
22 def setAllP2k{
23     P2k(0)=1
24     for (i <- 1 until n0)
25         P2k(i) = -P2k(i-1)*((i << 1)-1)/(i << 1)
26 }
27
28 // time
29 var t: Double = 1e-3
30 // initial of {P2k}
31 def init = setAllP2k
32 }

```

Процедура *init* выполняет инициализацию множества полиномов Лежандра в точке 0, и вызывается лишь один раз за все время работы программы.

**3.2.2** Основные вычисления (в частности, вычисление искомой функции *u*) реализованы в объекте *Solution*. Сложные функции пояснены комментариями.

```

1 package solution
2 object Solution {
3     import math.exp
4     import solution.Data
5
6     val n0 = solution.Data.n0
7
8     /** e-k(k+1) $\frac{K}{cR^2}t$  */
9     def expK(k: Int, t: Double) =
10         exp(-(k*(k+1))*Data.kDelcR2*t);
11
12     /** Pn(x) =  $\frac{2n-1}{n}xP_{n-1}(x) - \frac{n-1}{n}P_{n-2}(x)$ ,  $\forall n > 1$  */
13     def getP_k(n: Int, pk1: Double, pk2: Double, x: Double)
14         = ((n+n-1)*x*pk1 - (n-1)*pk2)/n;
15
16     /** return result function u(t,θ)
17         * @param t --- time
18         * @param theta --- angle, radians */
19     def u(t: Double, theta: Double) = {
20         val x = math.cos(theta)
21         var pk2 = 1.0; // Pk-2
22         var pk1 = x; // Pk-1
23         var u = 0.0; // u(t,θ) --- result
24         var pk = 1.0; // P0

```

```

23   for (k <- 2 to n0<<1) {
24       pk = getP_k(k,pkm1,pkm2,x) //  $P_i$ 
25       pkm2 = pkm1
26       pkm1 = pk
27       //  $\frac{(4k+1)P_{2k-2}(0)}{16k^2(2k+1)(k+1)} \left(1 - e^{-2k(1+2k)\frac{K}{cR^2}t}\right) P_{2k}(\cos \theta)$ 
28       if ((k & 1) == 0)
29           u += ((k<<1)+1)*Data.P2k((k>>1)-1)*
30               (1.0 - expK(k,t))*pk/
31               (k.toLong*k*(k+1)*((k>>1)+1) << 2) //  $2k- > k$ 
32   }
33   u += 0.25*(1.0-expK(1,t))*x //  $1/4(1 - e^{-2\frac{K}{cR^2}t}) \cos(\theta)$ 
34   u *= Data.bR2DelK
35   u += 0.25*Data.beta*t/Data.c //  $\frac{\beta}{4c}t$ 
36   u
37 }
38 }

```

Класс, отвечающий за интерфейс программы, является громоздким (в нем 352 строки), к тому же, он не является обязательным для выполнения основной задачи — вычисления функции  $u$ .

Полную версию программы можно посмотреть по адресу <https://github.com/Mikhail42/EMPh>.

### 3.3 Тестирование

Тестирование происходило в операционной системе *Ubuntu 15.10*. Примеры результатов выполнения программы представлены ниже.

