Univerza v *Ljubljani*
Fakulteta za *pomorstvo in promet*

Numerične Metode v Tehniki

Diferencialne enačbe

Začetni problem

Aleksander GRM

aleksander.grm@fpp.uni-lj.si

Vsebina

$\boxed{\text{Poglavje – 1}}$

Uvod

Uvod

Vsebina tega poglavja se bo vrtela okoli numeričnega reševanja **začetnega problema**, oziroma

- **DE 1. reda** (samo ena enačba)

$$\frac{dy}{dx} = f(x, y), \quad x \in [x_0, x_e],$$

  ki ima začetni pogoj $y(x_0) = y_0$,

- **sistema DE 1. reda** (n enačb)

$$\begin{aligned} \frac{dy_1}{dx} &= f(x, y_1, y_2, \ldots, y_n), \\ \frac{dy_2}{dx} &= f(x, y_1, y_2, \ldots, y_n), \\ &\vdots \\ \frac{dy_n}{dx} &= f(x, y_1, y_2, \ldots, y_n), \end{aligned}$$

  za $x \in [x_0, x_e]$ in začetnim pogojem $y_1(x_0) = y_1^0, y_2(x_0) = y_2^0, \ldots, y_n(x_0) = y_n^0$.

Poglavje – 2

Navadne Diferencialne Enačbe 1. reda – enočlene metode

Eulerjeva metoda

Eulerjeva metoda

Eulerjeva metoda, je najbolj enostavna metoda za reševanje začetnega problema. Metoda aproksimira rešitev **dobro pogojenega** začetnega problema

$$\frac{dy}{dx} = f(x, y), \quad x \in [x_0, x_e],$$

z začetnim pogojem $y(x_0) = y_0$.

Dobljena rešitev ne bo zvezna, ampak bo izračunana na končnem številu točk (mrežnih točk) na intervalu $x_n \in [x_0, x_e]$.

Mrežne točke določimo

$$x_n = x_0 + n\,h, \quad n = \{1, 2, 3, 4, 5, \ldots, N\},$$

kjer je h **velikost koraka** in N **število korakov**.

Velikost koraka lahko določimo vnaprej

$$h = \frac{x_e - x_0}{N}.$$

Eulerjeva metoda

Razvijmo rešitev $y(x)$ v Taylorjevo vrsto do 2. odvoda, po mrežnih točkah

$$y(x_{n+1}) = y(x_n) + (x_{n+1} - x_n)y'(x_i) + \frac{(x_{n+1} - x_n)^2}{2} y''(\xi_n), \quad \xi_n \in (x_n, x_{n+1}).$$

Vpeljemo korak $h = x_{n+1} - x_n$ in in nadomestimo odvod $y'(x) = f(x, y)$, tako dobimo

$$y(x_{n+1}) = y(x_n) + h\, y'(x_n) + \frac{h^2}{2} y''(\xi_n), \quad \xi_n \in (x_n, x_{n+1}).$$

Eulerjeva metoda se zadovolji z natančnosto $\mathcal{O}(h^2)$, torej lahko zadnji člen izpustimo. Za vrednosti funkcije v mrežnih točkah pišemo skrajšano $y(x_n) = y_n$, tako dobimo

Eulrejevo explicitno shemo

$$y_{n+1} = y_n + h\, f(x_n, y_n) + \mathcal{O}(h^2), \quad n = \{0, 1, 2, 3, \ldots, N-1\},$$

kjer je y_0 začetni pogoj.

Eulerjeva metoda je numerična metoda **1. reda**, saj nastopa napaka v drugem redu ($\mathcal{O}(h^2)$).

Eulerjeva metoda

Enočlene numerične metode so takšne metode, ki v shemi vsebuje samo en prejšni člen (y_n) za določitev naslednjega (y_{n+1}).

$$y_{n+1} = g(x_n, y_n).$$

V funkciji odvoda Eulerjeve metode spremenimo indekse $n \rightarrow n+1$

$$y_{n+1} = y_n + h\, f(x_{n+1}, y_{n+1}) + \mathcal{O}(h^2), \quad n = \{0, 1, 2, 3, \ldots, N-1\}.$$

Tako dobimo **implicitno Eulerjevo metodo**, saj se neznanke y_{n+1} ne da izraziti v eksplicitni formi kakor v prejšnjem eksplicitnem primeru sheme.

