10. Obične diferencijalne jednačine, problem graničnih uslova

1. Rešavanje ODJ 2. reda

Metoda konačnih razlika

ODJ 2. reda je definisana kao:

$$\begin{aligned} ODJ_{leva}\big(x, f(x), f'(x), f''(x)\big) &= ODJ_{desna}(x) \\ m_a(x) f''(x) + m_b(x) f'(x) + m_c(x) f(x) &= ODJ_{desna}(x) \end{aligned} \tag{1}$$

, gde je ODJ_{leva} leva strana diferencijalne jednačine u kojoj figurišu svi izvodi funkcije, čiji poznati množioci $m(x)$ mogu da zavise od nezavisno promenljive x, a mogu biti i konstante, a ODJ_{desna} je desna strana diferencijalne jednačine u kojoj figuriše nezavisno promenljiva x i slobodne konstante.

Ako se funkcija nad intervalom $x \in [a, b]$ izdeli na n podintervala širine h, tada je:

$$h = \frac{b-a}{n}$$

Za poznatu razliku h izvodi funkcije u tački x se mogu aproksimirati konačnim razlikama:

$$\begin{aligned} \frac{d^2 f(x)}{dx^2} &= \frac{f(x-h) - 2f(x) + f(x+h)}{h^2} \\ \frac{df(x)}{dx} &= \frac{-f(x) + f(x+h)}{h} \\ f(x) &= f(x) \end{aligned}$$

Iterativni zapis:

$$\begin{aligned} f''(x_i) &= \frac{f(x_{i-1}) - 2f(x_i) + f(x_{i+1})}{h^2} \\ f'(x_i) &= \frac{-f(x_i) + f(x_{i+1})}{h} \\ f(x_i) &= f(x_i) \end{aligned}$$

Vektorski zapis:

$$\begin{aligned} f''(x_i) &= \frac{1}{h^2}\begin{bmatrix} 1 & -2 & 1 \end{bmatrix}\begin{bmatrix} f_{i-1} \\ f_i \\ f_{i+1} \end{bmatrix} \\ f'(x_i) &= \frac{1}{h}\begin{bmatrix} 0 & -1 & 1 \end{bmatrix}\begin{bmatrix} f_{i-1} \\ f_i \\ f_{i+1} \end{bmatrix} \\ f(x_i) &= \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} f_{i-1} \\ f_i \\ f_{i+1} \end{bmatrix} \end{aligned} \tag{2}$$

Zamenom (2) u (1) dobija se:

$$m_a(x_i)\frac{1}{h^2}\begin{bmatrix} 1 & -2 & 1 \end{bmatrix}\begin{bmatrix} f_{i-1} \\ f_i \\ f_{i+1} \end{bmatrix} + m_b(x_i)\frac{1}{h}\begin{bmatrix} 0 & -1 & 1 \end{bmatrix}\begin{bmatrix} f_{i-1} \\ f_i \\ f_{i+1} \end{bmatrix} + m_c(x_i)\begin{bmatrix} 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} f_{i-1} \\ f_i \\ f_{i+1} \end{bmatrix} = ODJ_{desna}(x_i)$$

Ako su $m_a(x_i), m_b(x_i)$ i $m_c(x_i)$ poznate funkcije i ako su x_i i h poznati, prethodni izraz se za svaku tačku $x_i, i \in [1, n-1]$ može zapisati na sledeći način:

$$\begin{bmatrix} m_0 & m_1 & m_2 \end{bmatrix} \begin{bmatrix} f_0 \\ f_1 \\ f_2 \end{bmatrix} = ODJ_{desna}(x_1)$$
$$\begin{bmatrix} m_3 & m_4 & m_5 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} = ODJ_{desna}(x_2)$$
$$\vdots$$
$$\begin{bmatrix} m_{3n-6} & m_{3n-5} & m_{3n-4} \end{bmatrix} \begin{bmatrix} f_{n-2} \\ f_{n-1} \\ f_n \end{bmatrix} = ODJ_{desna}(x_{n-1})$$

