Санкт-Петербургский государственный политехнический университет
Институт прикладной математики и механики
Кафедра «Теоретическая механика»

РАСЧЁТНОЕ ЗАДАНИЕ

по дисциплине «Вычислительная механика»

Решение задачи теплопроводности методом конечных элементов

Выполнил
студент гр.53604/2 Г.А. Мирошник

Проверил
Ассистент каф «Теоретическая механика» С.А. Ле-Захаров

«21» декабря 2015 г.

Санкт-Петербург

2015

Содержание

Цель работы

Численно решить задачу теплопроводности для одномерного теплоизолированного стержня, используя заданную схему точек при заданных начальных и граничных условиях (см. Рисунок 1).

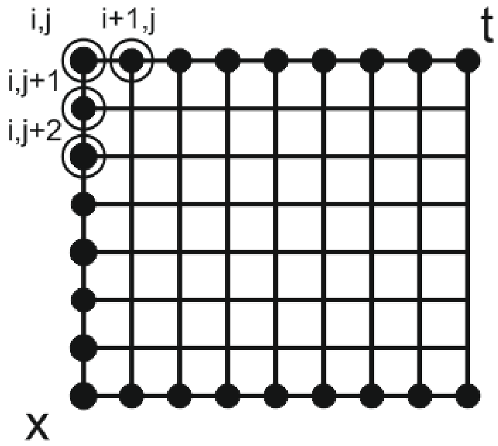

Рисунок 1. Схема задачи. Точки с известной температурой выделены кругами; система точек, использующихся для решения, обозначена окружностями.

Граничные и начальные условия задачи:

$$\begin{cases} T(x,t)|_{t=0} = 1\ \text{К}, x < \dfrac{1}{2} \\ T(x,t)|_{t=0} = 0\ \text{К}, x \geq \dfrac{1}{2} \\ T(x,t)|_{x=0} = 1\ \text{К} \\ T(x,t)|_{x=1} = 0\ \text{К} \end{cases}$$

Решение

В общем виде уравнение теплопроводности

$$\frac{\partial T}{\partial t} - \varkappa \Delta T = f(r,t)\ (1)$$

Так как мы рассматриваем теплоизолированного одномерный стержень, $\Delta T \to \frac{\partial^2 T}{\partial x^2}$, а $f(r,t) \to f(r)$ (начальные условия), следовательно

$$\frac{\partial T}{\partial t} - \varkappa \frac{\partial^2 T}{\partial x^2} = 0\ (2)$$

Для взятия производной в точке (i,j) будем использовать метод правосторонней разности, тогда

$$\frac{\partial T}{\partial t} = \frac{T(i+1,j) - T(i,j)}{\Delta_t}$$

$$\frac{\partial T}{\partial x} = \frac{T(i,j+1) - T(i,j)}{\Delta_x}$$

$$\frac{\partial^2 T}{\partial x^2} = \frac{\frac{T(i,j+2) - T(i,j+1)}{\Delta_x} - \frac{T(i,j+1) - T(i,j)}{\Delta_x}}{\Delta_x} =$$

$$= \frac{T(i,j+2) - 2T(i,j+1) + T(i,j)}{{\Delta_x}^2}$$

$$\frac{T(i+1,j) - T(i,j)}{\Delta_t} - \varkappa \frac{T(i,j+2) - 2T(i,j+1) + T(i,j)}{{\Delta_x}^2} = 0 \; (3)$$

Уравнение для нахождения $T(i+1,j)$:

$$T(i+1,j) = T(i,j) + \Delta_t \cdot \varkappa \frac{T(i,j+2) - 2T(i,j+1) + T(i,j)}{{\Delta_x}^2} \; (4)$$

При помощи уравнения (4), смещаясь вниз по оси x, пока координата j+2 не дойдёт до нижней границы стержня, после чего смещаясь вправо по оси t, можно заполнить часть области, но мы сталкиваемся с ограничениями, связанными с геометрией схемы. Последовательность заполнения и область, которую можно заполнить точкой (i+1,j), можно увидеть на рисунке 2.

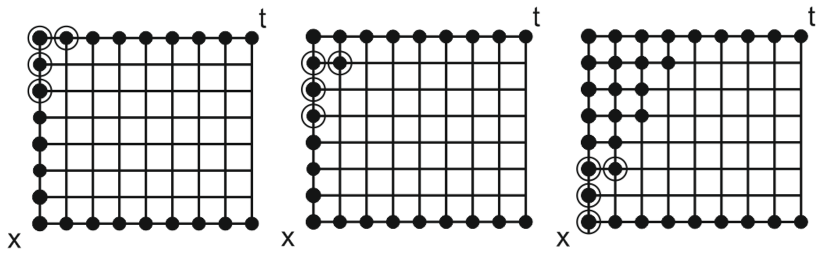

Рисунок 2. Порядок нахождения температуры и область,
которую можно заполнить при помощи точки (i+1,j).

