МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

Федеральное государственное бюджетное

образовательное учреждение

высшего образования

«ВЯТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ФАКУЛЬТЕТ КОМПЬЮТЕРНЫХ И ФИЗИКО-МАТЕМАТИЧЕСКИХ НАУК

КАФЕДРА ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ

Отчёт по лабораторной работе №7

по дисциплине «Математические модели в естествознании и методы их исследования»

Численное решение уравнения теплопроводности методом конечных разностей.

Выполнил: студент группы ФИб-4302-51-00 \_\_\_\_\_\_\_\_\_\_\_ / Савин Д. А/

Проверил: к. б. н., доцент каф. ПМиИ \_\_\_\_\_\_\_\_\_\_\_/ Чупраков П. Г./

Киров 2021

# Цель работы

**Цель работы:** Получение навыков разработки и реализации программ на основе математической модели объекта.

**План:**

1. Ознакомиться с теоретическим материалом.
2. Разработать программу, рассчитывающую и визуализирующую уравнения теплопроводности методом конечной разности.
3. Построить модель для выданного варианта.
4. Оформить отчёт по результатам исследований.

# Краткое описание модели

Дана начально-граничная задача для неоднородного уравнения теплопроводности:

где

Изображение выглядит как стол

Автоматически созданное описание

Вариант 9

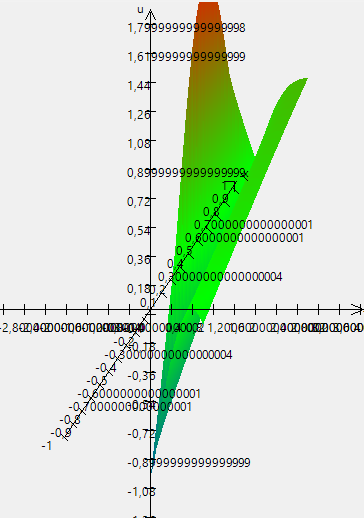
Система решена явной и неявной двухслойных схем.

В явной схеме шаг координаты – 1/30, шаг времени 1/10000

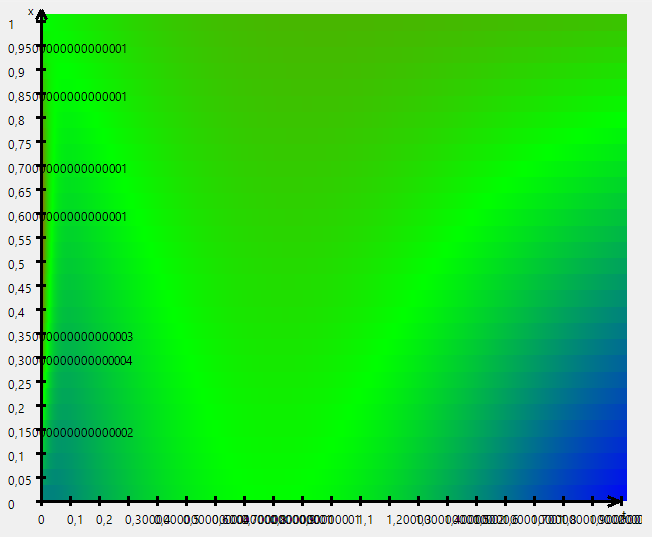
В неявной схеме используется метод Гаусса для решения СЛАУ. Шаг координаты 1/400 шаг времени 1/1000.

