Министерство образования Республики Беларусь

Учреждение образования «Белорусский Государственный Университет Информатики и Радиоэлектроники»

Кафедра «Информатика»

Отчет по лабораторной работе № 1

ЧИСЛЕННОЕ РЕШЕНИЕ СТАЦИОНАРНЫХ И НЕСТАЦИОНАРНЫХ ЗАДАЧ ТЕПЛОПРОВОДНОСТИ

по предмету «Методы численного анализа»

Студент: Паньков Е.В. гр. 052002

Руководитель: Анисимов В. Я.

Минск 2013

**Цель: изучить алгоритмы решения стационарных и нестационарных задач теплопроводности.**

**Задание 1. Промоделировать стационарные процессы теплопроводности стержня в зависимости от входных данных задачи**



**ПОРЯДОК РЕШЕНИЯ ЗАДАЧИ**:

1. Представить коэффициент теплопроводности *K(x)* в виде функции двух переменных *x* и *с*: *K(x)=K(x,c),* где *с* - параметр*.*

2. При заданных в индивидуальном варианте функциях *K(x) (*чтосоответствует *K(x,*1*)), f(x)* и значениях *UA, UB* найти аналитическое решение задачи символьно.

3. Изменяя значения параметра *с* в коэффициенте теплопроводности, найти решения задачи для наборов параметров 1-3 (см. таблицу ниже).

4. На одном чертеже построить графики найденных решений. Cравнить полученные результаты.

5. Аналогично п.2, найти аналитическое решение для набора параметров 4. На одном чертеже построить графики решений для наборов 1 и 4. Cравнить полученные результаты.

6. Изменяя граничные условия *UA, UB*, построить решения для наборов параметров 5-7.

Таблица наборов параметров

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Параметры | 1 набор | 2 набор | 3 набор | 4 набор | 5 набор | 6 набор | 7 набор |
| *с* | 1 | 2 | 0.1 | 1 | 1 | 1 | 1 |
| *K(x)* | *k(x)* | *ck(x)* | *ck(x)* | 1*/k(x)* | *k(x)* | *k(x)* | *k(x)* |
| *UA* | *ua* | *ua* | *ua* | *ua* | *-ua* | *ua* | *-ua* |
| *UB* | *ub* | *ub* | *ub* | *ub* | *ub* | *-ub* | *-ub* |

**Уравнение теплопроводности**



где





- плотность тепловых источников, - коэффициент теплопроводности

Если , то мы имеем стационарное уравнение теплопроводности

 - уравнение Пуассона.

Если плотность тепловых источников тоже равна 0, то получаемое уравнение  называется уравнением Лапласа.

Стационарное уравнение теплопроводности может быть решено, если известны граничные условия.

Это могут быть или значения искомой функции на границе  (задача Дирихле).

Или значения потока тепла на границе  (задача Неймана)

Либо смешанные условия.

**Решение задачи**

% Data

clear

syms x u C1 C2

ua\_ = 3;

ub\_ = 4;

kx\_ = x ^ 2;

v\_c = [1, 2, 0.1, 1, 1, 1, 1];

v\_kx = [kx\_, v\_c(2) \* kx\_, v\_c(3) \* kx\_, 1 / kx\_, kx\_, kx\_, kx\_];

v\_ua = [ua\_, ua\_, ua\_, ua\_, -ua\_, ua\_, -ua\_];

v\_ub = [ub\_, ub\_, ub\_, ub\_, ub\_, -ub\_, ub\_];

a = 0.1;

b = 2;

f = x ^ (1/4) + 4;

figure

hold on

sols = sym('a', [1 7]);

for v = 1:7

eq = int(-(int(f, x) + C1) / v\_kx(v)) + C2;

b1 = subs(eq, x, a) - v\_ua(v);

b2 = subs(eq, x, b) - v\_ub(v);

