БГУИР

Кафедра информатики

ОТЧЕТ

по лабораторной работе № 3

Метод сеток решения одномерного

нестационарного уравнения теплопроводности

Выполнил: Проверил:

ст. гр. 553504 Гербик А. И.

Криницин А. В.

Минск 2017

ОГЛАВЛЕНИЕ

[1. Введение 3](#_Toc465628474)

[1.1. Постановка задачи 3](#_Toc465628475)

[1.2. Краткие теоретические сведения 3](#_Toc465628476)

2.  [Исходный код программы………….](#_Toc465628485) 6

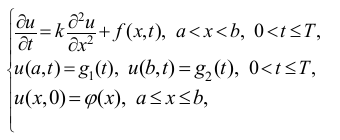
3[. Анализ полученных результатов](#_Toc465628485) 7

# 1.Введение

## Постановка задачи

Найти приближенное решение начально-краевой задачи для уравнения

теплопроводности:



используя явную разностную схему.

Взять h = ( b − a ) /10 ; шаг τ выбрать из условия устойчивости. Изобразить графики зависимости приближенного решения от x при t= 0 , 2 τ , 4 τ ,...T.

УКАЗАНИЕ. Условие устойчивости для явной разностной схемы имеет

вид τ ≤ 0.5( h 2 / k ) .

Вариант 13

a = 13

b = 0

k = 0.2

T = 0.25 f(x, t) = 1- x

## Краткие теоретические сведения

Требуется найти непрерывную на замкнутом пря­моугольнике *u*(*x, t*), которая на удовлетворяет уравнению теплопроводности

|  |  |  |
| --- | --- | --- |
|  | , | (1) |

при *t* = 0 удовлетворяет начальному условию

|  |  |  |
| --- | --- | --- |
|  | *u*(*x*, 0) *= s*(*x*), | (2) |

а при *х* = 0 и *х* = 1 подчиняется краевым условиям

|  |  |  |
| --- | --- | --- |
|  | *u*(0, *t*) *= p*(*t*), *u*(1, *t*) *= q*(*t*), | (3) |

где *f*(*x*, *t*), *s*(*x*), *p*(*t*), *q*(*t*) — заданные достаточно гладкие функции, причем *s*(0) = *p*(0), *s*(l) = *q*(0).

Задача (1) — (3) называется *смешанной*, поскольку она содержит как *начальное* *условие*, так и *краевые* *условия*. Известно, что у поставленной задачи сущестует единственное решение *u*(*х*, *t*). Мы будем предполагать, что это решение имеет на замкнутом прямо­угольнике непрерывные частные производные , , , .

Сетки и нормы. Пусть *h* = 1/*N*, = *T*/*M* — шаги по *x* и *t*, где *N, M* — натуральные, = *kh*, =, =. Построим сетки (рис. 1)

={: *k* = 0, 1, . . . , *N*, = 0, 1, . . . , *M*},

={: *k* = 1, 2, . . . , *N*-1, = 1, 2, . . . , *M*},

=\.

Сетка состоит из узлов сетки , обозначенных на рис. 1 крестиками. Эти узлы расположены на трех сторонах прямоугольника , на которых заданы начальное и краевые условия. Сетка состоит из осталь­ных узлов сетки . Зададим для сеточных функции определенных на или на , следующие нормы:

|  |  |  |
| --- | --- | --- |
|  | , . | (4) |

Разностные схемы. Введем разностный опе­ратор:

|  |  |  |
| --- | --- | --- |
|  | =. | (5) |

Здесь под выражением подразумевается значение сеточной функции *у* в точке (, ), т. е. .

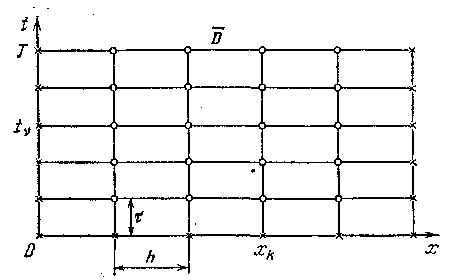


Рис. 1.

Скобки опущены для упрощения записи. Аналогичные упрощения в записи будем допускать и при введении других операторов.