# Результаты исследования

1. Явная схема

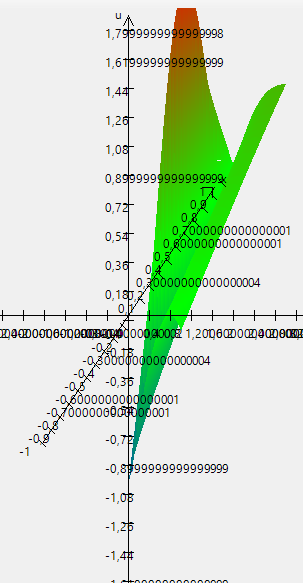


x-координата, t-время, u-температура, цвет поверхности меняется от температуры

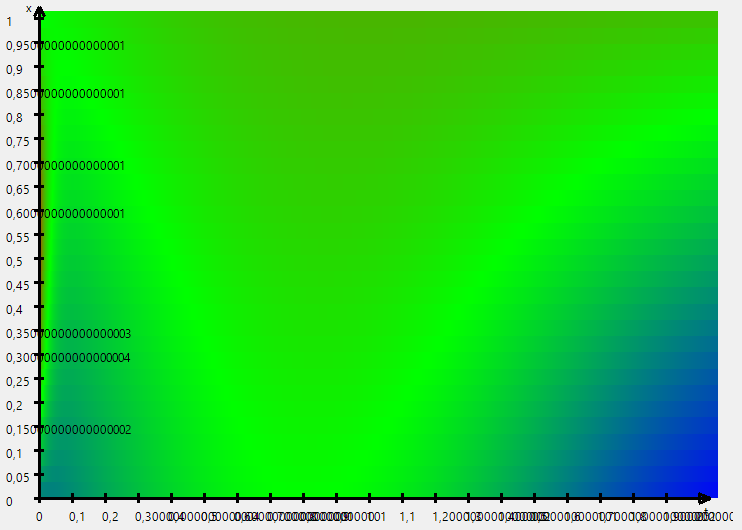


x-координата, t-время, цвет поверхности меняется от температуры

1. Неявная схема



x-координата, t-время, u-температура, цвет поверхности меняется от температуры



x-координата, t-время, цвет поверхности меняется от температуры

Результат – один конец стержня нагревается, другой остывает, причем температура у второго манятся быстрее чем у первого.

Сравнение: явная схема совпадает с неявной при выбранных шагах, но требует подбора шагов для того, чтобы дать правильное решение.

Неявная схема дает корректное решение при любых шагах, но требует намного большее количество расчетов.

# Заключение

В результате выполнения лабораторной работы была исследована и разработана модель уравнения теплопроводности и смоделирован случай нагревания одного конца стержня, и остывания второго.

# Листинг

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.IO;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace task

{

public partial class Form1 : Form

{

double mod = 2;

double coordinate = 0;

double u = 0;

double t = 0;

int n = 30;

double maxt = 2;

int m;

double maxCoordinate = 1;

double maxTemp = 0;

double[,] tempValues;

double gam;

public Form1()

{

InitializeComponent();

}

private double Function0(double x, double t)

{

//return -Math.Sqrt(x);

return x - Math.Cos(2\*Math.PI\*x);

}

private double Function1(double x, double t)

{

//return 1 - Math.Exp(-t);

return Math.Sin(2 \* t) - 1;

}

private double Function2(double x, double t)

{

//return -Math.Sqrt(1 + t);

return t / (1 + t \* t);

}

private double FunctionG(double x, double t)

{

return Math.Log(3 \* x + t \* Math.Exp(-t));

}

private void getLayerNYA(int index)

{

double[,] slau = new double[n - 1, n];

slau[0, 0] = -(1 + 2 \* gam);

slau[0, 1] = gam;

slau[0, n - 1] = -tempValues[1, index - 1] - maxt / m \* FunctionG((double)1 / n, index) - gam \* tempValues[0, index];

for (int i = 1; i < n - 2; i++)

{

slau[i, i - 1] = gam;

slau[i, i] = -(1 + 2 \* gam);

slau[i, i + 1] = gam;

slau[i, n - 1] = -tempValues[i + 1, index - 1] - maxt / m \* FunctionG((double)(i + 1) / n, index);

}

slau[n - 2, n - 3] = gam;

slau[n - 2, n - 2] = -(1 + 2 \* gam);

slau[n - 2, n - 1] = -tempValues[n - 1, index - 1] - maxt / m \* FunctionG((double)(n - 1) / n, index) - gam \* tempValues[n, index];

for (int i = 1; i < n - 1; i++)