[CC1, CC2] = solve(b1, b2, C1, C2);

eq = subs(eq, 'C1', CC1);

eq = subs(eq, 'C2', CC2);

uu = solve(eq - u, u);

sols(v) = char(uu);

end

xs = a:0.01:b;

subplot(1,3,1);

hold on;

for i = 1:3

plot(xs, subs(sols(i), x, xs));

end

subplot(1,3,2);

hold on;

for i = [1 4]

plot(xs, subs(sols(i), x, xs));

end

subplot(1,3,3);

hold on;

for i = [5 6 7]

plot(xs, subs(sols(i), x, xs));

end

display(sols);

Аналитические решения вариантов:

486325434498296385/42784196460019712 - 16876000687650395/(10696049115004928\*x) - (16\*x^(1/4))/5 - 16\*log(x^(1/4))

659714020152060481/85568392920039424 - 18001900594493019/(21392098230009856\*x) - (8\*x^(1/4))/5 - 8\*log(x^(1/4))

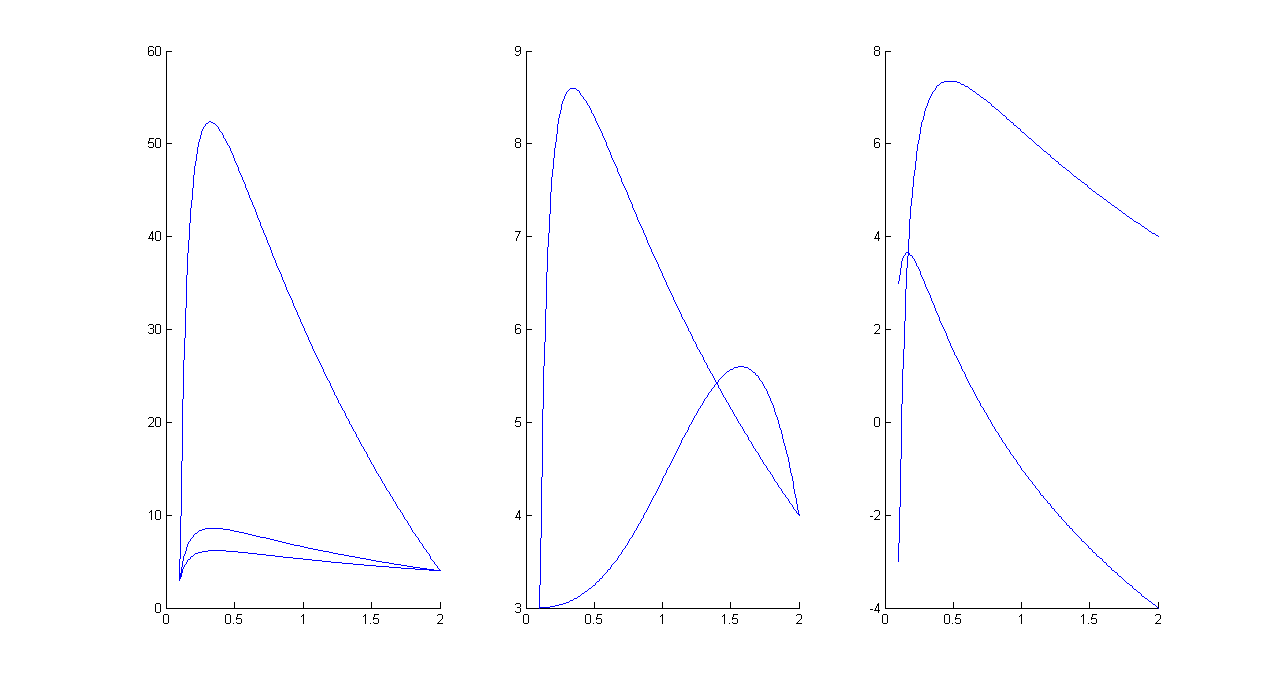
206422317131192945/2674012278751232 - 19828363464365043/(1337006139375616\*x) - 32\*x^(1/4) - 160\*log(x^(1/4))