V tem primeru je potrebno v vsakem koraku rešiti **nelinearno enačbo** (uporaba znanja iz prvega dela NMT).

Prednost implicitnih metod je v tem, da so zelo stabilne, kar pomeni bistveno bolj kakor eksplicitne. Vendar je zato potrebno plačati ceno reševanja nelinearne enačbe!

Eulerjeva metoda – primer

Imamo DE 1. reda

$$y' = f(x, y) = y - x^2 + 1,$$

kjer iščemo rešitev na intervalu $I_1 = [0, 2]$ in $I_2 = [0, 4]$.

Za začetni pogoj vzemimo $y_0 = 0.5$.

Analitična rešitev zgornjega problema je (uporabi *SageMath*®)

$$y(x) = x^2 + 2x - \frac{e^x}{2} + 1,$$

Naloga:

- Izračunaj rešitve v točkah ekvidistantne mreže, kjer imaš podano število intervalov $N = \{10, 20, 40, 80\}$,
- Nariši graf rešitev in primerjaj napako rešitve s točno rešitvijo za različno število intervalov,
- Implementiraj eksplicitno in implicitno varianto.

- Postopek rešitve implementiraj v *Python*® okolju z uporabo *NumPy*® knjižnice. Za reševanje nelinearne enačbe uporabi *SciPy*® modul scipy.optimize.fsolve

Metode višjega reda – Taylorjev razvoj

Razvoj v Taylorjevo vrsto

Enako kakor prej rešujemo **IVP 1. reda**

$$\begin{aligned} y' &= f(x,y), \quad x \in [x_0, x_e] \\ y(x_0) &= y_0. \end{aligned}$$

Privzamemo, da je rešitev $y = y(x)$ zvezna in obstaja n odvodov. Tako funkcijo lahko razvijemo v Taylorjevo vrsto

$$y(x+h) = y(x) + h\,y'(x) + \frac{h^2}{2}y''(x) + \frac{h^3}{6}y'''(x) + \cdots + \frac{h^n}{n!}y^{(n)}(x) + \frac{h^{n+1}}{(n+1)!}y^{(n+1)}(\xi), \quad \xi \in [x, x+h]$$

Ne enak način kakor prej lahko določimo rešitev v naslednji točki $x_{n+1} = x_n + h$ z znanimi vrednostmi odvodov v točki x_n.

Taylorjeva metoda k-tega reda je tako določena z

$$\boxed{y_{n+1} = y_n + hT^{(k)}(x_n, y_n)}, \quad T^{(k)}(x_n, y_n) = f(x_n, y_n) + \frac{h}{2}f'(x_n, y_n) + \cdots + \frac{h^{k-1}}{k!}f^{(k-1)}(x_n, y_n).$$

Razvoj v Taylorjevo vrsto

Pri razvoju v Taylorjevo vrsto, smo opazili, da je natančnost izračuna pogojena s številom členov v vrsti. Tako lahko ocenimo napako numerične metode.

Definicija – Lokalna napaka

Naj bo $x_{n+1} = x_n + h$. Napaka enočlene numerične metode za reševanje začetnega problema, je razlika

$$\boxed{\tau_h(x_{n+1}) := y(x_{n+1}) - y_{n+1}}$$

med točno vrednostjo $y(x_{n+1})$ in numerično dobljenim približkom y_{n+1}, ob predpostavki, da je prejšnji približek točen $y_n = y(x_n)$.

Posledica

Enočlenska metoda ima **red napake** k, če se pri točni vrednosti $y_n = y(x_n)$ izračunani približek y_{n+1} ujema z razvojem $y(x_n + h)$ v Taylorjevo vrsto okrog točke x_n do vključno člena h^k, kar je tako ekvivalentno

$$\tau_h(x_{n+1}) = \mathcal{O}(h^{k+1}).$$

Metode Runge – Kutta

Metode Runge – Kutta

Pri metodah z razvojem v Talorjevo vrsto, potrebujemo analitičen izraz za vse višje odvode, kar pripelje do nepraktične uporabe Taylorjeve metode.

Z uporabo Eulerjeve metode v vmesnih točkah intervala $[x_n, x_{n+1}]$ pridemo tako do še vedno **enočlene** metode, ki pa ne potrebuje višjih analitičnih odvodov, a hkrati zagotovimo višjo natančnost metode.

