Санкт-Петербургский национальный исследовательский университет
информационных технологий, механики и оптики
Факультет программной инженерии и компьютерной техники

Вычислительная математика
Лабораторная №4

Выполнил:
Беляков Дмитрий
Группа:
P3210
Преподаватель:
Перл О.В.

Санкт-Петербург
2020

Описание метода Милна

Метод Милна — это многошаговый метод, то есть он использует для вычисления не одну, а несколько ранее вычисленных значений. Для метода требуется начальные значения, которые обычно вычисляются с помощью одношаговых методов (например, Рунге-Кутты). Для метода порядка k требуется k начальных точек. Метод милна также относится к методам прогноза и коррекции. На каждой итерации для очередного y_{i+1} явно (предиктор) вычисляется значение, а после с помощью неявно (корректор) вычисляется уточнение, пока не будет достигнута необходимая точность.

Алгоритм для поиска начальных точек и шага:

1. При помощи метода Рунге-Кутты находим начальные 4 точки
2. Вычисляем значения предиктора и корректора
3. Если модуль разности значений, деленный на 29, больше введенной погрешности, уменьшаем шаг в 2 раза и возвращаемся к пункту 1
4. Вычисляем значение следующих точек по алгоритму Милна

Используемые формулы

$$k_1 = f(x_i, y_i)$$

$$k_2 = f\left(x_i + \frac{h}{2}, y_i + k_1 * \frac{h}{2}\right)$$

$$k_3 = f\left(x_i + \frac{h}{2}, y_i + k_2 * \frac{h}{2}\right)$$

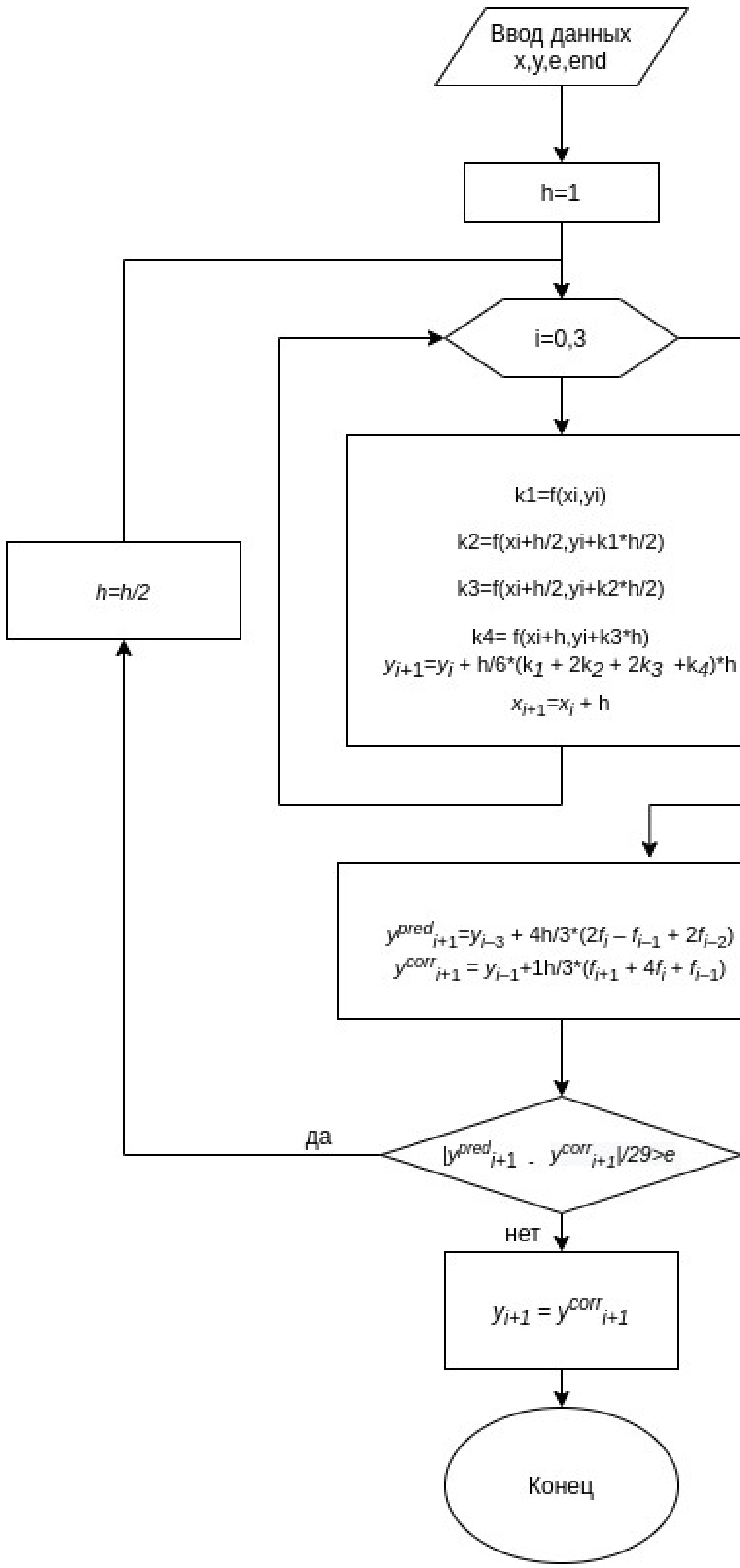
$$k_4 = f(x_i + h, y_i + k_3 * h)$$

$$y_{pred} = y_{i-4} + \frac{4 * h}{3}(2 f_{i-3} - f_{i-2} + 2 f_{i-1})$$

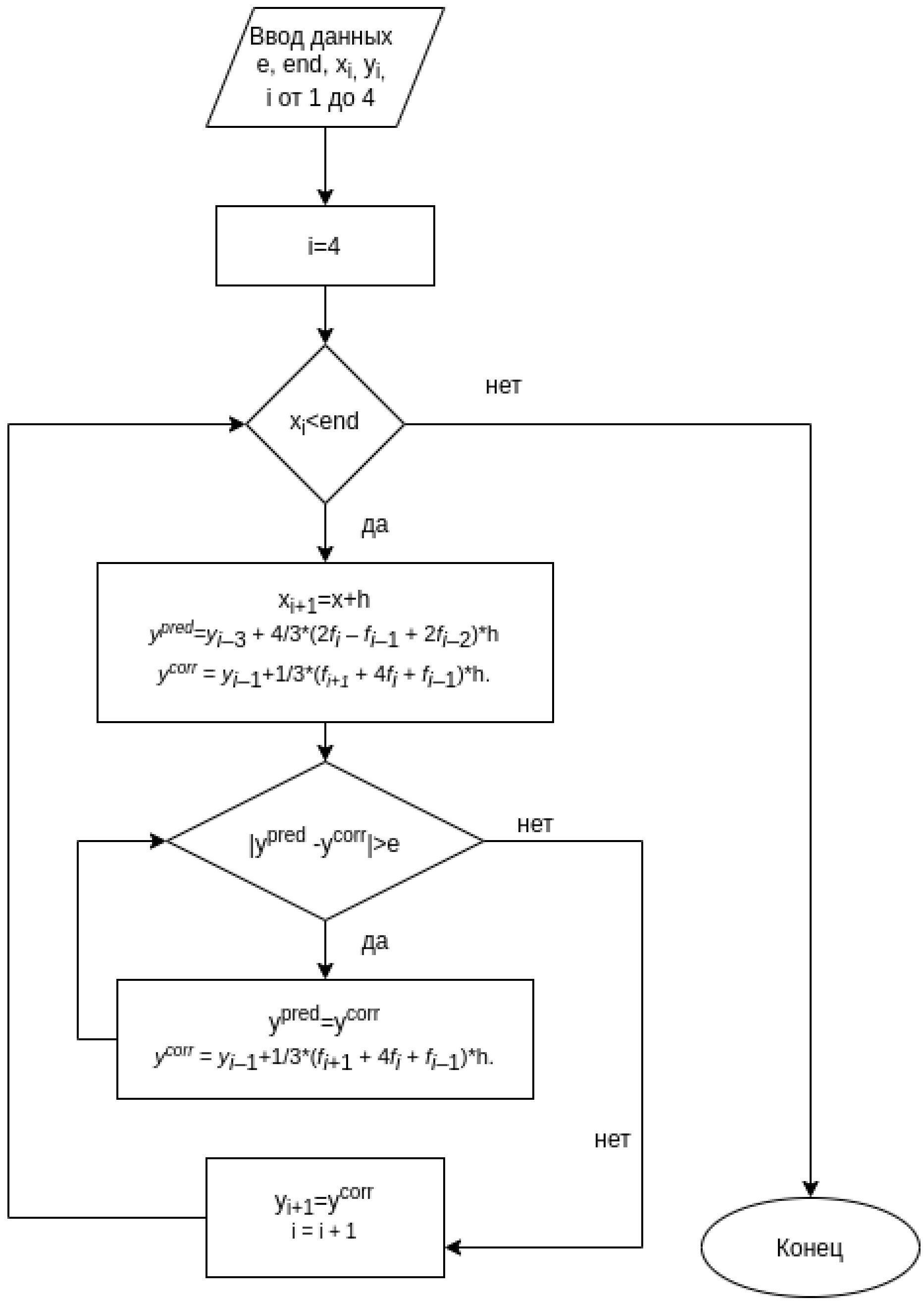
$$y_{corr} = y_{i-2} + \frac{h}{3}(f_{i-2} + 4 f_{i-1} + f_i)$$

