

## 2. DOMAČA NALOGA

Katarina Černe, 27172036

21. maj 2019

### 1. naloga

Implementacija 1. naloge se nahaja v datotekah `EulerImplicitna.m`, `EulerIzboljsana.m` in `Heunova.m`, rezultati pa se preverijo v datoteki `preverjanje1.m`. Implementiramo tri različne Eulerjeve metode za reševanje robnega problema oblike  $\frac{dy}{dx} = fun(x, y)$ , kjer je  $x \in [a, b]$ , podan imamo še  $y(a)$  in korak  $h$ .

V datoteki `EulerImplicitna.m` implementiramo implicitno Eulerjevo metodo. Interval  $[a, b]$  razdelimo na  $n = \frac{b-a}{h}$  delov z delilnimi točkami  $\{x_i\}_{i=0}^n$ ,  $x_i = a + i \cdot h$ . Približke  $y_i \approx y(x_i)$  izračunamo na naslednji način:

$$y_i = y_{i-1} + h \cdot fun(x_i, y_i).$$

Formula je implicitna, zato moramo  $y_i$  računati s pomočjo navadne iteracije, ali pa za reševanje uporabimo kar vgrajeno funkcijo `fzero`, pri čemer računamo ničle funkcije

$$g(t) = t - y_{i-1} - h \cdot fun(x_i, t)$$

Pri tem rabimo še začetni približek za  $y_i$ , ki ga izračunamo s pomočjo eksplicitne Eulerjeve metode:

$$y_i = y_{i-1} + h \cdot fun(x_{i-1}, y_{i-1}).$$

V datoteki `EulerIzboljsana.m` implementiramo izboljšano Eulerjevo metodo. Interval  $[a, b]$  razdelimo na  $n = \frac{b-a}{2h}$  delov z delilnimi točkami  $\{x_{\frac{i}{2}}\}_{i=0}^n$ ,  $x_{\frac{i}{2}} = a + \frac{i}{2}h$ . Približke  $y_i \approx y(x_i)$  izračunamo na naslednji način:

$$y_{i-\frac{1}{2}} = y_{i-1} + \frac{h}{2} fun(x_{i-1}, y_{i-1}), \text{ kjer } i = 1, 2, \dots \text{ in}$$

$$y_i = y_{i-1} + h \cdot fun(x_{i-\frac{1}{2}}, y_{i-\frac{1}{2}}), \text{ kjer } i = 1, 2, \dots$$

V datoteki `Heunova.m` implementiramo Heunovo metodo. Interval  $[a, b]$  razdelimo na  $n = \frac{b-a}{h}$  delov z delilnimi točkami  $\{x_i\}_{i=0}^n$ ,  $x_i = a + i \cdot h$ . Butcherjeva shema za Heunovo metodo je oblike

$$\begin{array}{c|cc} 0 & 0 & \\ 1 & 1 & 0 \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

Približke  $y_i \approx y(x_i)$  izračunamo na naslednji način:

$$k_1 = fun(x_{i-1}, y_{i-1})$$

$$k_2 = fun(x_{i-1} + \frac{2}{3}h, y_{i-1} + \frac{2}{3}h \cdot k_1)$$

$$y_i = y_{i-1} + h(\frac{1}{4}k_1 + \frac{3}{4}k_2).$$

Dobimo naslednje rezultate: Število okuženih po 10 dneh:

- z implicitno Eulerjevo metodo:  $0.050172760083025 \cdot 10^4$
- z izboljšano Eulerjevo metodo:  $0.049524536918739 \cdot 10^4$
- s Heunovo metodo:  $0.049524536867780 \cdot 10^4$

Število okuženih po 20 dneh:

- z implicitno Eulerjevo metodo:  $0.251678987879975 \cdot 10^4$
- z izboljšano Eulerjevo metodo:  $0.245220099105067 \cdot 10^4$
- s Heunovo metodo:  $0.245220097603792 \cdot 10^4$

Število okuženih po 30 dneh:

- z implicitno Eulerjevo metodo:  $1.261187470506422 \cdot 10^4$
- z izboljšano Eulerjevo metodo:  $1.213031743398884 \cdot 10^4$
- s Heunovo metodo:  $1.213031705461628 \cdot 10^4$

## 2. naloga

Implementacija 2. naloge se nahaja v datotekah `BDF.m`, `RungeKutta4.m` in `EulerEksPLICITna.m`, rezultati pa se preverijo v datoteki `preverjanje2.m`.

