Жеские ОДУ
Юля Прохорова, Б01-907

March 8, 2022

0.0.1 Решение модельного уравнения

Необходимо решить модельное уравнение:

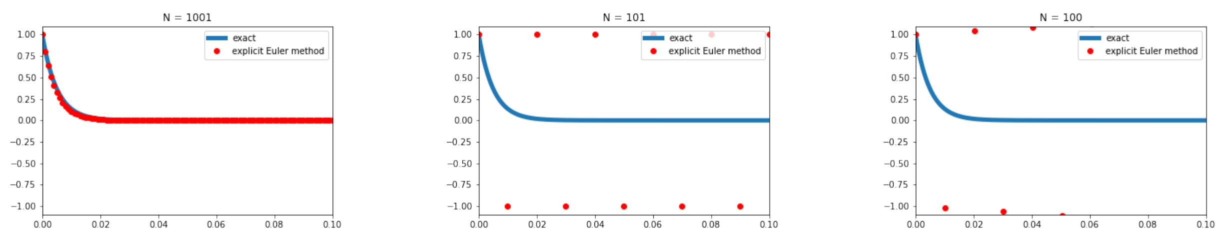
$$u'(t) = \lambda u(t) \tag{1}$$

с начальным условием $u(0) = 1$ с помошью явного и неявного метода Эйлера. Считать, что $\lambda = -200$. Провести расчеты на сетке с числом узлов $N = 1001; 101; 100$. Построить графики решения на одном рисунке, объяснить результат. А именно объяснить, что мы видим на графике для каждого случая, какое это имеет отношение к А-устойчивости, почему именно так. Если поведение графиков отличается, объяснить чем вызвано отличие.

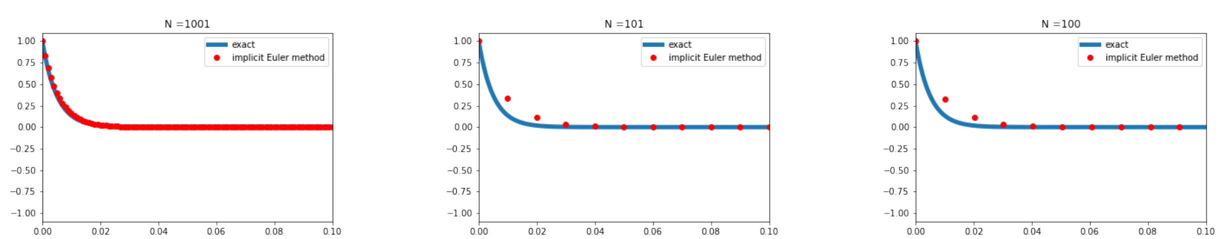
[15]:
```python
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline

#для модельного уравнения
#explicit Euler method
def ex_euler(u, l, h, N):
    for i in range(0, N-1):
        u[i+1] = (1+l*h)*u[i]

#implicit  Euler method
def im_euler(u, l, h, N):
    for i in range(0, N-1):
        u[i+1]=u[i]/(1-l*h)

T = 1 # считае до этого момента
N = 101
h = T/(N-1)

x = np.linspace(0,T,N) # сетка
x_teor = np.linspace(0,T,(N-1)*100+1) # сетка для точного решения
l = -200

u = np.zeros(N)
```

```python
u[0] = 1 #начальное условие
ex_euler(u, l, h, N)
# im_euler(u, l, h, N)

plt.figure(1)
plt.title('N = 101')
plt.plot(x_teor, np.exp(l*x_teor), linewidth=5.0, label='exact')
plt.plot(x, u, 'ro', label='explicit Euler method')
plt.legend()
plt.axis([0, 0.1, -1.1, 1.1])
plt.savefig(f'explicit_{N}')

# plt.figure(1)
# plt.title('N = 1001')
# plt.plot(x_teor, np.exp(l*x_teor), linewidth=5.0, label='exact')
# plt.plot(x, u, 'ro', label='implicit Euler method')
# plt.legend()
# plt.axis([0, 0.1, -1.1, 1.1])
# plt.savefig(f'implicit_{N}')
```

Явный метод Эйлера:



Неявный метод Эйлера:



Объяснение эффекта:

Область устойчивости явного метода Эйлера задаётся выражением $|1 + z| \leq 1$. $|Re(z)| \leq 0$ не входит в его область устойчивости, соответственно явный метод Эйлера не является A-устойчивым. Из области устойчивости получаем ограничение на шаг $h \leq \frac{2}{|\lambda|} = 10^{-2}$.
Теория согласуется с практикой:

- при N=1001 условие выполнено и метод сходится к точному решению
- при N=101 условие выполняется по границе, поэтому метод ни сходится, ни расходится
- при N=100 условие не выполнено и метод расходится

Область устойчивости неявного метода Эйлера задаётся выражением $|z-1| \geq 1$. $|Re(z)| \leq 0$ входит в его область устойчивости, а значит неявный метод Эйлера А-устойчив, поэтому он сходится к точному решению не зависимо от шага сетки, что и наблюдается на графиках.