$$e_m = \frac{\left|y_m^{pred} - y_m^{corr}\right|}{29}$$

Блок схема численного метода Рунге-Кутты для определения первых точек и шага



Блок схема численного метода Милна



Листинг численного метода

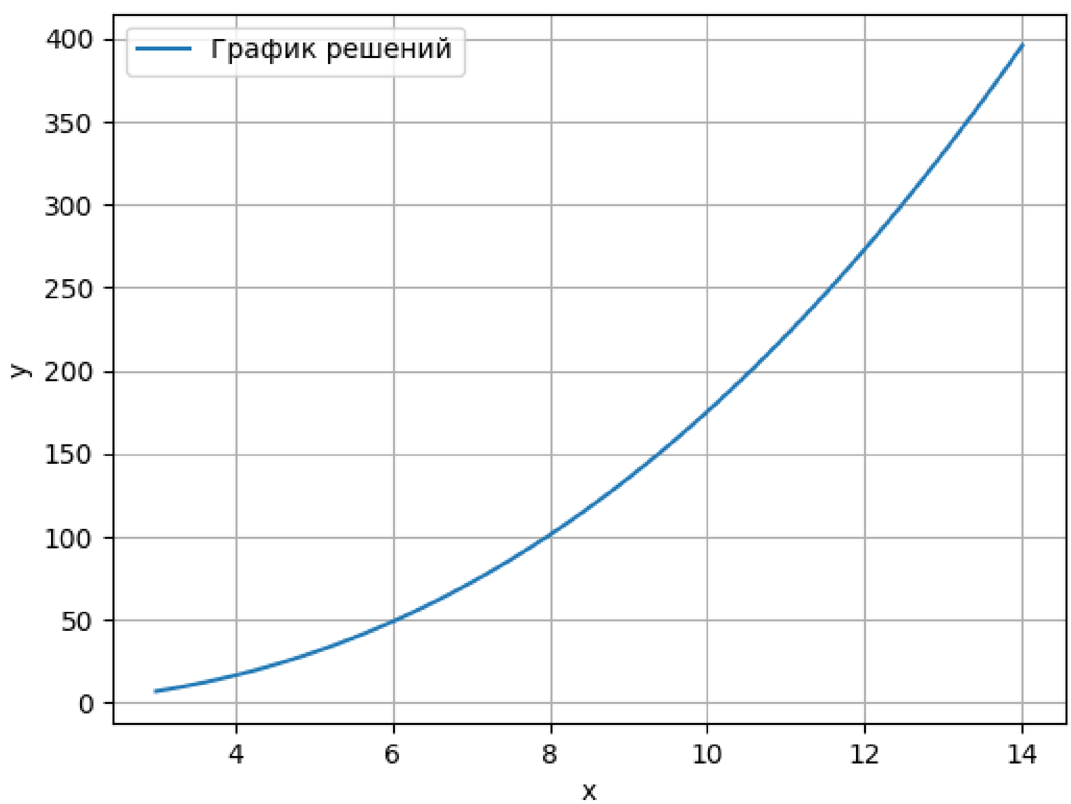
```
limit = 100000

def calcRKmethod(f, x, y, h):
    result = []
    result.append([x,y])
    for i in range(3):
        k1 = f(*result[i])
        k2 = f(result[i][0] + h/2, result[i][1] + k1*h/2)
```

```
        k3 = f(result[i][0] + h/2, result[i][1] + k2*h/2)
        k4 = f(result[i][0] + h  , result[i][1] + k3*h  )
        result.append([
            result[i][0] + h,
            result[i][1] + h/6*(k1 + 2*k2 + 2*k3 + k4)
            ])
    return result

def predict(f, h, points):
    return points[0][1] + 4*h/3*(
        2*f(*points[1]) - f(*points[2]) + 2 * f(*points[3])
    )

def correct(f, h, points):
    return points[0][1] + h/3*(
        f(*points[0]) + 4*f(*points[1]) + f(*points[2])
    )

def initMiln(f, x, y, eps):
    h = 1.0
    i = 0
    while True:
        i+=1
        if i > limit:
            raise Exception("Ошибка. Не удалось вычислить начальные точки и шаг за {0}
итераций".format(limit))
        first_points = calcRKmethod(f, x, y, h)
        y_pr = predict(f, h, first_points[0:])
        first_points.append([
            first_points[3][0] + h,
            y_pr
        ])
        y_cr = correct(f, h, first_points[2:])
        if abs(y_pr-y_cr)/29 <= eps:
            first_points[4][1] = y_cr
            return first_points, h
        h /=2

def calcMiln(f, x, y, eps, end):
    result, h = initMiln(f, x, y, eps)
    x_i = result[4][0]
    i = 4
    while x_i < end:
        i += 1
        x_i += h
        y_pr = predict(f, h, result[i-4:])
        result.append([
            x_i,
            y_pr
```

```
    ])
    y_cr = correct(f, h, result[i-2:])
    j = 0
    while abs(y_cr - result[i][1]) > eps:
        j += 1
        if j > limit:
            raise Exception("Ошибка. Не удалось вычислить y_{0} за {1} итераций".format(i, limit))
        result[i][1] = y_cr
        y_cr = correct(f, h, result[i-2:])
    result[i][1] = y_cr
return result
```

Примеры

```
Выберите функцию:
1. y' = -sin(x) - y
2. y' = 2y/(x+1) + x + 1
3. y' = cos(x) - y
Введите число от 1 до 3
2
Введите начальное условие  - x, y через пробел
3 7
Введите точность
0.00000001
Введите конец отрезка
14
```