V datoteki `BDF.m` implementiramo implicitno 4-člensko BDF metodo za robnega problema oblike  $\frac{dy}{dx} = fun(x, y)$ , kjer je  $x \in [a, b]$ , podan imamo še  $y(a)$  in korak  $h$ .

Približke  $y_n$  pri tej metodi računamo po formuli

$$\sum_{i=1}^4 \frac{1}{i} \nabla^i y_n = fun(x_n, y_n).$$

Tu se  $\nabla^i y_n$  izraža kot

$$\nabla^i y_n = \sum_{j=0}^i (-1)^j \binom{i}{j} y_{n-j},$$

torej lahko izrazimo

$$y_n = \frac{1}{\sum_{i=0}^4 \frac{1}{i}} (h \cdot fun(x_n, y_n) - \sum_{i=1}^4 \frac{1}{i} \sum_{j=1}^i (-1)^j \binom{i}{j} y_{n-j}).$$

Metoda je implicitna, zato  $y_n$  računamo s pomočjo funkcije `fzero`, kot pri 1. nalogi pri implicitni Eulerjevi metodi. Začetni približek za  $y_n$  izračunamo z Runge-Kutta metodo reda 4, nato pa računamo ničle funkcije

$$g(t) = t - \frac{1}{\sum_{i=0}^4 \frac{1}{i}} (h \cdot fun(x_n, t) - \sum_{i=1}^4 \frac{1}{i} \sum_{j=1}^i (-1)^j \binom{i}{j} y_{n-j}),$$

Pri implicitni 4-stopenjski BDF metodi moramo najprej določiti  $y_i$  za  $i = 0, 1, 2, 3$ . Poznamo že  $y_0$ ,  $y_1$ ,  $y_2$  in  $y_3$  pa določimo z Runge-Kutta metodo reda 4, ki ima naslednjo Butcherjevo shemo:

0			
$\frac{1}{2}$	$\frac{1}{2}$		
$\frac{1}{2}$	0	$\frac{1}{2}$	
1	0	0	1
$\frac{1}{6}$	$\frac{2}{6}$	$\frac{2}{6}$	$\frac{1}{6}$

To metodo implementiramo v datoteki `RungeKutta4.m`.

V datoteki `EulerEksPLICITna.m` implementiramo eksplcitno Eulerjevo metodo, pri kateri računamo

$$y_i = y_{i-1} + h \cdot fun(x_{i-1}, y_{i-1}).$$

Pri koraku  $h = \frac{1}{25}$  dobimo z BDF metodo  $y(b) = 2.0000000000000326$ , pri koraku  $h = \frac{1}{20}$  dobimo  $y(b) = 2.0000000000309457$ , pri  $h = \frac{1}{15}$  dobimo  $y(b) = 1.999999909530579$  in pri  $h = 1/10$  dobimo  $y(b) = 1.999997577067534$ . Pri vseh štirih  $h$ -jih se dobljene rešitve za večje  $x_i$  gibljejo okrog 2, kar se precej približa točni rešitvi, ki konvergira proti 2 in za katero velja  $y(b) = 2$ . Po drugi strani pa z eksplcitno Eulerjevo metodo dobimo  $y(b) = 2$  za  $h = \frac{1}{25}$  in  $h = \frac{1}{20}$  in  $y(b) = 2.0000000000000005$  za  $h = \frac{1}{15}$ . V teh treh primerih se rešitve za velike  $x_i$  gibajo okrog 2. Za  $h = \frac{1}{10}$  pa dobimo rešitev, ki oscilira med 3 in 1, za sode  $i$  je  $y_i = 3$ , za lihe  $i$  je  $y_i = 1$ . Rešitev se torej v tem primeru ne približa točni rešitvi, torej ta metoda za ta primer ni stabilna.

### 3. naloga

Implementacija 3. naloge se nahaja v datotekah `MilneSistem.m` in `RungeKutta4.m`. V datoteki `primer3.m` se nahaja sistem diferencialnih enačb, na katerega prevedemo dano diferencialno enačbo drugega reda, v datoteki `preverjanje3.m` pa preverimo rešitve.

Pri  $h = 0.1$  dobimo rešitev  $y(b) = -0.441424015188975$ , pri  $h = 0.05$  pa  $y(b) = -0.422277970368344$ .