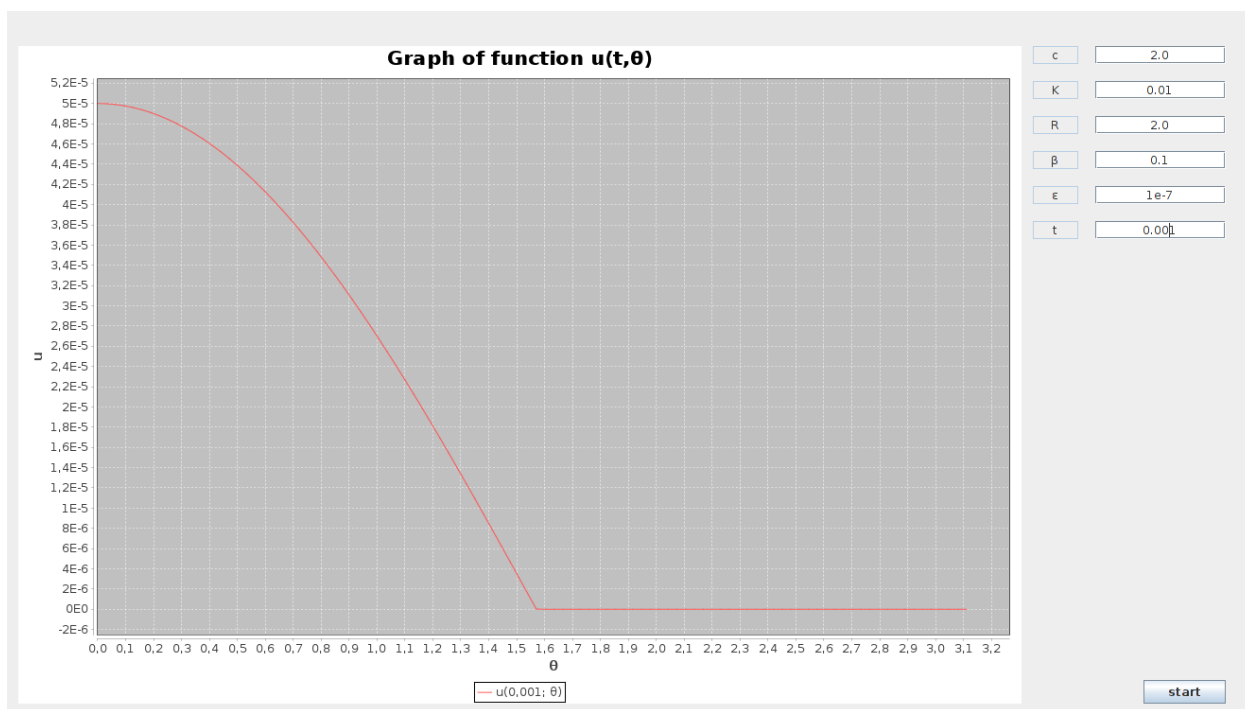


Рисунок 1 – График с исходными значениями при  $t = 0.001$

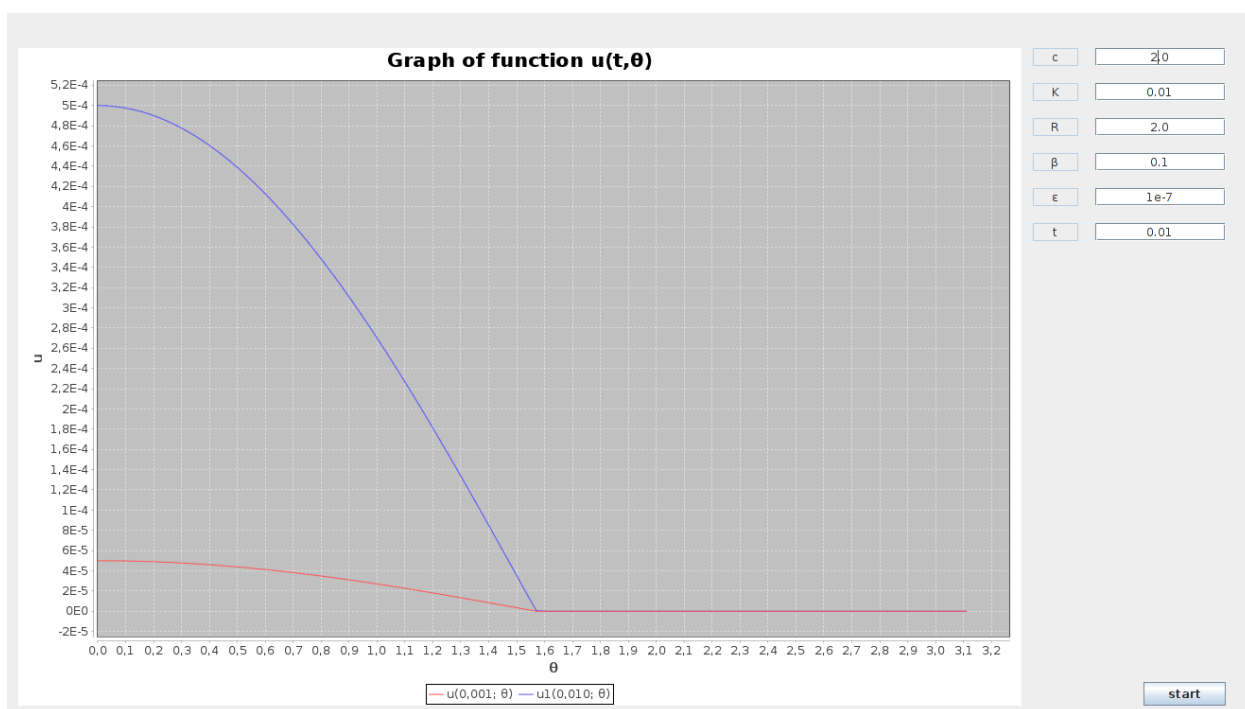


Рисунок 2 – График после изменения  $t$  на 0.01

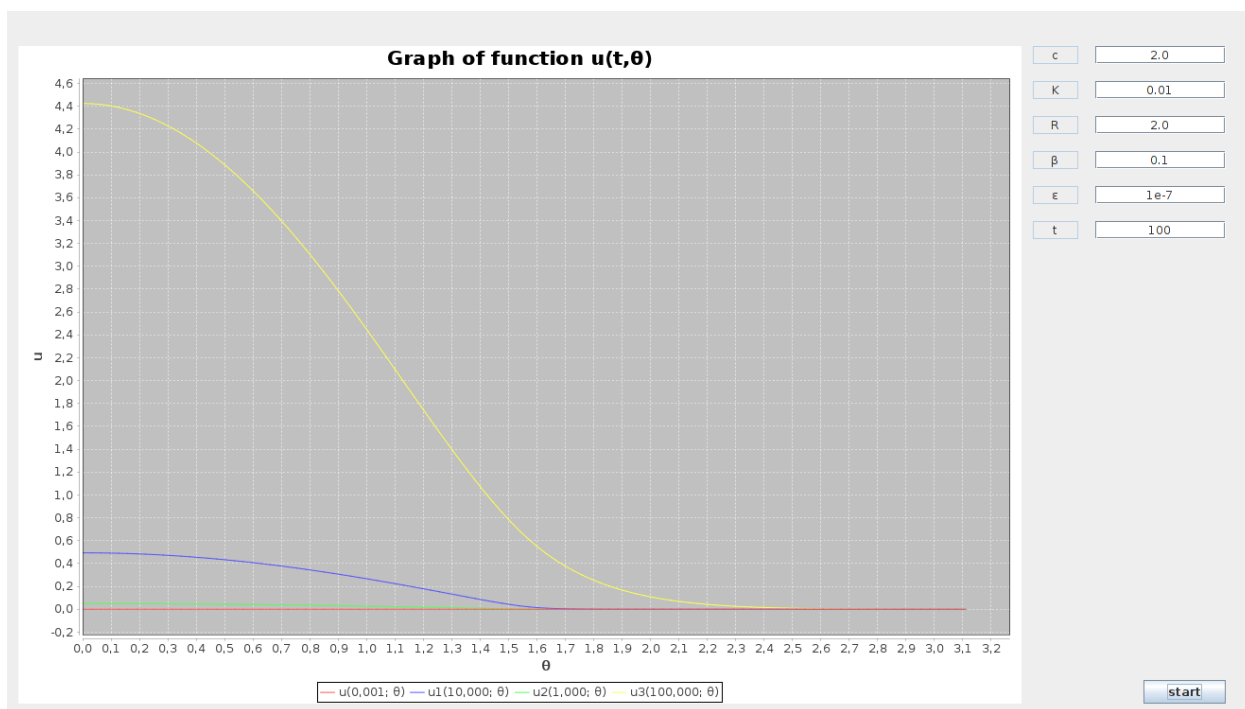


Рисунок 3 – Графики при различных  $t \geq 0.001$

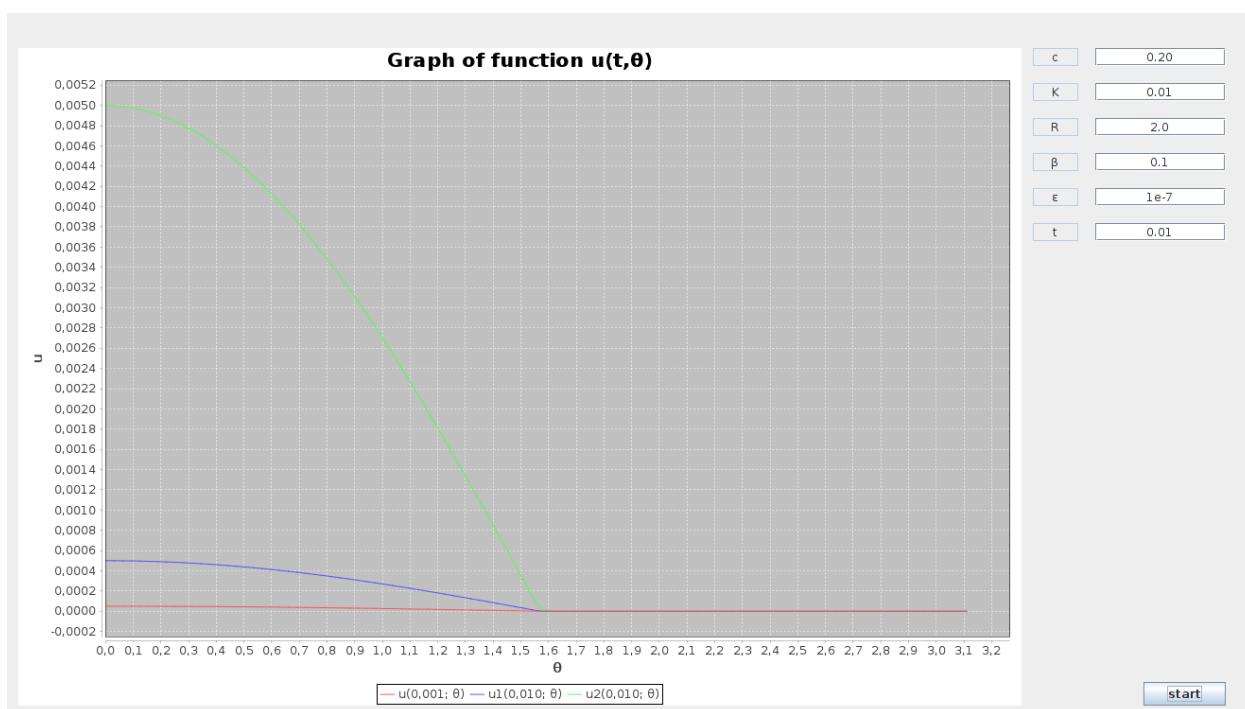


Рисунок 4 – График после изменения  $c$  на 0.2

## ЗАКЛЮЧЕНИЕ

По результатам данной курсовой работы были выполнены следующие задачи:

- получено решение задачи математической физики в виде ряда по функциям Лежандра;
- исследована сходимость ряда в  $L_1$ , получена оценка остатка ряда;
- разработана программа расчёта решения задачи с требуемой точностью;