{

slau[i, i] = slau[i, i] - slau[i - 1, i] \* slau[i, i - 1] / slau[i - 1, i - 1];

slau[i, n - 1] = slau[i, n - 1] - slau[i - 1, n - 1] \* slau[i, i - 1] / slau[i - 1, i - 1];

slau[i, i - 1] = slau[i, i - 1] - slau[i - 1, i - 1] \* slau[i, i - 1] / slau[i - 1, i - 1];

}

tempValues[n - 1, index] = slau[n - 2, n - 1] / slau[n - 2, n - 2];

maxTemp = Math.Max(Math.Abs(tempValues[n - 1, index]), maxTemp);

for (int i = n - 3; i >= 0; i--)

{

tempValues[i + 1, index] = (slau[i, n - 1] - slau[i, i + 1] \* tempValues[i + 2, index]) / slau[i, i];

maxTemp = Math.Max(Math.Abs(tempValues[i + 1, index]), maxTemp);

}

}

private void getLayerYA(int index)

{

for (int i = 1; i < n; i++)

{

tempValues[i, index] = gam \* tempValues[i - 1, index - 1] + (1 - 2 \* gam) \* tempValues[i, index - 1] + gam \* tempValues[i + 1, index - 1] + maxt / m \* FunctionG((double)i / n, index - 1);

}

}

private void toolStripButton1\_Click(object sender, EventArgs e)

{

m = 10000 \* (int)(maxt);

tempValues = new double[n + 1, (int)m + 1];

gam = 4 \* (maxt / m) / ((double)1 / n \* 1 / n);

/\*try

{\*/

for (int j = 0; j <= n; j++)

{

tempValues[j, 0] = Function0(maxCoordinate / n \* j, 0);

maxTemp = Math.Max(Math.Abs(tempValues[j, 0]), maxTemp);

}

for (int i = 0; i <= m; i++)

{

tempValues[0, i] = Function1(0, maxt / m \* i);

maxTemp = Math.Max(Math.Abs(tempValues[0, i]), maxTemp);

tempValues[n, i] = Function2(1, maxt / m \* i);

maxTemp = Math.Max(Math.Abs(tempValues[n, i]), maxTemp);

}

for (int i = 1; i <= m; i++)

{

getLayerNYA(i);

//getLayerYA(i);

}

maxt \*= mod;

int X0, Y0;

X0 = 25;

Graphics graph = pictureBox2.CreateGraphics();

Pen myPen = new Pen(Brushes.Black, 1);

graph.Clear(BackColor);

X0 = 25;

Y0 = pictureBox2.Height - 25;

for (int i = 0; i <= m / mod - 1; i++)

{

for (int j = 0; j <= n - 1; j++)

{

int R, G, B;

if (tempValues[j, i] > 0)

{

R = (int)(255 / maxTemp \* tempValues[j, i]);

G = (int)(255 - 255 / maxTemp \* tempValues[j, i]);

B = 0;

}

else

{

R = 0;

G = (int)(255 - 255 / maxTemp \* Math.Abs(tempValues[j, i]));

B = (int)(255 / maxTemp \* Math.Abs(tempValues[j, i]));

}

Brush b = new SolidBrush(Color.FromArgb((byte)R, (byte)G, (byte)B));

PointF[] p = {

new PointF( (pictureBox2.Width) / 2 + (float)(((float)i/ mod / m + (float)j / n / maxCoordinate / 2) \* (pictureBox2.Width / 2 - 25)),

(Y0 + 25) / 2 - (float)((tempValues[j, i] / maxTemp + (float)j / n / maxCoordinate /2) \* (pictureBox2.Height / 2 - 25))),

new PointF((pictureBox2.Width) / 2 + (float)(((float)(i+1)/ mod / m + (float)j / n / maxCoordinate /2) \* (pictureBox2.Width / 2 - 25)),

(Y0 + 25) / 2 - (float)((tempValues[j, i+1] / maxTemp + (float)j / n / maxCoordinate / 2) \* (pictureBox2.Height / 2 - 25))),

new PointF((pictureBox2.Width) / 2 + (float)(((float)(i+1)/ mod / m + (float)(j+1) / n / maxCoordinate /2) \* (pictureBox2.Width / 2 - 25)),