Takim metodam pravimo **Runge – Kutta** in temelji na določitvi vrednosti v vmesnih točkah

$$k_i = h\, f\Big(x_n + \alpha_i h, y_n + \sum_{j=1}^{m} \beta_{ij}\, k_j\Big), \quad i = 1, \ldots, m$$

V primeru, ko je koeficient $\beta_{ij} = 0$ za vse $i \leq j$, je metoda **eksplicitna**, v nasprotnem pa je **implicitna**.

Rešitev poiščemo v obliki

$$y_{n+1} = y_n + \sum_{i=1}^{m} \gamma_i\, k_i$$

Kar preostane, je poiskati vrednosti za koeficiente α_i, β_{ij}, γ_i v povezavi z napako metode.

Runge – Kutta: dvostopenjska eksplicitna

Iščemo dvostopenjsko RK metodo, v tem primeru je metoda 2. reda, kar pomeni, da je ostanek reda $\mathcal{O}(h^3)$?

Naloga

Določi proste koeficiente dvostopenjske RK metode (vsebuje le dva člena k_1 in k_2). Iščemo shemo oblike

$$\begin{aligned} k_1 &= hf(x_n, y_n) \\ k_2 &= hf(x_n + \alpha_1 h, y_n + \beta_{21} k_1) \\ y_{n+1} &= y_n + \gamma_1 k_1 + \gamma_2 k_2 \end{aligned}$$

Postopek rešitve

Metoda naj aproksimira Taylorjev člen 2. reda

$$T^{(2)}(x, y) = f(x, y) + \frac{h}{2} f'(x, y),$$

kjer zahtevamo, da lokalna napaka Taylorjeve metode ne presega $\mathcal{O}(h^2)$.

Runge – Kutta: dvostopenjska eksplicitna

Izpeljava temelji na razvoju Tajlorjeve vrste za funkcijo dveh spremenljivk $f(x,y)$!

1. Taylorjev razvoj funkcije $f(x, y(x))$ okoli točke (x_n, y_n) s korakom h

$$y(x_n + h) = y(x_n) + h\, f(x_n, y_n) + \frac{1}{2} h^2 \left(\frac{\partial f}{\partial x} + f \frac{\partial f}{\partial y} \right) (x_n, y_n) + \mathcal{O}(h^3)$$

2. Taylorjev razvoj funkcije y_{n+1} okoli točke (x_n, y_n) s korakom $\alpha_1 h$ v x smeri in $\beta_{21} k_1$ v y smeri

$$\begin{aligned}
k_1 &= hf, \quad \text{(ni razvoja, zakaj?)} \\
k_2 &= h \left(f + \alpha_1 h \frac{\partial f}{\partial x} + \beta_{21} k_1 \frac{\partial f}{\partial y} \right) + \mathcal{O}(h^3), \\
y_{n+1} &= y_n + \gamma_1 k_1 + \gamma_2 k_2 = \gamma_1 hf + \gamma_2 hf + \gamma_2 \alpha_1 h^2 \frac{\partial f}{\partial x} + \gamma_2 \beta_{21} h k_1 \frac{\partial f}{\partial y} + \mathcal{O}(h^3) \\
&= y_n + (\gamma_1 + \gamma_2) hf + h^2 \left(\gamma_2 \alpha_1 \frac{\partial f}{\partial x} + \gamma_2 \beta_{21} f \frac{\partial f}{\partial y} \right) + \mathcal{O}(h^3)
\end{aligned}$$

Runge – Kutta: dvostopenjska eksplicitna

Izenačimo člene z isto potenco h za Taylorjev razvoj iz 1. in razvoj iz 2. dobimo sistem enačb

$$\begin{aligned} \gamma_1 + \gamma_2 &= 1, \\ \gamma_2\,\alpha_1 &= \frac{1}{2}, \\ \gamma_2\,\beta_{21} &= \frac{1}{2}. \end{aligned}$$

Sistem ima tako poljubno število rešitev, saj je neznank več kakor enačb (izpeljemo lahko več različnih pravil).

V primeru, izboljšane Eulerjeve metode lahko privzamemo $\gamma_1 = 0$. Izpeljava pokaže, da so tako iskane neznanke

$$\gamma_2 = 1, \quad \alpha_1 = \frac{1}{2}, \quad \beta_{21} = \frac{1}{2}.$$

Modificirana Eulerjeva metoda je RK metode 2. reda

$$y_{i+1} = y_i + h\,f\left(x_i + \frac{h}{2}, y_i + \frac{h}{2}\,f(x_i, y_i)\right) + \mathcal{O}(h^3)$$

Drugače ji pravimo tudi **sredinska metoda** (zakaj?).