539920865856350714059/180121467096682987520 - x^3\*(x + (16\*x^(5/4))/85 - 3707633687141339641941/1440971736773463900160)

499836233380407873/42784196460019712 - 23631400128706139/(10696049115004928\*x) - (16\*x^(1/4))/5 - 16\*log(x^(1/4))

6633550753087195/2251799813685248 - 414147443837337/(562949953421312\*x) - (16\*x^(1/4))/5 - 16\*log(x^(1/4))

499836233380407873/42784196460019712 - 23631400128706139/(10696049115004928\*x) - (16\*x^(1/4))/5 - 16\*log(x^(1/4))



**Задание 2. Найти приближенное решение краевой задачи методом конечных разностей:**



c заданной точностью  и построить его график.

**ПОРЯДОК РЕШЕНИЯ ЗАДАЧИ**:

1. Cоставить разностную схему второго порядка точности и выписать коэффициенты матрицы системы уравнений и коэффициенты правой части.

2. Подготовить тестовый пример и провести расчет для него. Построить на одном чертеже графики приближенного и точного решений для тестового примера. После проверки правильности работы программы перейти к решению основной задачи.

3. Для отыскания решения задачи с заданной точностью произвести расчет с начальным шагом *h*, затем уменьшить шаг вдвое. Вывести на экран два соседних приближенных решения и сравнить результаты. Если заданная точность не достигнута, то продолжить уменьшение шага.

4. Построить график найденного решения и указать шаг, при котором заданная точность достигается.

**Основные этапы метода конечных разностей:**

1) область непрерывного изменения аргумента ([a,b]) заменяется дискретным множеством точек, называемых узлами: http://pers.narod.ru/study/methods/5.files/image071.gif.

2) Искомая функция непрерывного аргумента x, приближенно заменяется функцией дискретного аргумента на заданной сетке, т.е. http://pers.narod.ru/study/methods/5.files/image072.gif. Функция http://pers.narod.ru/study/methods/5.files/image073.gifназывается сеточной.

3) Исходное дифференциальное уравнение заменяется разностным уравнением относительно сеточной функции. Такая замена называется разностной аппроксимацией.

Таким образом, решение дифференциального уравнения сводится к отысканию значений сеточной функции в узлах сетки, которые находятся из решения алгебраических уравнений.

**Аппроксимация производных.**

Для аппроксимации (замены) первой производной можно воспользоваться формулами:

http://pers.narod.ru/study/methods/5.files/image074.gif- правая разностная производная,

http://pers.narod.ru/study/methods/5.files/image075.gif- левая разностная производная,

http://pers.narod.ru/study/methods/5.files/image076.gif- центральная разностная производная.

т.е., возможно множество способов аппроксимации производной.

Все эти определения следуют из понятия производной как предела: http://pers.narod.ru/study/methods/5.files/image077.gif.

Опираясь на разностную аппроксимацию первой производной можно построить разностную аппроксимацию второй производной:

http://pers.narod.ru/study/methods/5.files/image078.gif(3)

Аналогично можно получить аппроксимации производных более высокого порядка.

**Решение задачи**

Для аппроксимации производных воспользуемся формулами:

Разностная схема:

**Вариант 2:**

syms x u Du DDu

p = 0.5 \* x;

q = 1.5 \* x ^ 2;

f = 2.5 \* x ^ 3;

a = 1.5;

b = 5.5;

ua = 0;

ub = 1;

eq = DDu + p \* Du + q \* u - f;

steps = 100;

step = (b-a) / steps;

xs = a:step:b-step;

us = sym('u', [1 steps]);

dscheme = sym('d', [1 steps]);

dscheme(1) = us(1) - ua;

dscheme(steps) = us(steps) - ub;

for i = 2:(steps-1)

dscheme(i) = subs(eq, {x, u, Du, DDu}, {

xs(i), us(i), (us(i+1) - us(i-1)) / step, (us(i+1) + us(i-1) - 2 \* us(i)) / (step ^ 2)

