

3 Arvutuste organiseerimine

Sisukord

3 Arvutuste organiseerimine	28
3.1 Sammu pikkuse muutmine	29
3.2 Sammu poolitamise võte	30
3.3 Runge-Kutta meetodite paariskasutamine	33
3.4 Jäigad diferentsiaalvõrrandid	36
3.5 Tagurpidi sammuga Euler'i meetod *	38

Märksõnad

1. Sammu h muutmine (adaptiivne meetod).
2. Sammu poolitamise meetod.
3. Runge-Kutta paaride kasutamine.
4. Jäigad diferentsiaalvõrrandid.

3.1 Sammu pikkuse muutmine

Seni vaatlesime olukorda, kus sammude pikkus h fikseeritakse ning kõik arvutused viiakse läbi selle sama sammuga h . Viimane on aga arvuti ressursi ja arvutusaega raiskav tegevus. Sõltuvalt lahendi muutumise kiirusest, võib vaja minna kas siis väga väikest sammude pikkust h või siis hoopis suuremat h väärtust. Osutub, et viimast saab arvutuste käigus hinnata.

Oletame hetkeks, et meil õnnestub kuidagi ligikaudselt arvutada sammul tekkiva vea suurust

$$E_i = |y_i - y(x_i)|.$$

Varasemast teame, et k -järku lähismeetodi sammul tekkiv viga on $E_i = \mathcal{O}(h^{k+1})$ (s.t. $E_i \leq C h^{k+1}$ protsessis $h \rightarrow 0$). Andes ette meile sobiva maksimumsuhelise vea ε , tahame, et kehtiks võrratus

$$\frac{E_i}{|y_i|} < \varepsilon.$$

Kuna leidub konstant $\Theta > 0$, et

$$E_i \approx \Theta h^{k+1},$$

siis leidub ka selline sammude pikkus h_* , et kehtib võrdus

$$\varepsilon \cdot |y_i| \approx \Theta \cdot h_*^{k+1}. \quad (3.1)$$

Lahendades viimased kaks võrrandit tundmatu konstandi Θ suhtes, saame $\Theta \approx \frac{E_i}{h_i^{k+1}}$ ja $\Theta \approx \frac{\varepsilon \cdot |y_i|}{h_*^{k+1}}$ ning siit ka seose

$$\mathbf{h}_* = 0.8 \cdot \left(\frac{\varepsilon \cdot |y_i|}{E_i} \right)^{\frac{1}{k+1}} \cdot h_i \quad (3.2)$$

1. Olgu suhteline viga väiksem kui lubatud arvutustäpsus, s.t.

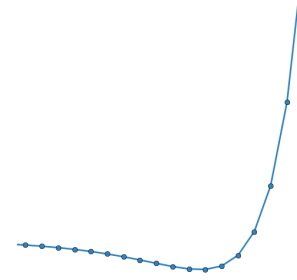
$$\frac{E_i}{|y_i|} < \varepsilon.$$

Sel juhul juba arvutatud lähisväärtus y_i sobib (leitud sammude pikkusega h_i) ning järgmiseks sammuks võtame $h_{i+1} = h_*$.

2. Kui suhteline viga on suurem kui lubatud arvutustäpsus, s.t.

$$\frac{E_i}{|y_i|} > \varepsilon,$$

siis hetkel kasutusel oleva h_i asemel võtame h_* ja arvutame y_i uuesti. Kui selliselt arvutatav uus y_i peaks ikkagi olema ebatäpne, siis jätkame sammude poolitamise taktikaga, s.t. võtame $h_* = \frac{h_*}{2}$ ning jätkame kuni soovitud täpsus on saavutatud.



Kuidas õnnestub seda arvutada, näitame hiljem.

Ligikaudse arvu \tilde{a} relatiivseks veaks loetakse arvu $\delta > 0$, mis rahuldab võrratust

$$\frac{|a - \tilde{a}|}{|\tilde{a}|} \leq \delta.$$

NB! Konstant Θ on meile tundmatu, me ei saa seda välja arvutada. AGA ... me taandame ta välja.

