

Оглавление

Введение	3
1 Аналитическая часть	4
1.1 Постановка задачи	4
1.2 Метод Пикара	4
1.3 Явный метод Эйлера	4
1.4 Неявный метод Эйлера	5
2 Технологическая часть	6
2.1 Выбор ЯП	6
2.2 Листинг кода алгоритмов	6

Введение

Задачи данной лабораторной работы:

- Изучить метод Пикара;
- реализовать метод Пикарда, явный и неявный методы Эйлера для решения уравнения $y'(x) = x^2 + y^2$;
- сравнить методы между собой.

1 | Аналитическая часть

В данной части будут рассмотрены теоретические основы методов Пикара, явного и неявного методов Эйлера.

1.1 Постановка задачи

Пусть поставлена задача Коши:

$$\begin{cases} u'(x) = f(x, u(x)) \\ u(x_0) = u_0 \\ x_0 \leq x \leq x_l \end{cases}$$

1.2 Метод Пикара

Данный метод является представителем класса приближенных методов решения. Идея метода чрезвычайно проста и сводится к процедуре последовательных приближений для решения интегрального уравнения, к которому приводится исходное дифференциальное уравнение.

Проинтегрируем выписанное уравнение

$$u(x) = u_0 + \int_{x_0}^x f(t, u(t)) dt \quad (1.1)$$

Процедура последовательных приближений метода Пикара реализуется согласно следующей схеме.

$$y_s(x) = u_0 + \int_{x_0}^x f(t, y_{s-1}(t)) dt \quad (1.2)$$

причем $y_0(t) = u_0$

1.3 Явный метод Эйлера

Самый простой метод решения уравнения - дискретизация расчетного интервала и замена производной в левой части $\frac{du(x)}{dt}$ разностным аналогом. Для некоторой i -ой точки сетки разностная производная определяется следующим образом:

$$\frac{du(t_i)}{dt} \approx \frac{y_{i+1} - y_i}{h} \quad (1.3)$$

Для того, чтобы схема имела простое решение, правую часть уравнения возьмем в той же точке t_i :

$$\frac{y_{i+1} - y_i}{h} = f(t_i, y_i) \quad (1.4)$$

Таким образом, мы сразу получаем рекуррентную формулу определения нового значения y в точке t_{i+1} , т.е. y_{i+1} по значению y в точке t_{i+1} . Это значение обозначим как y_i , а y_{i+1} запишем как:

$$y_{i+1} = y_i + h \cdot f(x_i, y_i) \quad (1.5)$$

1.4 Неявный метод Эйлера

$$y_{i+1} = y_i + h \cdot f(x_i + 1, y_i + 1) \quad (1.6)$$

Геометрическая интерпретация одного шага этого метода заключается в том, что решение на отрезке $[t_i; t_{i+1}]$ аппроксимируется касательной $y = y_{i+1} + y'(t_{i+1})(t - t_{i+1})$, проведенной в точке (t_{i+1}, y_{i+1}) к интегральной кривой, проходящей через эту точку.

2 | Технологическая часть

2.1 Выбор ЯП

В качестве языка программирования был выбран `golang`. Время работы алгоритмов было замерено с помощью `time`.

2.2 Листинг кода алгоритмов

В данном разделе будут приведены листинги кода решения методом Пикара (Листинг 2.1) и Эйлера (Листинг 2.2)

Листинг 2.1: Метод Пикара

```
1
2 func picard(x float64 , n int)float64{
3     u0 := 0.0
4     answer := 0.0
5     poly := make(map[int]float64)
6     curr := make(map[int]float64)
7     poly[2] = 1.0
8     var res float64
9     for i:=0;i<n;i++){
10         curr = poly_pow(curr)
11         curr[2] = 1.0
12         curr, res = integrate(curr, 0.0, x)
13         answer = u0 + res
14     }
15     return answer
16 }
```

Листинг 2.2: Явный и неявный методы Эйлера

```
1 func euler_explicit(xn float64 , n int)float64{
2     h := xn / float64(n)
3     y:=0.0
4     x:=0.0
5     for i:=0; i<=n;i++){
6         y = y + h*f(x, y)
7         x+=h
8     }
9     return y
```

```

10 }
11 func euler_implicit(xn float64, n int)float64{
12     h := xn / float64(n)
13     y:=0.0
14     x:=0.0
15     var a, b, c, dis, x1 float64
16     for i:=0;i<=n;i++){
17         a = 1; b = -1.0/h; c = 1.0/h*y+(x+h)*(x+h)
18         dis = D(a, b, c)
19         if dis>=0{
20             x1 = (-b - math.Sqrt(dis))/2/a
21         }
22         y = x1
23         x+=h
24     }
25     return y
26 }

```