, gde su množioci $m_0, m_1, ..., m_{3n-4}$ izračunati na osnovu poznatih vrednosti.
Matrični zapis prethodnog sistema jednačina je:

$$\begin{bmatrix} m_0 & m_1 & m_2 & 0 & 0 & & & & & & \\ 0 & m_3 & m_4 & m_5 & 0 & \cdots & & & & & \\ 0 & 0 & m_6 & m_7 & m_8 & & & & & & \\ & & \vdots & & & \ddots & & & \vdots & & \\ & & & & & & m_{3n-12} & m_{3n-11} & m_{3n-10} & 0 & 0 \\ & & & & & \cdots & 0 & m_{3n-9} & m_{3n-8} & m_{3n-7} & 0 \\ & & & & & & 0 & 0 & m_{3n-6} & m_{3n-5} & m_{3n-4} \end{bmatrix} \begin{bmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_{n-3} \\ f_{n-2} \\ f_{n-1} \\ f_n \end{bmatrix} = \begin{bmatrix} ODJ_{desna}(x_1) \\ ODJ_{desna}(x_2) \\ ODJ_{desna}(x_3) \\ \vdots \\ ODJ_{desna}(x_{n-3}) \\ ODJ_{desna}(x_{n-2}) \\ ODJ_{desna}(x_{n-1}) \end{bmatrix}$$

Prehodni sistem ima $(n+1)$ kolona, a $(n-1)$ vrsta.

Vrednost funkcije je poznata u krajevima intervala (granični uslovi):

$$f_0 = f(a) = f_a$$
$$f_n = f(b) = f_b$$

Prebacivanjem poznatih vrednosti na desnu stranu prve i poslednje jednačine, suvišne kolone se mogu eliminisati:

$$\begin{bmatrix} m_1 & m_2 & 0 & 0 & & & & & \\ m_3 & m_4 & m_5 & 0 & \cdots & & & & \\ 0 & m_6 & m_7 & m_8 & & & & & \\ & \vdots & & & \ddots & & \vdots & & \\ & & & & & m_{3n-12} & m_{3n-11} & m_{3n-10} & 0 \\ & & & & \cdots & 0 & m_{3n-9} & m_{3n-8} & m_{3n-7} \\ & & & & & 0 & 0 & m_{3n-6} & m_{3n-5} \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_{n-3} \\ f_{n-2} \\ f_{n-1} \end{bmatrix} = \begin{bmatrix} ODJ_{desna}(x_1) - m_0 f_a \\ ODJ_{desna}(x_2) \\ ODJ_{desna}(x_3) \\ \vdots \\ ODJ_{desna}(x_{n-3}) \\ ODJ_{desna}(x_{n-2}) \\ ODJ_{desna}(x_{n-1}) - m_{3n-4} f_b \end{bmatrix} \quad (3)$$

Rešenje sistema je vektor:

$$f = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_{n-1} \end{bmatrix}$$

Ako se u vektor uključe poznate vrednosti vrednosti:

$$f = \begin{bmatrix} f_a \\ f_1 \\ f_2 \\ \vdots \\ f_{n-1} \\ f_b \end{bmatrix}$$

Vektor predstavlja vrednosti funkcije u tačkama $a + ih, i \in [0, n]$, tj. numeričko rešenje ODJ sa problemom graničnih uslova (PGU).

Zadatak 1. Napisati metodu konačnih razlika za rešavanje diferencijalne jednačine 2. reda sa PGU nad proizvoljnim intervalom.