Kordaja 0.8 on nõ. "turvalisuse parameeter" vältimaks igasugu erijuhte ja arvestamiseks juba tehtud ümardamisvigu. Arvu 0.8 asemel võib olla ka muu ühest väiksem positiivne arv.

Märkus 3.1

Kui $\frac{E_i}{|y_i|} < \varepsilon$, siis üldiselt uus samm pikkus $h_* > h_i$. Kui $\frac{E_i}{|y_i|} > \varepsilon$, siis üldiselt uus samm pikkus $h_* < h_i$.

Numbrilist meetodit, milles on arvesse võetud ka sammu pikkuse muutmine, nimetatakse ka **adaptiivseks meetodiks**.

Märkus 3.2

Kogu eelnevas arutluses peaks tegelikult praktikas kasutama suhtelise vea $\frac{E_i}{|y_i|}$ asemel suhet

$$\frac{E_i}{\max\{|y_i|, \theta\}},$$

kus $\theta > 0$ on mingi sobivalt väike väärtus kaitsmaks nulliga jagamise vastu.

Näide 3.1 Kui me kasutame Runge-Kutta 4. järku meetodis $h = 0.001$, saades näiteks

$$\frac{E_1}{|y_1|} = \frac{|y_1 - y(x_1)|}{|y_1|} \approx 0.5$$

ja soovime, et punktis $x_1 = x_0 + h$ oleks suhteline viga umbes $\varepsilon = 10^{-8}$, siis peaksime oma viimased arvutused tühistama ja kasutama uute arvutuste jaoks uut sammu pikkusega

$$h_* = 0.8 \times 0.001 \times \left| \frac{10^{-8}}{0.5} \right|^{\frac{1}{5}} \approx 2.3 \times 10^{-5}.$$

Kui aga oleks selgunud, et saime vea mitte 0.5, vaid 10^{-15} , siis liikudes punktist x_1 edasi võiksime järgmiseks sammuks h_2 võtta suurema väärtuse, näiteks $h_2 = 0.02$ (kuna $0.8 \times 0.001 \times \left| \frac{10^{-8}}{10^{-15}} \right|^{1/5} \approx 0.02$).

◇ ◇ ◇

3.2 Sammu poolitamise võte

Toodud arutlused sammu pikkuse muutmiseks eeldasid, et me teame sammul tekkivat viga E_i . Tegelikult me seda täpselt ei tea (meil on olemas ainult hinnang $E_i = O(h^{k+1})$). Küll aga on võimalik E_i arvutada ligikaudu.

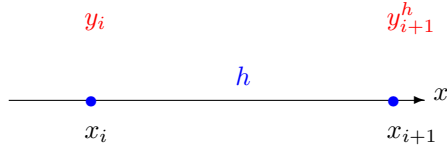
Siinjuures on oluline, et me ei soovi selleks teha liiga palju lisaarvutusi. Vaatleme hetkel esimest võimalust: sammu h poolitamise võtet. Oletame, et me kasutame y_{i+1} leidmiseks k -järku meetodit, kusjuures eelmine väärtus y_i on leitud täpselt. Siis sammul tekkiv viga on

$$E_{i+1} = |y_{i+1} - y(x_{i+1})| \approx \Theta h^{k+1},$$

kus Θ on tundmatu konstant, millest me järgnevalt üritamegi vabaneda.

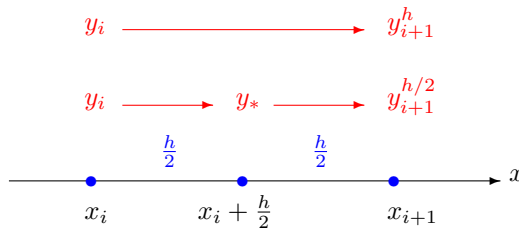
Esiteks leiame ühesammulise meetodiga lähislahendi sammuga h , s.t. nii nagu tavaliselt,

$$y_{i+1}^h := y_i + \Delta y_i.$$



Järgnevalt leiame sama meetodiga lähislahendi punktis $x_{i+1} = x_i + h$, kuid seekord poole sammuga $h/2$, s.t. me rakendame oma meetodit kaks korda järjest. Selleks kasutame lõigu keskpunkti $x_i + h/2$, kus lähislahendi tähistame y_* ,