0.0.2 L-устойчивость

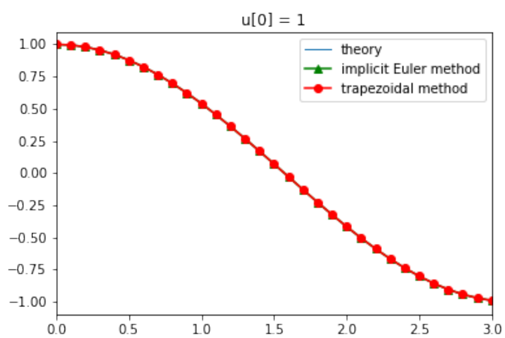
Рассматриваем задачу $u' = l(u - cos(t)) - sin(t)$. Для начального условия $u(0) = 1$ решение $u(t) = cos(t)$. Для начального условия $u(0) = g$ решение $u(t) = e^{lt}(g-1) + cos(t)$. Решите задачу на сетке с шагом $h = 0.1$ для начального условия $u(0) = 1$ и $u(0) = 1.5$ методом Эйлера и методом трапеции. В качестве решения постройте две картинки для разных начальных условий. На каждой картинке должно быть два графика для каждого из методов и сравнение с точным решением. Объясните получившиеся результаты. Как они согласуются с теорией? При чем тут L-устойчивость?

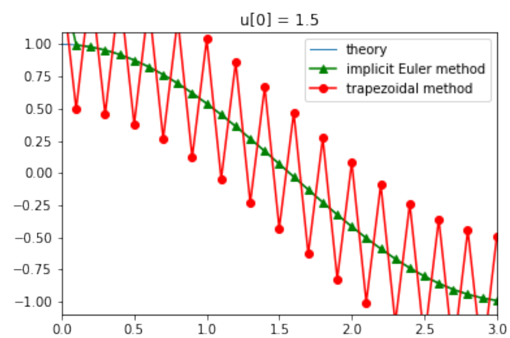
Метод трапеции

$$\frac{u_{n+1} - u_n}{h} = \frac{f(t_n, u_n) + f(t_{n+1}, u_{n+1})}{2}.$$

[28]:
```
%matplotlib inline
import numpy as np
import matplotlib.pyplot as plt

#implicit  Euler method
def im_euler(u, l, h, N):
    for i in range(0, N-1):
        u[i+1]=( u[i] - h*l*np.cos((i+1)*h) - h*np.sin((i+1)*h) )/(1-h*l)



#trapezoidal method
def trapezoidal(u, l, h, N):
    for i in range(0, N-1):
        free = l*(np.cos((i+1)*h)+np.cos(i*h))+(np.sin((i+1)*h)+np.sin(i*h))
        u[i+1] = (u[i]*(1+l*h/2)-h/2*free)/(1-l*h/2)

T = 3
N = 31
h = T/(N-1)
x_teor = np.linspace(0,T,(N-1)*100+1)
x = np.linspace(0,T,N)
l = -1e6

u_euler = np.zeros(N)
u_trap = np.zeros(N)

#initial condition
```

```
# u_euler[0] = 1
# u_trap[0] = 1

u_euler[0] = 1.5
u_trap[0] = 1.5


#method
im_euler(u_euler, l, h, N)
trapezoidal(u_trap, l, h, N)

plt.figure(1)
plt.title('u[0] = 1.5')
plt.plot(x_teor, np.cos(x_teor), linewidth=1.0, label='theory')
plt.plot(x, u_euler, '-g^', label='implicit Euler method')
plt.plot(x, u_trap, '-ro', label='trapezoidal method')
plt.legend()
plt.axis([0, 3, -1.1, 1.1])
# plt.savefig('graph_15')
```





Объяснение эффекта:

L-устойчивость выполняется, если метод A-устойчив и $lim_{z\to\infty}R(z) = 0$. В случае неявного метода Эйлера - L-устойчивость выполняется, а вот метод трапеции уже не L-устойчив $lim_{z\to\infty}R(z) = lim_{z\to\infty}(\frac{z+2}{2-z}) \neq 0$ (хотя и является A-устойчивым). Когда |R(z)| близок к единице затухание происходит медленно, а в случае как у метода трапеции |R(z)|=1 затухание не наблюдается. Значение начального условия $u[0] = 0$ - совпало с теоретическим, поэтому осцилляций нет в отличие от случая, когда $u[0] = 1.5$.