RK metoda 2. reda – primer

Imamo DE 1. reda

$$y' = y - x^2 + 1,$$

kjer iščemo rešitev na intervalu $I_1 = [0, 2]$ in $I_2 = [0, 4]$.
Privzamemo začetni pogoj $y_0 = 0.5$.

Izračunaj rešitve v točkah ekvidistantne mreže, kjer imaš podano število intervalov $N = \{10, 20, 40, 80\}$. Nariši graf rešitev in primerjaj napako rešitve za različno število intervalov.

Postopek rešitve implementiraj v *Python*® okolju z uporabo *NumPy*® knjižnice.

Rezultate primerjaj z Eulerjevo metodo.

Poglavje – 3

Navadne Diferencialne Enačbe višjega reda – enočlene metode

Kako se pa lotimo problema reševanja NDE 2. reda in več?

Vse NDE višjega reda, se da reducirati na sistem enačb 1. reda, z uvedbo nove spremenljivke za odvod.

Imamo začetni problem 2. reda

$$\begin{aligned} y'' &= f(x, y, y'), \\ y(x_0) &= y_0, \\ y'(x_0) &= y_0'. \end{aligned}$$

Numerično ga rešimo tako, da ga prevedemo na sistem enačb 1. reda

$$\begin{aligned} y' &= p, \\ p' &= f(x, y, p), \end{aligned}$$

$$\begin{aligned} y(x_0) &= y_0, \\ p(x_0) &= y_0'. \end{aligned}$$

Sedaj se pojavi vprašanje: Kako rešimo pa sistem enačb 1. reda?

Sistem NDE 1. reda

V primeru sistema enačb 1. reda uporabimo **vektorski zapis** **prvih odvodov** in **začetnih pogojev**

$$\boldsymbol{y}' = (y', p_1', p_2', \ldots, p_m')$$

$$\boldsymbol{y}_0 = (y(x_0), p_1(x_0), p_2(x_0), \ldots, p_m(x_0))$$

kjer smo uporabili zapis novih spremenljivk

$$y' = p_1, \quad y'' = p_2, \quad y''' = p_3, \ \ldots, \quad y^{(m)} = p_m.$$

Posledica: Enako lahko zapišemo **numerično metodo** v **vektorski obliki**!

Primer Eulerjeva metoda v **vektorski** obliki

$$\boldsymbol{y}_{n+1} = \boldsymbol{y}_n + h\, \boldsymbol{f}(x_n, \boldsymbol{y}_n) + \mathcal{O}(h^2), \quad n = \{0, 1, 2, 3, \ldots, N-1\},$$

$$\boldsymbol{y}(x_0) = \boldsymbol{y}_0.$$

Matematičen model gibanje planetov

Privlačna sila med planeti je gravitacijska sila, ki pojema s kvadratom razdalje

$$F_g = G\frac{m_1\, m_2}{r^2},$$

kjer je **gravitacijska konstanta** $G = 6.674 \times 10^{-11}\,\mathrm{m}^3\mathrm{kg}^{-1}\mathrm{s}^{-2}$, m_1 masa Sonca, m_2 masa planeta in r razdalja med njima.

2. Newtonov zakon pravi $F = m\,a = m\,\ddot{x}$, tako lahko sedaj silo in pospeške povežemo med seboj

$$G\frac{m_1\, m_2}{r^2}\frac{\boldsymbol{r}}{r} = m_2\ddot{\boldsymbol{r}}, \quad \boldsymbol{r} = (x, y),$$

kjer pa je sedaj $\boldsymbol{r}$ položaj planeta v ravnini gibanja okoli Sonca. Končna enačba gibanja je tako vektorska NDE 2. reda, ki jo zapišemo v reducirani obliki

$$\boxed{\ddot{\boldsymbol{r}} = -\kappa\frac{\boldsymbol{r}}{r^3}, \quad \kappa = G\,m_1.}$$

PAZI: Minus, ker je privlačna sila. Za vsako koordinato sledi

$$\ddot{x} = -\kappa\frac{x}{(x^2 + y^2)^{3/2}}, \quad \ddot{y} = -\kappa\frac{y}{(x^2 + y^2)^{3/2}}$$