Самый простой способ заполнить всю область – использовать схему, изображённую на рисунке 3.

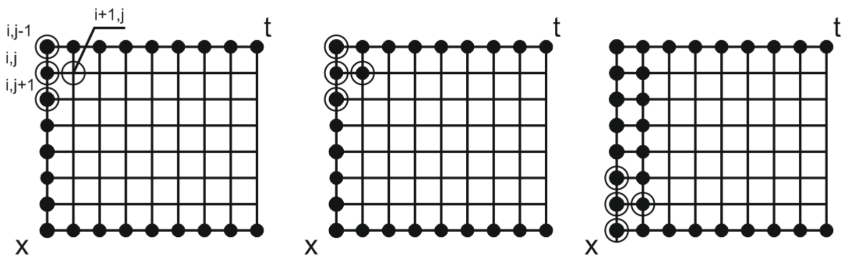

Рисунок 3. Оптимальная схема точек и заполнение с её помощью столбца i+1.

Как видно на рисунке 3, при помощи данной схемы можно заполнить весь столбец без пропусков, и, следовательно – всю область.

Составим уравнение, аналогичное уравнению (4), для новой схемы. В данном случае первую производную по x будем брать по методу левосторонней разности, а вторую – по методу правосторонней разности, чтобы использовать как точку (i,j-1), так и (i,j+1).

$$\frac{\partial T}{\partial x} = \frac{T(i,j) - T(i,j-1)}{\Delta_x}$$

$$\frac{\partial^2 T}{\partial x^2} = \frac{\frac{T(i,j+1) - T(i,j)}{\Delta_x} - \frac{T(i,j) - T(i,j-1)}{\Delta_x}}{\Delta_x} = $$

$$= \frac{T(i,j+1) - 2T(i,j) + T(i,j-1)}{{\Delta_x}^2}$$

$$\frac{T(i+1,j) - T(i,j)}{\Delta_t} - \varkappa \frac{T(i,j+1) - 2T(i,j) + T(i,j-1)}{{\Delta_x}^2} = 0 \quad (5)$$

$$T(i+1,j) = T(i,j) + \Delta_t \cdot \varkappa \frac{T(i,j+1) - 2T(i,j) + T(i,j-1)}{{\Delta_x}^2} \quad (6)$$

На языке Pascal был реализован алгоритм интегрирования (см. приложение 1), были проведены вычисления для $i \in [0..99], j \in [0..9]$ при $\Delta_t =$ 0.001,0.0001,0.0001,0.00001, значение $\varkappa$ выбрано для стали. Графики изменения температуры элементов стержня показаны на рисунках 4-7.