- с использованием вычислительного эксперимента, исследовано качество полученной аналитической оценки остатка ряда.

Таким образом, достигнута цель курсовой работы.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 А.В. Кузнецов. Методы математической физики. — Ярославль : Яросл. гос. ун-т, 2004. — 200 с.
- 2 А.Ф. Никифоров, В.Б. Уваров. Специальные функции математической физики. — 2-е издание. — М. : Наука, 1984. — 346 с.
- 3 А.Н. Тихонов, А.А. Самарский. Уравнения математической физики: Учеб. пособие. — 6-е издание. — М. : Изд-во МГУ, 1999. — 799 с.
- 4 A. Clark, Jr. Legendre Polynomials // Applications and Examples. — University of Rochester, 2010. — P. 21. — URL: <http://www.me.rochester.edu/courses/ME201/webexamp/legendre.pdf>.
- 5 Hesthaven Jan S., Kirby Robert M. Filtering in Legendre spectral methods // Mathematics of computation. — 2008. — July. — Vol. 77, no. 263. — P. 1425–1452. — URL: <http://www.ams.org/journals/mcom/2008-77-263/S0025-5718-08-02110-8/S0025-5718-08-02110-8.pdf>.
- 6 Canuto C., Quarteroni A. Approximation Result for Orthogonal Polynomials in Sobolev Spaces // Mathematics of computation. — 1982. — January. — Vol. 38, no. 157. — P. 67–86. — URL: <http://www.ams.org/journals/mcom/1982-38-157/S0025-5718-1982-0637287-3/S0025-5718-1982-0637287-3.pdf>.
- 7 Г.И. Архипов, В.А. Садовничий, В.Н. Чубариков. Лекции по математическому анализу. Классический университетский учебник. — 5-е издание. — М. : Дрофа, 2004. — 639 с.