1. Definisati 3 funkcije koje vraćaju 3-člane vektore množioca uz f_{i-1}, f_i i f_{i+1} iz jednačina (2):

- za konačnu razliku 2. izvoda:

```
def ddf(h):
      arr = np.array([1, -2, 1])
      fX = arr/h**2
      return fX
```

- za konačnu razliku 1. izvoda:

```
def ddf(h):
      arr = np.array([0, -1, 1])
      fX = arr/h
      return fX
```

- za konačnu razliku 0. izvoda (same funkcije):

```
def ddf(h):
      arr = np.array([0, 1, 0])
      fX = arr
      return fX
```

Pokušati prvo ručno jednu iteraciju metode konačnih razlika nad funkcijom na intervalu $x \in [0,2\pi]$ sa 10: podintervala za sledeći PGU:

$$f''(x) - \sin x\, f'(x) + 2f(x) = 1 - x$$

$$f(0) = 0$$
$$f(2\pi) = 0$$

Definisati funkciju koja odgovara levoj strani diferencijalne jednačine oslanjajući se na funkcije definisane u koraku 1:

```
left = lambda *args:  ddf(args[1]) – sin(args[0]) * df(args[1]) + 2* f(args[1])
```

Definisati funkciju koja odgovara desnoj strani diferencijalne jednačine:

```
right = lambda x:  1 - x
```

Definisati granične uslove i izračunati korak h:

```
x0 = 0
fX0 = 0
xN = 2*np.pi
fXN = 0
h = (xN - x0)/10
```

2. Napraviti vektor vrednosti nezavisno promenljive x za dati korak h:

```
x = np.arange(x0, xN, h)
```

3. Izračunati dimenziju dim matrice A i vektora b sistema jednačina (3), a zatim ih inicijalizovati:

```
dim = len(x) - 2
A = np.zeros([n, n])
b = np.zeros([n,1])
```

Izračunati levu i desnu stranu diferencijalne jednačine za x_1:

```
m = left(x(1), h)
b[it] = right(x(1))
```

Rezultat:

```
m = [2.5330   -2.1306    1.5975]

b =
    [[0.3717]
           [0]
           [0]
           [0]
           [0]
           [0]
           [0]
           [0]
           [0]]
```

4. Popuniti 3-dijagonalnu matricu A i vektor b sistema jednačina (3), koristeći izračunate vrednosti leve i desne strane diferencijalne jednačine u novoj tački x za svaku sledeću vrstu:

```
A = np.zeros([dim, dim], dtype = np.float64)
b = np.zeros([dim, 1])
for it in range(dim):
    m = left(x[it +1], h)
    mid = np.round(len(m)/2)-1
    fromA = max(0, it - mid)
    toA = min(dim, it + mid+1)
    fromM = mid - it  + fromA
    toM = mid - it + toA
    A[it, int(fromA):int(toA)] = m[int(fromM):int(toM)]
    b[it ] = right(x[it +1])
```

5. Oduzeti poznate vrednosti $fX0$, odnosno fXn, pomnožene odgovarajućim množiocima od prvog, odnosno poslednjeg elementa vektora b:

```
mA = left(x[1], h)
b[0] = b[0] – mA[int(mid)-1]*fX0
mB = = left(x[-2], h)
b[-1] = b[-1] – mB[int(mid)+1]*fXN
```

Rezultat:

```
 A =                                                                                          b =
    -2.1306    1.5975         0         0         0         0         0         0         0        0.3717
     2.5330   -1.5524    1.0194         0         0         0         0         0         0       -0.2566
          0    2.5330   -1.5524    1.0194         0         0         0         0         0       -0.8850
          0         0    2.5330   -2.1306    1.5975         0         0         0         0       -1.5133
          0         0         0    2.5330   -3.0661    2.5330         0         0         0       -2.1416
          0         0         0         0    2.5330   -4.0015    3.4685         0         0       -2.7699
          0         0         0         0         0    2.5330   -4.5797    4.0467         0       -3.3982
          0         0         0         0         0         0    2.5330   -4.5797    4.0467       -4.0265
          0         0         0         0         0         0         0    2.5330   -4.0015       -4.6549
```