});

end

sol = solve(dscheme);

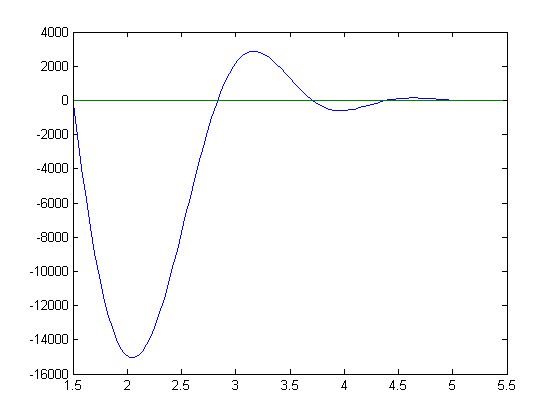
ys = zeros(steps);

for i = 1:steps

ys(i) = getfield(sol, char(us(i)));

end

plot(xs, ys);



**Задание 3. Промоделировать стационарные процессы теплопроводности стержня в зависимости от входных данных задачи – переменного коэффициента теплопроводности *k(x)* и плотности источников тепла *f(x)*:**



**ПОРЯДОК РЕШЕНИЯ ЗАДАЧИ**

Составить разностную схему второго порядка точности для решения указанной задачи.

1. Взять исходные данные из 1-го набора параметров для задачи 1
2. Шаг сетки положить равным .

3. Промоделировать процесс теплопроводности в зависимости от коэффициента *k(x):*

3.1. Полагать, что стержень состоит из 2-x материалов с различными *k(x)*:

, a) *k1<<k2*, б) *k1>>k2*.

3.2. Пусть стержень состоит из 3-x материалов с различными свойствами:



a) *k1<k2 <k3*, б) *k1>k2 >k3*,

в) *k1=k, k2=2k, k3=k*, г) *k1=20k*, *k2=k*, *k3=20k*.

4. Промоделировать процесс теплопроводности в зависимости от правой части – функции *f(x),* предполагая, что *f(x)* - точечный источник тепла. Задать точечный источник тепла можно следующим образом: , где - некоторая константа (мощность источника), - дельта-функция, - точка из отрезка [*a,b*], в которой располагается источник.

Рассмотреть следующие варианты расположения источника:

а) точечный источник поставлен в середину отрезка [*a,b*];

б) два одинаковых по мощности источника поставлены в разные точки

отрезка, симметричные относительно середины отрезка;

в) два различных по мощности источника поставлены симметрично;

г) предложить свой вариант расположения источников.

**Решение задачи**

Уравнение:

Разностная схема:

При k = const:

syms x u Du DDu dkxs kxs

a = 0.5;

b = 1.5;

ua = 0;

ub = 0;

k0 = @(x) (1);

k2 = @(x) (0.05 + heaviside(x - 1) \* 100);

k3 = @(x) (0.15 + heaviside(x - 0.5 - 1/3) \* 5 + heaviside(x - 0.5 - 2/3) \* 25);

f0 = @(x) (1);

f1 = 1 / (abs(x - 1) \* 5);

f2 = 1 / (abs(x - 0.75) \* 25) + 1 / (abs(x - 1.25) \* 25);

f2d = 1 / (abs(x - 0.6) \* 25) + 1 / (abs(x - 1.4) \* 125);

v\_f = [5, 1, f1, f2, f2d];

v\_kx = {k2, k3, k0, k0 };

for v = 4:4

figure

if v == 1

kx = k2;

elseif v == 2

kx = k3;

elseif v == 3

kx = k0;

elseif v == 4

kx = k0;

elseif v == 5

kx = k0;

end

eq = -(dkxs \* Du + kxs \* DDu) - v\_f(v);

eq = -(kxs \* DDu) - v\_f(v);

steps = 25;

step = (b-a) / steps;

xs = a:step:b-step;

us = sym('u', [1 steps]);

dscheme = sym('d', [1 steps]);

dscheme(1) = us(1) - ua;

dscheme(steps) = us(steps) - ub;

for i = 2:(steps-1)

dscheme(i) = subs(eq, {x, u, dkxs, kxs, DDu}, {

xs(i), us(i), (kx(xs(i+1)) - kx(xs(i-1))) / step, kx(xs(i)), (us(i+1) + us(i-1) - 2 \* us(i)) / (step ^ 2)