$$y_* = y_i + \Delta y_i^{h/2}, \quad y_{i+1}^{h/2} = y_* + \Delta y_*^{h/2}.$$



Sellisel juhul saame esimesel poolsammul tekkiva vea

$$y_* - y(x_i + h/2) \approx \Theta \left(\frac{h}{2} \right)^{k+1}.$$

Järgmisel poolsammul tehakse sama suurusjärku viga, mistõttu saame tulemuseks nende vigade summa

$$y_{i+1}^{h/2} - y(x_{i+1}) \approx 2\Theta \left(\frac{h}{2} \right)^{k+1}.$$

Toodud seostest $y_{i+1}^h - y(x_{i+1})$ ja $y_{i+1}^{h/2} - y(x_{i+1})$ kohta leiame, et

$$y_{i+1}^h - y(x_{i+1}) \approx \Theta h^{k+1} = 2^k \left[2\Theta \left(\frac{h}{2} \right)^{k+1} \right] \approx 2^k \left(y_{i+1}^{h/2} - y(x_{i+1}) \right).$$

Seega $y_{i+1}^h - y(x_{i+1}) \approx 2^k \left(y_{i+1}^{h/2} - y(x_{i+1}) \right)$ ja siit saame avaldada

$$y(x_{i+1}) \approx y_{i+1}^{h/2} + \frac{y_{i+1}^{h/2} - y_{i+1}^h}{2^k - 1} \quad (3.3)$$

Märkus 3.3

Märgime, et lähislahendiks punktis x_{i+1} võib võtta kas siis $y_{i+1}^{h/2}$ (kuna see peaks olema täpsem kui sammuga h leitud y_{i+1}^h) või siis valemiga (3.3) täpsustatud lahend $y(x_{i+1})$. Valemis (3.3) toodud lahendi täpsustamist nimetatakse ka lokaalseks ekstrapoleerimiseks.

Sammul tekkiva vea E_i ligikaudne väärtus on leitav valemiga

$$E_i = |y_i^{h/2} - y(x_i)| \approx \frac{|y_i^{h/2} - y_i^h|}{2^k - 1}. \quad (3.4)$$

Näide 3.2 Vaatleme Cauchy ülesannet

$$y' = x, \quad y(-1) = 0.$$

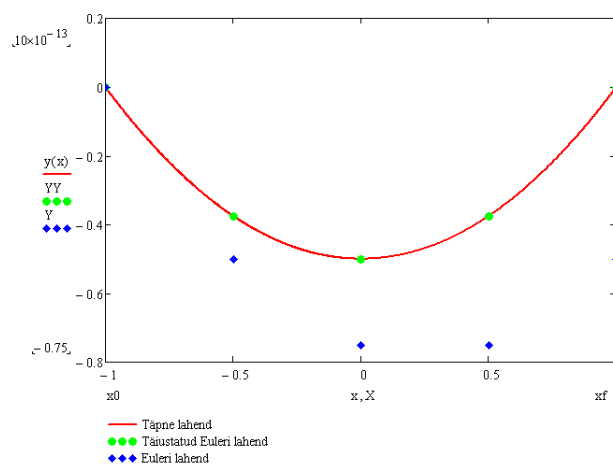
Leiame Euler'i meetodiga y väärtuse punktis $x = 0$ sammuga $h = 1$:

$$y_0^h = 0 + 1 \times (-1) = -1.$$

Leiame uuesti väärtuse punktis null, seekord sammuga $h = \frac{1}{2}$:

$$y_{-\frac{1}{2}} = 0 + \frac{1}{2} \times (-1) = -\frac{1}{2},$$

$$y_0^{h/2} = -\frac{1}{2} + \frac{1}{2} \times (-\frac{1}{2}) = -\frac{3}{4}.$$



Seega sammul tekkinud viga on Euler'i meetodiga ($k = 1$) ligikaudselt leitav

$$E_0 = |y_0^{h/2} - y(0)| \approx \frac{|y_0^{h/2} - y_0^h|}{2^1 - 1} = \left| -\frac{3}{4} - (-1) \right| = \frac{1}{4}.$$