Matematičen model gibanje planetov

Pretvorimo naš sistem enačb 2. reda v nam bolj razumljivo obliko. V tem primeru je neodvisna spremenljivka čas in zapisana kot parameter t, posamezne koordinate pa v obliki vektorja $\boldsymbol{x} = (x_1, x_2)$, ki pa je časovno odvisen $\boldsymbol{x} = \boldsymbol{x}(t)$.

$$\frac{\mathrm{d}^2 x_1}{\mathrm{d}t^2} = \ddot{x}_1 = -\kappa \frac{x_1}{(x_1^2 + x_2^2)^{3/2}},$$
$$\frac{\mathrm{d}^2 x_2}{\mathrm{d}t^2} = \ddot{x}_2 = -\kappa \frac{x_2}{(x_1^2 + x_2^2)^{3/2}}.$$

Če sedaj zamenjamo imena spremenljivk $t \to x$ in $\boldsymbol{x} \to \boldsymbol{y}$ ali $(x_1, x_2) \to (y_1, y_2)$, ki smo jih uporabljali pri zapisu numeričnih metod, dobimo

$$\frac{\mathrm{d}^2 y_1}{\mathrm{d}x^2} = y_1'' = -\kappa \frac{y_1}{(y_1^2 + y_2^2)^{3/2}},$$
$$\frac{\mathrm{d}^2 y_2}{\mathrm{d}x^2} = y_2'' = -\kappa \frac{y_2}{(y_1^2 + y_2^2)^{3/2}}.$$

Sledi pretvorba enačb 2. reda v sistem enačb 1. reda z začetnimi pogoji

$$
\begin{aligned}
y_1' &= p_1, & y_2' &= p_2,\\
p_1' &= -\kappa \frac{y_1}{(y_1^2+y_2^2)^{3/2}}, & p_2' &= -\kappa \frac{y_2}{(y_1^2+y_2^2)^{3/2}},
\end{aligned}
$$

$$
\begin{aligned}
y_1(x_0) &= y_0,\ \text{(to je } x \text{ koordinata)} & y_2(x_0) &= y_0,\ \text{(to je } y \text{ koordinata)}\\
p_1(x_0) &= y_0',\ \text{(to je } v_x \text{ hitrost)} & p_2(x_0) &= y_0',\ \text{(to je } v_y \text{ hitrost)}
\end{aligned}
$$

kjer smo določiti tudi začetne pogoje (koordinate in hitrosti) v začetnem času $t_0 = x_0$.

Sedaj v bistvu rešujemo problem $\boldsymbol{y} = (y_1, p_1, y_2, p_2)$ z začetnim pogoji $\boldsymbol{y}_0 = (y_{10}, p_{10}, y_{20}, p_{20})$.

Naloga – Gibanje planetov

Reši sistem NDE 1. reda

$$y_1' = p_1, \qquad y_2' = p_2,$$

$$p_1' = -\kappa \frac{y_1}{(y_1^2 + y_2^2)^{3/2}}, \qquad p_2' = -\kappa \frac{y_2}{(y_1^2 + y_2^2)^{3/2}},$$

$$y_1(x_0) = y_{10}, \text{ (to je } x \text{ koordinata)} \qquad y_2(x_0) = y_{20}, \text{ (to je } y \text{ koordinata)}$$

$$p_1(x_0) = y_{10}', \text{ (to je } v_x \text{ hitrost)} \qquad p_2(x_0) = y_{20}', \text{ (to je } v_y \text{ hitrost)}$$

s pomočjo **Eulerjeve** in **modificirane Eulerjeve** metode.

Kot parametre vzemi

- začetni položaj: $\boldsymbol{r} = (1, 0)$
- začetna hitrost: $\boldsymbol{v} = (0, 0.15)$
- konstanta: $\kappa = 0.015$
- časovni korak: $\Delta t = 1$

Metodo implementiraj v *Python*® okolju (*NumPy*® in *MatPlotLib*®). Dejte se poigrat s parametri!

$\boxed{\text{Poglavje – 4}}$

Navadne Diferencialne Enačbe – več člene metode

Več člene metode

Pri več členih metodah , uporabimo izračunane vrednosti iz prejšnjih točk

$$y_{n+3} = f(x_n, y_n, y_{n+1}, y_{n+2})$$

Več členih metod mi ne bomo obravnavali!

Za pregled več členih metod je priporočljivo branje knjige

Bor Plestenjak, **Razširjen uvod v numerične metode**, DMFA, 2015