});

end

sol = solve(dscheme);

ys = zeros(steps);

for i = 1:steps

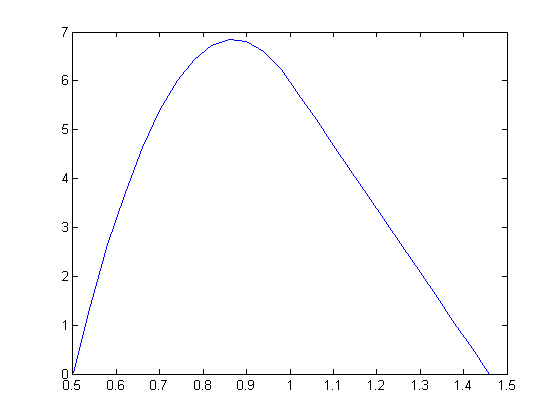
ys(i) = getfield(sol, char(us(i)));

end

plot(xs, ys);

end

Пусть стержень состоит из 2-x материалов с различными *k(x)*:

****

Пусть стержень состоит из 3-x материалов с различными свойствами:

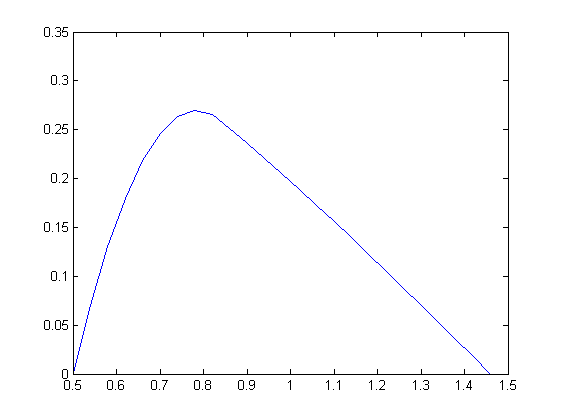
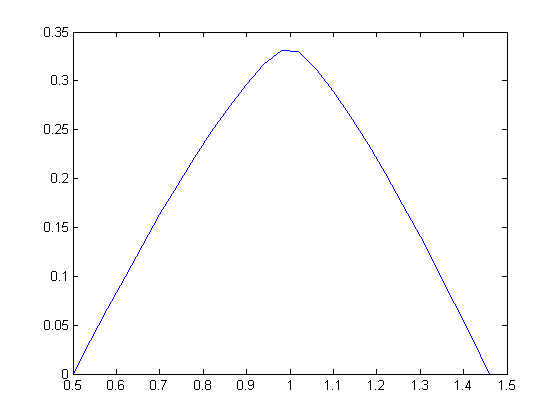
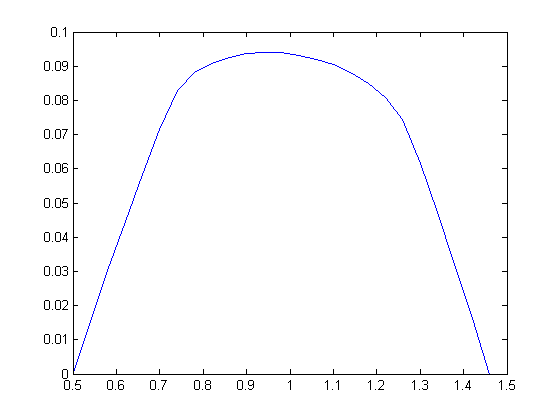