Vahetult võib kontrollida, et meie ülesande täpne lahend on $y(x) = \frac{x^2}{2} - \frac{1}{2}$. Seega punktis null on tegelik viga

$$|y_0^{h/2} - y(0)| = \left| -\frac{3}{4} + \frac{1}{2} \right| = \frac{1}{4}.$$

Enamasti siiski ei juhtu, et saaksime ligikaudse vea ja tegeliku vea omavahel nii täpselt klappima panna. Kasutame $y_0^{h/2}$ täpsustamiseks valemit (3.3):

$$\tilde{y}_0 = y_0^{h/2} + \frac{y_0^{h/2} - y_0^h}{2^1 - 1} = -\frac{3}{4} + \frac{1}{4} = -\frac{1}{2}.$$

Seekord on tegelik viga

$$y(0) - \tilde{y}_0 = -\frac{1}{2} + \frac{1}{2} = 0.$$

Antud juhul täpsustamine aitas nii hästi, et kogemata saime koguni täpse väärtuse.

◇ ◇ ◇

Näide 3.3 Vaatleme näiteks Runge-Kutta 4. järku meetodit (sel juhul $k = 4$). Sammul tekkiv viga on siis ligikaudselt leitav valemiga

$$E_{i+1} = |y_{i+1}^{h/2} - y(x_{i+1})| \approx \frac{|y_{i+1}^{h/2} - y_{i+1}^h|}{15}. \quad (3.5)$$

Lahendi y väärtus punktis x_{i+1} on täpsustatav valemiga

$$y_{i+1} = y_{i+1}^{h/2} + \frac{y_{i+1}^{h/2} - y_{i+1}^h}{15}, \quad (3.6)$$

mis on tinglikult 5-järku meetod (vt. kommentaare [2] Runge-Kutta meetodite peatükist). \diamond

$\diamond \diamond \diamond$

Märkus 3.4

Lokaalne ekstrapoleeriv valem (3.6) on küll väga kõrget järku meetod (tinglikult 5. järku meetod) koos võimalusega arvutada ligikaudselt ka sammul tekkivat viga, kuid antud meetod nõuab igal sammul 11 uut funktsiooni $f(x, y)$ väärtuse arvutamist (4 sammuga h ja 4+4 sammuga $h/2$, millest ühine on ainult $K_{i,1} = f(x_i, y_i)$).

Lisaks, näiteks 5. järku Runge-Kutta meetodi korral peaksime tegema 17 uut funktsiooni $f(x, y)$ väärtuse arvutamist ($6 + (2 \times 6) - 1 = 17$). See teeb sammu poolitamise võtte kõrget järku meetodite korral väga ebaefektiivseks.

3.3 Runge-Kutta meetodite paariskasutamine

Olgu y_{i+1} leitud madalamat ja w_{i+1} leitud kõrgemat järku meetodiga. Sel juhul võib sammul tekkiva vea "arvutamiseks" kasutada ligikaudset võrdust

$$E_{i+1} \approx |y_{i+1} - w_{i+1}|. \quad (3.7)$$

Antud ideed saab täpsustada võttes konkreetsed meetodid.

Märkus 3.5

Kõrgemat järku meetodiga leitud lähislahendi w_{i+1} rollis võib olla ka selline lahend, mis on leitud selle sama madalamat järku meetodiga, kuid kaks korda väiksema sammu pikkusega h . Sellisel võttel on omad eelised, kuid Runge-Kutta meetodite korral on see liialt ajamahukas, kuna igal sammul tuleb teha liiga palju funktsiooni $f(x, y)$ väärtuse arvutamisi.

Runge-Kutta 2/3 järku paariks on järgmine 2. ja 3. järku meetodite paar:

$$\begin{aligned} y_{i+1} &= y_i + \frac{h}{2} (K_{i,1} + K_{i,2}), \\ w_{i+1} &= y_i + \frac{h}{6} (K_{i,1} + K_{i,2} + 4K_{i,3}) \end{aligned} \quad (3.8)$$

kus

$$\begin{aligned} K_{i,1} &= f(x_i, y_i), \\ K_{i,2} &= f(x_i + h, y_i + hK_{i,1}), \\ K_{i,3} &= f\left(x_i + \frac{h}{2}, y_i + \frac{h}{2} \frac{K_{i,1} + K_{i,2}}{2}\right). \end{aligned} \quad (3.9)$$