(Y0 + 25) / 2 - (float)((tempValues[j+1, i+1] / maxTemp + (float)(j+1) / n / maxCoordinate / 2) \* (pictureBox2.Height / 2 - 25))),

new PointF((pictureBox2.Width) / 2 + (float)(((float)i/ mod / m + (float)(j+1) / n / maxCoordinate / 2) \* (pictureBox2.Width / 2 - 25)),

(Y0 + 25) / 2 - (float)((tempValues[j+1, i] / maxTemp+ (float)(j+1) / n / maxCoordinate / 2) \* (pictureBox2.Height / 2 - 25))) };

graph.FillPolygon(b, p);

}

}

int Z0X = (pictureBox2.Width / 4 + 32);

int Z0Y = (pictureBox2.Height \* 3 / 4 - 32);

graph.DrawLine(myPen, X0, (Y0 + 25) / 2, pictureBox2.Width - 25, (Y0 + 25) / 2);

graph.DrawLine(myPen, (pictureBox2.Width) / 2, Y0, (pictureBox2.Width) / 2, 23);

graph.DrawLine(myPen, pictureBox2.Width / 4 + 32, pictureBox2.Height \* 3 / 4 - 32, pictureBox2.Width \* 3 / 4 - 32, pictureBox2.Height / 4 + 32);

int x = (pictureBox2.Width - 50) / 20;

int y = (pictureBox2.Height - 50) / 20;

float zx = (pictureBox2.Width - 100) / 40;

float zy = (pictureBox2.Height - 100) / 40;

int Y = Y0;

int X = X0;

float ZX = Z0X;

float ZY = Z0Y;

for (int i = 0; i <= 20; i++)

{

graph.DrawLine(myPen, (pictureBox2.Width) / 2 - 5, Y, (pictureBox2.Width) / 2 + 5, Y);

graph.DrawLine(myPen, X, (Y0 + 25) / 2 - 5, X, (Y0 + 25) / 2 + 5);

if (i != 10)

graph.DrawString((Math.Round(maxt / 10, 2) \* (i - 10)).ToString(), Font, Brushes.Black, (float)(X - 8), (Y0 + 25) / 2 + 10);

if (i != 10)

graph.DrawString((Math.Round(maxTemp / 10, 2) \* (i - 10)).ToString(), Font, Brushes.Black, (pictureBox2.Width) / 2 - 25, (float)(Y - 5));

if (i != 10)

{

graph.DrawLine(myPen, ZX - 3, ZY - 3, ZX + 3, ZY + 3);

graph.DrawString((Math.Round(maxCoordinate / 10, 2) \* (i - 10)).ToString(), Font, Brushes.Black, (float)(ZX - 25), (float)(ZY));

}

if (i == 10)

graph.DrawString((Math.Round(maxt / 10, 2) \* (i - 10)).ToString(), Font, Brushes.Black, (float)(X - 13), (Y0 + 25) / 2 + 10);

X += x;

Y -= y;

ZX += zx;

ZY -= zy;

}

graph.DrawString("t", Font, Brushes.Black, (float)(X - 20), (Y0 + 25) / 2 - 5);

graph.DrawString("x", Font, Brushes.Black, (pictureBox2.Width) / 4 \* 3 - 25, (float)(Y + pictureBox1.Height / 4 + 35));

graph.DrawString("u", Font, Brushes.Black, (pictureBox2.Width) / 2 - 15, (float)(Y + 5));

graph.DrawLine(myPen, pictureBox2.Width - 35, (Y0 + 25) / 2 + 5, pictureBox2.Width - 23, (Y0 + 25) / 2);

graph.DrawLine(myPen, pictureBox2.Width - 35, (Y0 + 25) / 2 - 5, pictureBox2.Width - 23, (Y0 + 25) / 2);

graph.DrawLine(myPen, (pictureBox2.Width) / 2 - 5, 30, (pictureBox2.Width) / 2, 20);

graph.DrawLine(myPen, (pictureBox2.Width) / 2 + 5, 30, (pictureBox2.Width) / 2, 20);

graph.DrawLine(myPen, (pictureBox2.Width / 4 \* 3) - 32, pictureBox2.Height / 4 + 32, (pictureBox2.Width / 4 \* 3) - 32, pictureBox2.Height / 4 + 42);

graph.DrawLine(myPen, (pictureBox2.Width / 4 \* 3) - 32, pictureBox2.Height / 4 + 32, (pictureBox2.Width / 4 \* 3) - 42, pictureBox2.Height / 4 + 32);

maxt /= mod;