6. Rešiti sistem jednačina i uklopiti rešenje sa poznatim vrednostima funkcije u krajevima intervala:

```
fX = np.linalg.solve(A,b)
fxx = [fX0]
for i in range(len(fX)):
    fxx.append(fX[i][0])
fxx.append(fXN)
```

Rezultat:

```
fX =
     0    3.7703    5.2610   -1.6086  -16.3908  -20.2564   -8.9736    3.6419    8.8988    6.7963         0
```

7. Sada je moguće definisati funkciju koja sadrži prethodni algoritam:

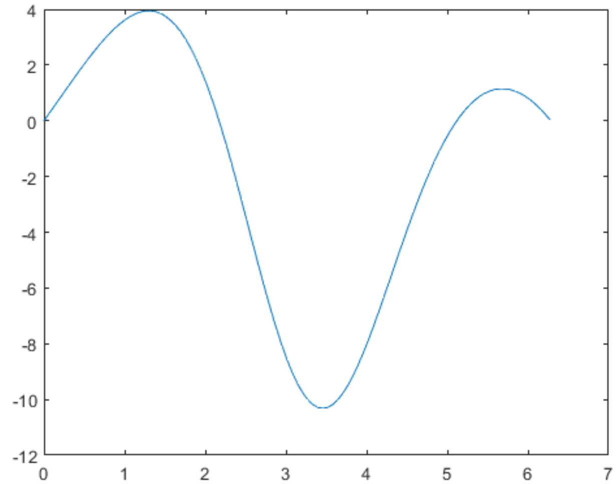
```
def finiteDifference(left, right, x0, fX0, xN, fXN, h):
     x = np.arange(x0,xN,h)
        .
        .
     fxx = [fX0]
     for i in range(len(fX)):
        fxx.append(fX[i][0])
     fxx.append(fXN)
```

8. Testirati funkciju na primeru:

```
left = lambda *args: ddf(args[1]) – np.sin(args[0]) * df(args[1]) + 2*f(args[1])
right = lambda x: 1 -x

x1,fX1, x2, fX2 = 0, 0, 2*n.pi, 0
h = (x2- x1)/ 100
x = np.arange(x1, x2, h)
fX = finiteDifference(left, right, x1, fX1, x2, fX2, h)
plt.plot(x, fX)
plt.show()
```

Rezultat:



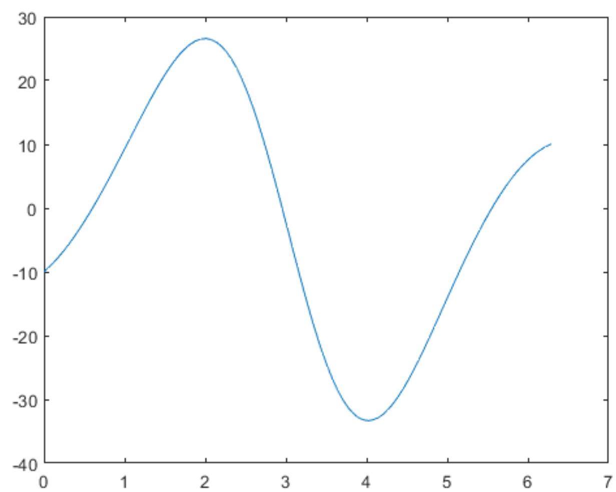
1. Grafik rešenja PGU

9. Pomeriti granične uslove:

```
left = lambda *args: ddf(args[1]) – np.sin(args[0]) * df(args[1]) + 2*f(args[1])
right = lambda x: 1 -x

x1,fX1, x2, fX2 = 0, -10, 2*n.pi, 10
h = (x2- x1)/ 100
x = np.arange(x1, x2, h)
fX = finiteDifference(left, right, x1, fX1, x2, fX2, h)
plt.plot(x, fX)
plt.show()
```

Rezultat:



2. Drugačije rešenje usled novih graničnih uslova