Общее решение

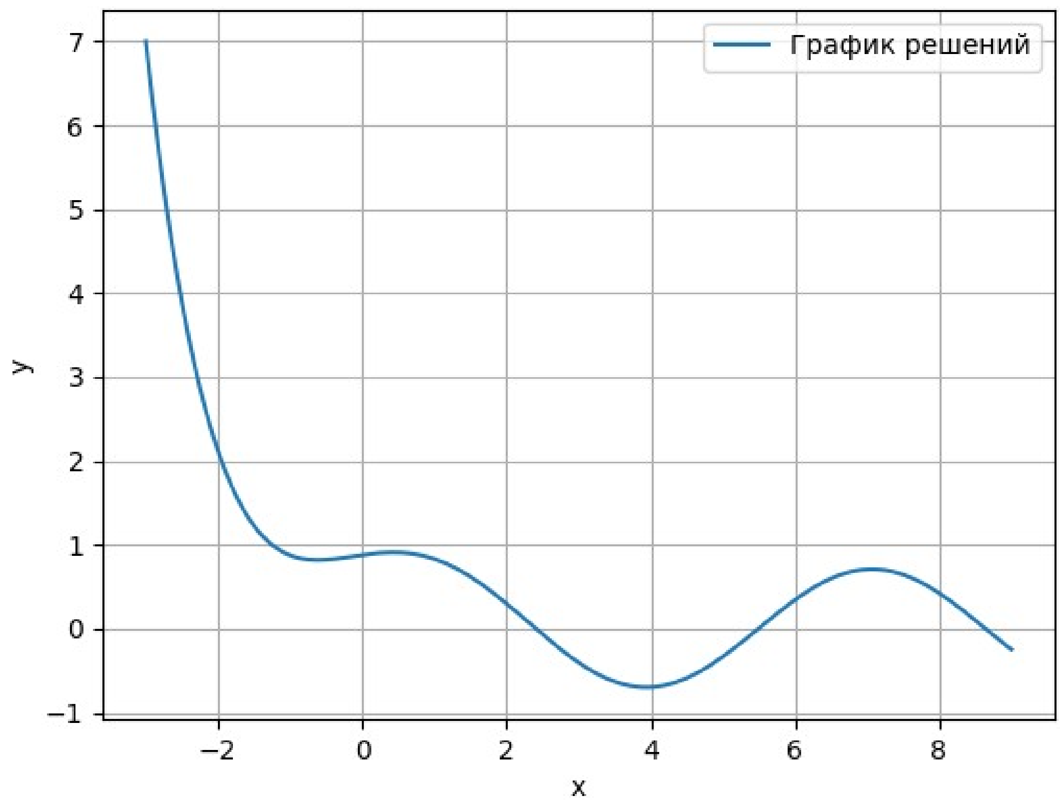
$$y(x)=(x+1)^2(\log(x+1)+c_0)$$

Решение задачи Коши

$$y(x)=(x+1)^2\left(\log(x+1)+\frac{7}{16}-\log 4\right)$$

Выберите функцию:
1. y' = -sin(x) - y
2. y' = 2y/(x+1) + x + 1
3. y' = cos(x) - y

Введите число от 1 до 3
3
Введите начальное условие - x, y через пробел
-3 7
Введите точность
0.00000001
Введите конец отрезка
9



Общее решение

$$y(x)=\frac{\cos x}{2}-\frac{\sin x}{2}+c_0e^{-x}$$

Решение задачи Коши

$$y(x)=1/2(\cos x-(\cos 3+\sin 3-14))e^{-x-3}-\sin x$$

Вывод: метод Милна является многошаговым методом, т. е. для вычисления значений требуется значение не одной предыдущей точки, а нескольких. Поэтому для вычисления первого значения, необходимо вычислить начальные точки, что можно отнести к недостаткам по сравнению с одношаговыми методами, однако засчёт таких вычислений повышается точность. Метод также является методом прогноза-коррекции, это значит, что каждое значение вычисляется до тех пор, пока его изменение не станет достаточно мало.
Отличие от метода Адамса состоит в том, что в методе Милна используется полином Ньютона, а в методе Адамса — полином Лагранжа.