Зададим на сетке тождественный оператор

|  |  |  |
| --- | --- | --- |
|  |  | (6) |

и сеточную функцию

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | *x* =, | *t* = 0, | *k* = 1, | 2, . . . , |  |
|  | *x* = 0, | *t* = , | = 0, | 1, . . . , *M*, | (7) |
|  | *x* = 1, | *t* = , | = 0, | 1, . . . , *M*. |  |

Рассмотрим две разностные схемы:

|  |  |  |
| --- | --- | --- |
|  | =, | (8) |

|  |  |  |
| --- | --- | --- |
|  | , | (9) |

|  |  |  |
| --- | --- | --- |
|  | =, | (10) |

|  |  |  |
| --- | --- | --- |
|  | . | (11) |

Здесь и далее индекс *k* изменяется от 1 до *N-*1, = l, 2, . . . , *М*.

Шаблоны разностных уравнений (8) и (10) представ­лены соответственно на рис. 2 и 3. Обе разностные

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | ◦ |  |  |  | ◦ | ◦ | ◦ | |  | |
|  |  | ◦ | ◦ | ◦ |  |  |  | ◦ |  | |  | |
|  |  |  |  |  |  |  |  |  |  | |  | |
|  |  |  |  |  |  |  |  |  |  | |  | |
|  |  | Рис. 2. | |  |  |  | Рис. 3. | | |  | |  |

схемы (8), (9) и (10), (11) называются *двухслойными*, так как шаблоны разностных уравнений (8) и (10) со­держат узлы, лежащие только на двух *слоях* — подмножествах сетки , отвечающих значениям времени и .

# Исходный код программы

|  |
| --- |
| # coding: utf-8  # In[2]:  import numpy as np  import sympy as sp  import pandas as pd  # In[3]:  get\_ipython().magic(u'pylab inline')  # In[4]:  x = sp.Symbol('x')  t = sp.Symbol('t')  # In[5]:  T = 0.25  a = 0  b = 1  k = 0.2  s = sp.sin(x \*1.0)  #s(x) | u(x, 0) = s(x)  g1 = 0 \* t  g2 = sp.sin(1.0 + 2 \* t)  f = 1 - x  h = (b - a) / 10.0  tetta = (h \* h \* k) / 2  # In[6]:  columns = np.arange(a, b + h, h)  index = np.arange(0, T + tetta, tetta)  values = pd.DataFrame(columns=range(len(columns)), index=range(len(index)))  # In[7]:  values.iloc[0] = [s.subs(x, i) for i in columns]  values.iloc[:,0] = [g1.subs(t, i) for i in index]  values.iloc[:,-1] = [g2.subs(t, i) for i in index]  # In[8]:  for row\_pos in range(1, len(index)):  for col\_pos in range(1, len(columns) - 1):  values[col\_pos][row\_pos] = (  (tetta / (h \* h)) \* (values[col\_pos-1][row\_pos-1] + values[col\_pos+1][row\_pos-1]) +  (1 - 2 \* tetta / (h \* h)) \* (values[col\_pos][row\_pos-1]) +  tetta \* f.subs(x, columns[col\_pos])  )  # In[12]:  values.applymap(lambda x: round(x, 6))  # In[10]:  x = np.arange(10)  fig = plt.figure()  ax = plt.subplot(111)  for i in np.linspace(0, len(index) - 1, 10):  line, = ax.plot(columns, values.iloc[int(i)], label='t={}'.format(index[int(i)]))  ax.legend(loc='best', bbox\_to\_anchor=(0.5, 1.05),  ncol=3, fancybox=True, shadow=True)  plt.show() |

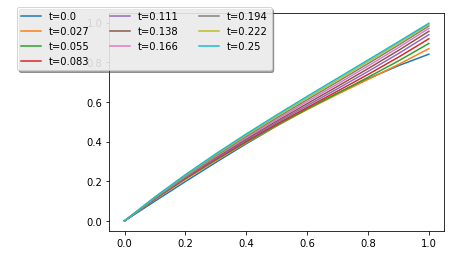
# Анализ полученных результатов

## В результате работы алгоритма получена таблица приближений функции f(x, t) а заданных диапазонах изменения t и х:

## Шаг по оси х составил 0.1, таким образом посчитано приближение для 10 точек отрезка [0, 1]. По оси T посчитано 250 точек на интервале [0, 0.25]

## 

## …

График поведения f(x ,t) при фиксированном значении t: