БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Факультет прикладной математики и информатики
Кафедра вычислительной математики

Лабораторная работа № 3
Методы решения задачи Коши
Вариант 1б

Выполнил:
Кражевский Алексей Игоревич,
3 курс 13 группа (ТП)
Преподаватель:
Будник Анатолий Михайлович

Минск, 2022

Часть 1. Решение задачи Коши

Постановка задачи

Дана задача Коши вида $u'(x)=f(x,u(x)), x\in[x_0,X],$
$u(x_0)=u_0$

где $[x_0,X]=[\alpha_j,1+\alpha_j], f(x,u)=\frac{u}{x}+x\tilde{f}'(x), u_0=\alpha_j\tilde{f}(\alpha_j)$. Здесь $\alpha_j,\tilde{f}(x)$ берем из задания 1.

Необходимо найти приближенное решение задачи Коши на сетке узлов при 10-ти разбиениях отрезка интегрирования, применяя методы 1-го и 2-го порядка точности по варианту.

Данные согласно варианта 1б:

- $\alpha_j=0.1+0.05\cdot j, j=14.$ Из лаб. 1 получаем $\alpha_j=0.8$
- $[x_0,X]=[\alpha_j,1+\alpha_j].$ Получаем $x_0=0.8,[x_0,X]=[0.8,1.8]$
- $f(x)=\alpha_j e^x+(1-\alpha_j)\sin(x).$ Получаем $\tilde{f}(x)=0.8e^x+0.2\sin(x)$
- $\tilde{f}'(x)=0.8e^x+0.2\cos(x)$
- $u_0=0.8\cdot\tilde{f}(0.8)=1.5358192677307$
- $f(x,u(x))=\frac{u}{x}+x\cdot(0.8e^x+0.2\cos(x))$
- Полученная задача Коши
$$u'(x)=\frac{u}{x}+x\cdot(0.8e^x+0.2\cos(x)), x\in[0.8,1.8]$$
$$u(0.8)=1.5358192677307$$
- Согласно варианта, метод 1-го порядка — явный метод Эйлера, метод 2-го порядка — метод последовательного повышения порядка точности, $\alpha_0=0, \alpha_1=1.$

Алгоритм решения

Формула явного метода Эйлера:
$$y_{n+1}=y_n+h\cdot f_n$$
$$y_0=u_0$$
где $f_n=f(x_n,y_n)$.

Метод последовательного повышения порядка точности:
$$y_{n+1}=y_n+h\sum_{i=0}^{q}A_i f(x_n+\alpha_i h; y(x_n+\alpha_i h))$$

Параметры A_i,α_i неизвестны (по условию заданы $\alpha_0=0,\alpha_1=1$). Для их нахождения используются условия:
$$\sum_{i=0}^{q}A_i=1,$$
$$\sum_{i=0}^{q}A_i\alpha_i^j=\frac{1}{j+1}, j=1,\cdots,k-1$$

Для определения $y_{n+\alpha_i}$ используем формулу $y_{n+\alpha_i}=y_n+\alpha_i h\sum_{j=0}^{q_i} B_j f_{n+\alpha_i\beta_j}, q_i\le q$.

Значение $y_{n+\alpha_i\beta_j}$ определяем с порядком $k-1$.

Аналогично нахождения A_i, α_i получаем:

$$\sum_{i=0}^{q_i} B_j\beta_i^j=\frac{1}{j+1},$$

$$j=\overline{0,k-2}$$

Для метода второго порядка точности получаем:

$$\sum_{i=0}^{q} A_i\alpha_i=\frac{1}{2}$$

Если взять $q=0$, то получаем единственное решение $A_0=1, \alpha_0=\frac{1}{2}$, что не удовлетворяет постановке задачи ($\alpha_0=0, \alpha_1=1$).

Возьмем $q=1$. Тогда система примет вид:

$$A_0+A_1=1,$$

$$A_0\alpha_0+A_1\alpha_1=\frac{1}{2}$$

Тогда получаем следующие значения:

$$\alpha_0=0, \alpha_1=1, A_0=A_1=\frac{1}{2}$$

Получаем следующий метод (неявный метод трапеций):

$$y_{n+1}=y_n+\frac{h}{2}(f_n+f_{n+1})$$

Найдем B_j, β_i. Получаем следующие значения: $B_j=1; \beta_i=0$.

Применим найденные значения и составим итоговую систему (явный метод трапеций):

$$y_{n+1}=y_n+hf_n,$$

$$y_{n+1}=y_n+\frac{h}{2}(f_n+f_{n+1})$$

Код решения

Константные значения:

```
x0 = 0.8
X = 1.8
u0 = 1.5391231687791029087912519577858707831051714
alpha0 = 0
alpha1 = 1
A1 = 0.5
A2 = 0.5
```

Производная функции из лаб. 1:

```
def der_func(x):
    return 0.8 * np.exp(x) + 0.2 * np.cos(x)
```

Функция для задачи Коши:

```
def func(x, u):
    return (u / x) + (x * der_func(x))
```

Метод Эйлера:

```
def euler(x: np.ndarray) -> np.ndarray:
    y = np.ndarray([11])
    y[0] = u0
    h = 0.1
    for i in range(1, 11):
        y[i] = y[i-1] + h * func(x[i-1], y[i-1])
    return y
```

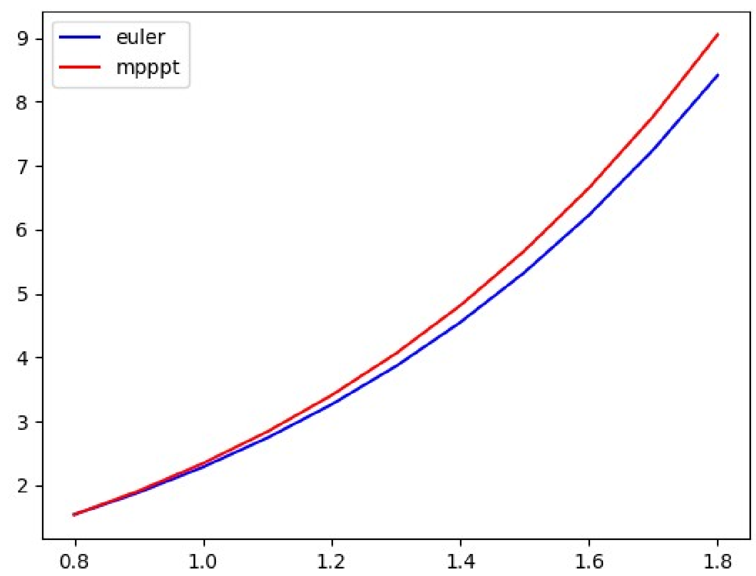
Метод последовательного повышения порядка точности:

```
def mpppt(x: np.ndarray) -> np.ndarray:
    y = np.ndarray([11])
    y[0] = u0
    for i in range(1, 11):
        y1 = y[i-1] + (0.1 * func(x[i-1], y[i-1]))
        y[i] = y[i-1] + (0.05 * (func(x[i-1], y[i-1]) + func(x[i], y1)))
    return y
```

Вывод программы

```
Points: [0.8 0.9 1.  1.1 1.2 1.3 1.4 1.5 1.6 1.7 1.8]
Values by Euler: [1.53912317 1.88509549 2.28283095 2.73938264 3.26276315 3.86208789
 4.54773339 5.33151297 6.22687197 7.24910524 8.41559932]
Values by MPPPT: [1.53912317 1.91097706 2.34099636 2.836779   3.40698584 4.06148952
 4.81154065 5.66995332 6.65131214 7.77220332 9.05147247]
```

График для метода Эйлера и для мпппт:



Часть 2. Подсчет погрешностей

Постановка задачи

Используя таблицу результатов, найти погрешности методов, сравнивая приблизительное значение с точным.

Алгоритм решения

Найдем аналитическое решение задачи Коши:

$$y = x\left(0.8e^x + 0.2\sin(x) + C\right)$$

После подстановки в полученное уравнение исходного условия, получаем $C = -2.77556 \cdot 10^{-16} \approx 0$.