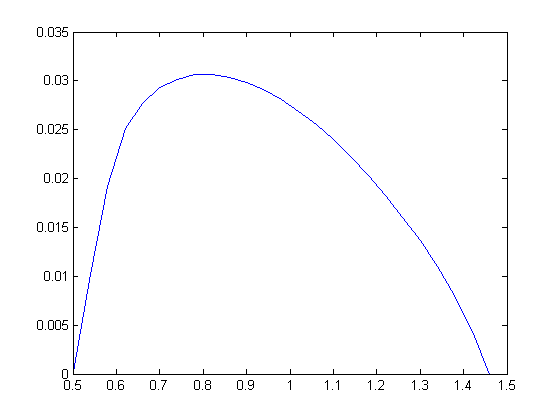
График для одного источника



Двух одинаковых источников



и двух разных источников:



**Задание 4. Методом конечных разностей найти приближенное решение краевой задачи**



с тремя верными значащими цифрами. Решение системы разностных уравнений найти, используя метод прогонки.

**ПОРЯДОК РЕШЕНИЯ ЗАДАЧИ**

1. Использовать разностную схему второго порядка точности.

2. При аппроксимации производных в граничных условиях использовать метод баланса.

**Метод баланса**

Напишем уравнение баланса тепла на отрезке :

Где – поток тепла, – мощность стоков тепла (при источников), - плотность распределения внешних источников (стоков) тепла.

Чтобы получить разностное уравнение, заменим и интеграл, содержащий , линейными комбинациями значений в узлах сетки. Для этого воспользуемся интерполяциями в окрестности узла . Возьмем простейшую интерполяцию

при ,

где есть среднее значение на отрезке длины

Проинтегрируем равенство на отрезке

Предполагая, что при имеем:

Отсюда находим приближенное значение потока

Подставив найденные выражения в уравнение баланса, получим разностную схему

где

**Решение задачи**

syms x u Du DDu

k = 0.5 \* x;

q = 1.5% \* x ^ 2;

f = 2.5 \* x ^ 3;

a = 1.5;

b = 5.5;

eq = -(diff(k) \* Du + DDu \* k) + q \* u - f;

steps = 100;

step = (b-a) / steps;

xs = a:step:b-step;

us = sym('u', [1 steps]);

dscheme = sym('d', [1 steps]);

dscheme(1) = -subs(k, x, a) \* (us(2) - us(1)) / step + 0.5 \* us(1);

dscheme(steps) = -subs(k, x, b) \* (us(steps) - us(steps-1)) / step + 0.5 \* us(steps);

for i = 2:(steps-1)

dscheme(i) = subs(eq, {x, u, Du, DDu}, {

xs(i), us(i), (us(i+1) - us(i-1)) / step, (us(i+1) + us(i-1) - 2 \* us(i)) / (step ^ 2)

});

end

sol = solve(dscheme);

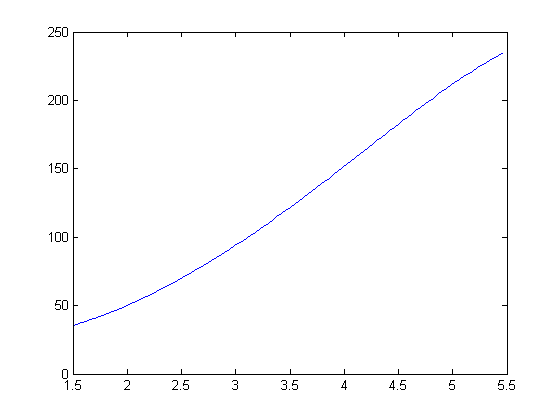
ys = zeros(steps);

for i = 1:steps

ys(i) = getfield(sol, char(us(i)));

end

plot(xs, ys);



**Задание 5. Найти приближенное решение начально-краевой задачи для уравнения теплопроводности**



**Решение задачи**

syms x u Du DDu Dt t

k = 1+x\*0;%0.5 \* x;

f = 0;

eq = (DDu \* k + diff(1\*k) \* Du) + f \* (1 - exp(-t)) - Dt;

a = 0;

b = 1;

ua = 0;

ub = 0;

steps = 50;

step = (b-a) / steps;

tstep = 0.03;

xs = a:step:b-step;

us = sym('u', [1 steps]);

uold = zeros([1 steps]);

for i = 1:steps

uold(i) = (1.25 - xs(i)) ^ 2;

uold(i) = sin(xs(i)\*pi \*2);

uold(i) = heaviside(xs(i) - 0.25) \* heaviside(0.75 - xs(i));

end

t = 0;

figure

hold on;

for st = 1:5

dscheme = sym('d', [1 steps]);

dscheme(1) = us(1) - ua;

dscheme(steps) = us(steps) - ub;

for i = 2:(steps-1)

dscheme(i) = subs(eq, {x, u, Du, DDu, t, Dt}, {

xs(i), us(i), (us(i+1) - us(i-1)) / step, (us(i+1) + us(i-1) - 2 \* us(i)) / (step ^ 2), t, (us(i) - uold(i)) / tstep

});

end

sol = solve(dscheme);

ys = zeros([1 steps]);

for i = 1:steps

ys(i) = getfield(sol, char(us(i)));

uold(i) = ys(i);

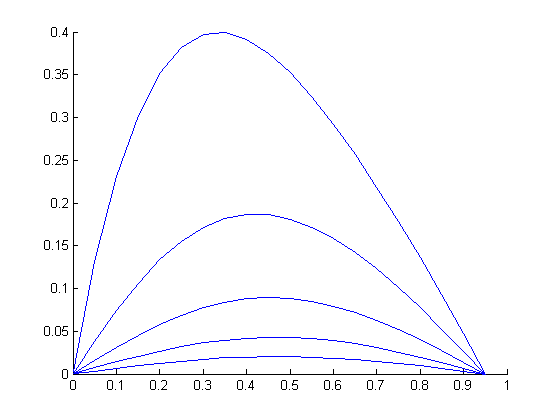
end

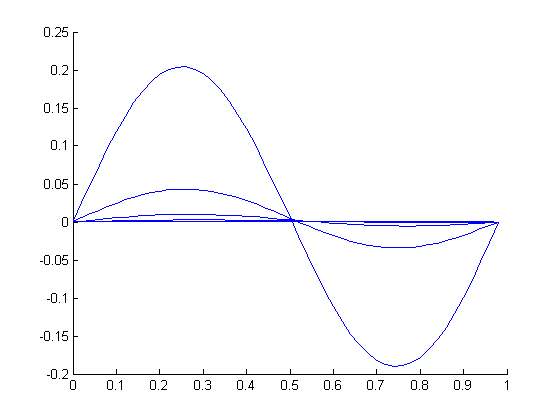
plot(xs, ys);

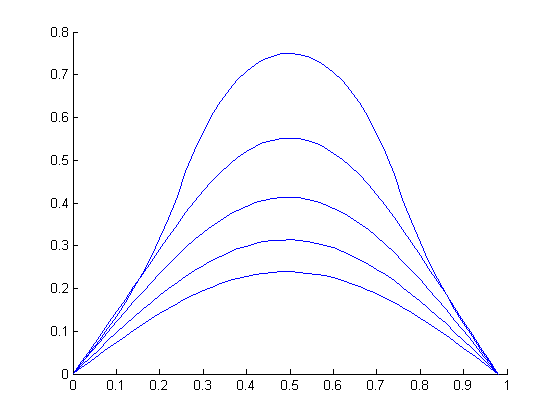
t = t + tstep;

end

Результаты при разных начальных температурах:







Вывод:

В ходе данной лабораторной работы были изучены численные алгоритмы решения стационарных и нестационарных задач теплопроводности.