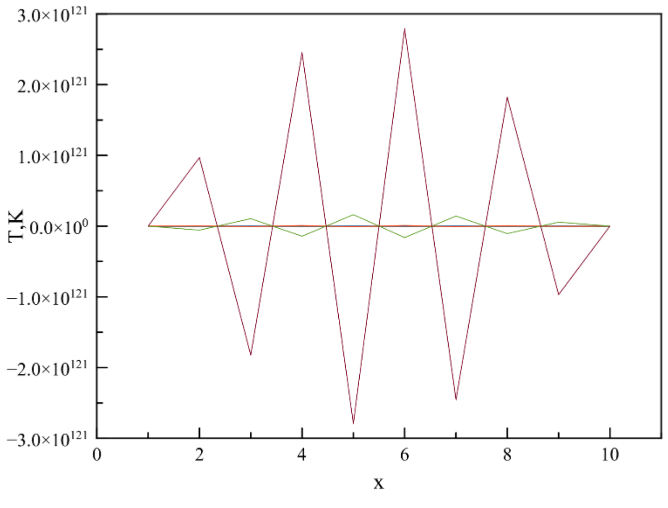

Рисунок 4. График изменения температуры стержня при $\Delta_t = 0.001$

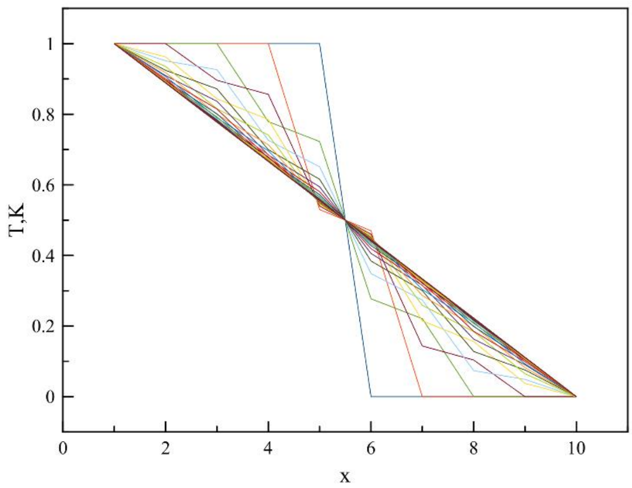

Рисунок 5. График изменения температуры стержня при $\Delta_t = 0.0001$

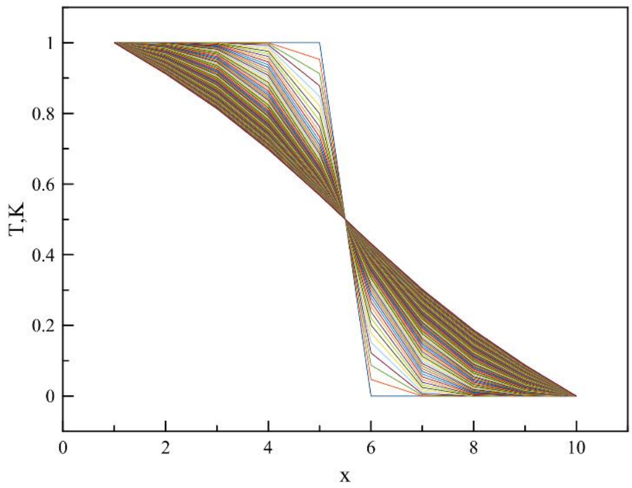

Рисунок 6. График изменения температуры стержня при $\Delta_t = 0.00001$

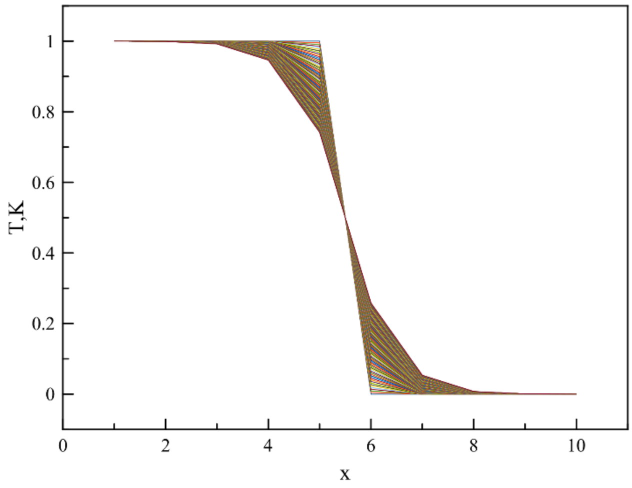

Рисунок 7. График изменения температуры стержня при $\Delta_t = 0.000001$

По рисунку 4 видно, что для шага $\Delta_t = 0.001$ получаются неадекватные результаты. Это связано со значением соотношения $\frac{\varkappa \cdot \Delta_t}{{\Delta_x}^2}$, которое определяет, насколько сильно будет изменяться температура при вычислениях. При заданных значениях $\Delta_x = 0.1$ и $\varkappa = 47$ соотношение можно записать как $4700 \cdot \Delta_t$. Для получения результатов необходимо добиться $\frac{\varkappa \cdot \Delta_t}{{\Delta_x}^2} < 1$, что достигается при шаге 0.0002 и меньше.

Шаг $\Delta_t = 0.0001$ (см. рисунок 5) всё ещё слишком мал, как как уже на первом шаге по времени изменение температуры слишком велико. Оптимален шаг $\Delta_t = 0.00001$ (см. рисунок 6). Дальнейшее уменьшение шага привело к тому, что не удалось завершить процесс за 100 итераций, но позволило отслеживать процесс на малых промежутках времени (см. рисунок 7).

Выводы

Численно решена задача теплопроводности для одномерного стержня. Рассмотрены две разностные схемы, для более удачной из них написана программа на языке Pascal, в программе проведены вычисления для разных шагов по времени, результаты представлены на графиках и проанализированы.

Приложение 1. Листинг программы.

```
program Transfer;

var
  i, j: integer;
  Length: integer := 1; // Длина стержня
  step: real := 0.000001; // Размер шага по времени
  step_n: integer := 100; // Количество шагов по времени
  step_x_n: integer := 10; // Количество разбиений стержня
  T: array [,] of real;
  kappa: real := 47; // Коэффициент теплопроводности,значение для стали
  C: real := (step * kappa / (sqr(Length) / sqr(step_x_n)));

Begin
  T := new real[step_n, step_x_n];
  for j := 0 to step_x_n - 1 do // Задание начальных условий
    if (j <= ((step_x_n div 2) - 1)) then
      T[0, j] := 1
    else T[0, j] := 0;

  for i := 0 to step_n - 1 do // Задание граничных условий
  begin
    T[i, 0] := 1;
    T[i, step_x_n - 1] := 0;
  end;

  for i := 0 to step_n - 2 do
    for j := 1 to step_x_n - 2 do
      T[i + 1, j] := T[i, j] + C * (T[i, j + 1] - 2 * T[i, j] + T[i, j - 1]);

  for j := 0 to step_x_n - 1 do // Вывод данных
  begin
    writeln ();
    for i := 0 to step_n - 1 do
      Write((T[i, j]), ' ');
  end;
end.
```