В итоге получаем:

$$y = x\left(0.8e^x + 0.2\sin(x)\right)$$

Код решения

Точное решение:

```
def real_solution(x):
    return x * (0.8 * np.exp(x) + 0.2 * np.sin(x) - 2.77556 * 10e-16)
```

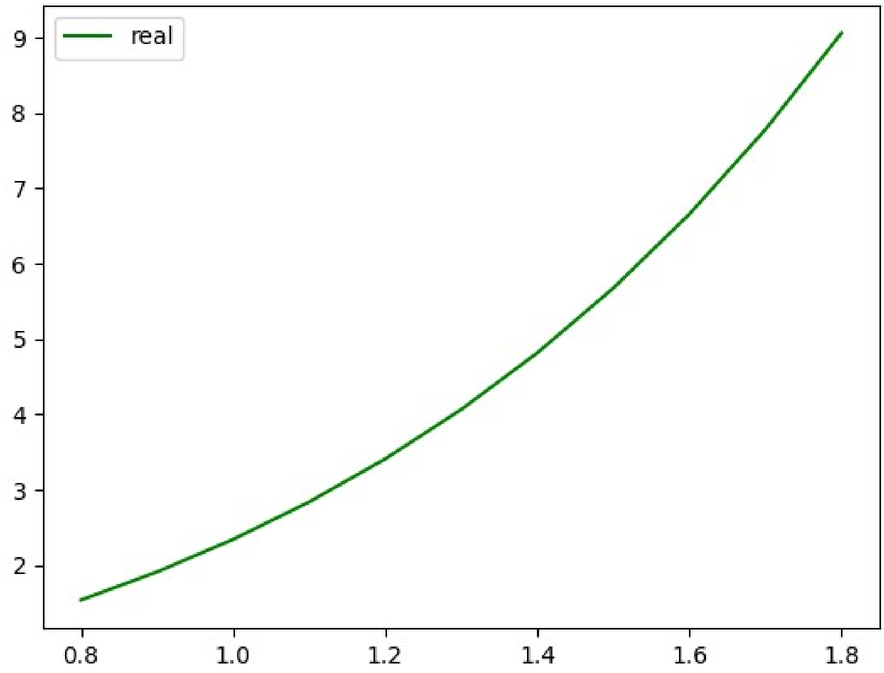
Подсчет погрешности:

```
error1 = []
error2 = []
for i in range(11):
    error1.append(abs(y1[i] - list_y_real[i]))
    error2.append(abs(y2[i] - list_y_real[i]))

print('Real errors using Euler:')
for i in error1:
    print('%.5f' % i, end=' ')
print()

print('Real errors using MPPPT:')
for i in error2:
    print('%.5f' % i, end=' ')
print()
```

График для аналитического решения:



Вывод программы

```
Real errors using Euler:
0.00000 0.02682 0.06009 0.10035 0.14824 0.20451 0.27002 0.34576 0.43287 0.53263 0.64648
Real errors using MPPPT:
0.00000 0.00094 0.00192 0.00295 0.00402 0.00510 0.00621 0.00732 0.00843 0.00953 0.01061
```

Часть 3. Сравнение методов

Постановка задачи

Исходя из вида главного члена локальной погрешности, а также оценивая величину истинной погрешности, сделать вывод и точности каждого используемого метода.

Алгоритм решения

Локальная погрешность явного метода Эйлера имеет вид

$$\Psi_{n+1}=u_{n+1}-u_n-hf_n=u_n+hu_n'+\frac{h^2}{2}u_n''+O(h^3)-u_n-hf_n=O(h^2)$$

Локальная погрешность метода последовательного повышения порядка точности:

$$\Psi_{n+1}=h^{k+1}u_n^{(k+1)}\left(\frac{1}{(k+1)!}-\frac{1}{k!}\sum_{i=0}^{q}A_i\alpha_i^k\right)+O(h^{k+2})=O(h^{k+1})$$

Вывод

В данной лабораторной работе мы рассмотрели решение задачи Коши явным методом Эйлера и методом второго порядка точности (мпппт). При оценке погрешности выяснилось, что погрешность метода пппт меньше, чем погрешность метода Эйлера. Это объясняется разницей в порядке точности методов, у Эйлера 1й порядок точности, у мпппт — второй (порядок точности метода вычисляется из локальной погрешности методов (порядок точности равен $k-1$, где k - степень h в $O(h^k)$)). Чем выше порядок точности у метода, тем меньше погрешность и точнее результаты вычисления.

По видам локальных погрешностей и значений истинных погрешностей в точках, можно сделать вывод, что метод последовательного повышения порядка точности решает задачу Коши точнее, чем метод Эйлера.