X0 = 35;

Y0 = pictureBox1.Height - 25;

graph = pictureBox1.CreateGraphics();

graph.Clear(BackColor);

myPen = new Pen(Brushes.Black, 3);

graph.DrawLine(myPen, X0, Y0, pictureBox1.Width - 25, Y0);

graph.DrawLine(myPen, X0, Y0, X0, 23);

for (int i = 0; i <= m - 1; i++)

{

for (int j = 0; j <= n - 1; j++)

{

int R, G, B;

if (tempValues[j + 1, i + 1] > 0)

{

R = (int)(255 / maxTemp \* tempValues[j + 1, i + 1]);

G = (int)(255 - 255 / maxTemp \* tempValues[j + 1, i + 1]);

B = 0;

}

else

{

R = 0;

G = (int)(255 - 255 / maxTemp \* Math.Abs(tempValues[j + 1, i + 1]));

B = (int)(255 / maxTemp \* Math.Abs(tempValues[j + 1, i + 1]));

}

Brush b = new SolidBrush(Color.FromArgb((byte)R, (byte)G, (byte)B));

PointF[] p = {

new PointF( (X0) + (float)(((float)i/ m) \* (pictureBox1.Width - 50)),

(Y0) - (float)(((float)j / n) \* (pictureBox1.Height - 50))),

new PointF((X0) + (float)(((float)(i+1) / m) \* (pictureBox1.Width - 50)),

(Y0) - (float)(((float)j / n) \* (pictureBox1.Height - 50))),

new PointF((X0) + (float)(((float)(i+1) / m) \* (pictureBox1.Width - 50)),

(Y0) - (float)(((float)(j+1) / n) \* (pictureBox1.Height - 50))),

new PointF((X0) + (float)(((float)i / m) \* (pictureBox1.Width - 50)),

(Y0 ) - (float)(((float)(j+1) / n) \* (pictureBox1.Height - 50))) };

graph.FillPolygon(b, p);

}

}

graph.DrawLine(myPen, X0, Y0, pictureBox1.Width - 25, Y0);

graph.DrawLine(myPen, X0, Y0, X0, 23);

x = (pictureBox1.Width - 50) / 20;

y = (pictureBox1.Height - 50) / 20;

Y = Y0;

X = X0;

for (int i = 0; i <= 20; i++)

{

graph.DrawLine(myPen, X0 - 5, Y, X0 + 5, Y);

graph.DrawLine(myPen, X, Y0 - 5, X, Y0 + 5);

graph.DrawString((Math.Round(maxt / 20, 2) \* i).ToString(), Font, Brushes.Black, (float)(X - 5), Y0 + 10);

graph.DrawString((Math.Round(maxCoordinate / 20, 2) \* i).ToString(), Font, Brushes.Black, X0 - 35, (float)(Y - 5));

X += x;

Y -= y;

}

graph.DrawString("t", Font, Brushes.Black, (float)(X - 30), Y0 + 5);

graph.DrawString("x", Font, Brushes.Black, X0 - 15, (float)(Y + 5));

graph.DrawLine(myPen, pictureBox1.Width - 35, Y0 + 5, pictureBox1.Width - 23, Y0);

graph.DrawLine(myPen, pictureBox1.Width - 35, Y0 - 5, pictureBox1.Width - 23, Y0);

graph.DrawLine(myPen, X0 - 5, 30, X0, 20);

graph.DrawLine(myPen, X0 + 5, 30, X0, 20);

/\*}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}\*/

}

}

}