Esimene meetod y_{i+1} leidmiseks on tegelikult trapetsmeetod ja teine meetod w_{i+1} arvutamiseks on üks Runge-Kutta 3. järku meetoditest. Ligikaudseks sammul tekkiva vea suuruseks saame

$$E_{i+1} \approx |y_{i+1} - w_{i+1}| = \frac{h}{3} |K_{i,1} + K_{i,2} - 2K_{i,3}| \quad (3.10)$$

Trapetsmeetodi sammul tekkiva vea hindamiseks peame seega tavapärase $K_{i,1}, K_{i,2}$ leidmisele lisaks tegema ühe lisaarvutuse, $K_{i,3}$ arvutamise.

Näide 3.4 Vaatleme ülesannet

$$y' = xy + x^3, \quad y(0) = 1, \quad x \in [0, 2],$$

mille täpne lahend on $y(x) = 3e^{x^2/2} - x^2 - 2$. Kasutame Runge-Kutta 2/3 järku paari (3.8), kusjuures sammu pikkuse muutmiseks valime lihtsa sammu poolitamise võtte (alustades sammust $h_0 = 0.5$). Toome tabeli kasutatud sammude arvu n , tegeliku vea ja suhtelise vea

$$\max_{i=0, \dots, n} |y_i - y(x_i)|, \quad \max_{i=1, \dots, n} \frac{|y_i - y(x_i)|}{|y_i|},$$

kohta. Algoritmis me nõuame, et ette antud veataseme ε jaoks kehtiks $\frac{E_i}{|y_i|} < \varepsilon$. Lisaks kuvame, mitu korda suhteline viga osutus väiksemaks, kui nõutud veatase ε .

n	ε	$\max y_i - y(x_i) $	$\max \frac{ y_i - y(x_i) }{ y_i }$	$\frac{\varepsilon y_i }{\max y_i - y(x_i) }$
8	10^{-2}	4.53×10^{-2}	3.18×10^{-3}	3.15
43	10^{-4}	4.14×10^{-4}	2.63×10^{-5}	3.80
184	10^{-6}	4.70×10^{-6}	2.91×10^{-7}	3.44
872	10^{-8}	6.81×10^{-8}	5.04×10^{-9}	1.99
4659	10^{-10}	5.16×10^{-10}	3.84×10^{-11}	2.60
21037	10^{-12}	5.66×10^{-12}	4.44×10^{-13}	2.25
90457	10^{-14}	1.07×10^{-13}	7.41×10^{-15}	1.35

Tabelist võime näha, et algoritm on lähislahendi arvutanud kõikides sõlmedes nõutud täpsusega (kuna maksimaalne suhteline viga ei ületa soovitud taset ε) ning seejuures ei ole tehtud üleliia palju tööd (viga ei ole palju kordi väiksem, kui ette antud veatase ε).

◇ ◇ ◇

Üheks tuntumaks sammu pikkust h kontrollivaks Runge-Kutta meetodiks peetakse **Runge-Kutta-Fehlbergi 4/5 järku paari**, mis sisaldab järgmisi 4. ja 5. järku Runge-Kutta meetodite paare:

$$\begin{aligned} y_{i+1} &= y_i + h \left(\frac{25}{216} K_{i,1} + \frac{1408}{2565} K_{i,3} + \frac{2197}{4104} K_{i,4} - \frac{1}{5} K_{i,5} \right), \\ w_{i+1} &= y_i + h \left(\frac{16}{135} K_{i,1} + \frac{6656}{12825} K_{i,3} + \frac{28561}{56430} K_{i,4} - \frac{9}{50} K_{i,5} + \frac{2}{55} K_{i,6} \right) \end{aligned} \quad (3.11)$$

kus

$$\begin{aligned} K_{i,1} &= f(x_i, y_i), \\ K_{i,2} &= f\left(x_i + \frac{h}{4}, y_i + \frac{h}{4} K_{i,1}\right), \\ K_{i,3} &= f\left(x_i + \frac{3}{8}h, y_i + \frac{h}{32} \left[3K_{i,1} + 9K_{i,2} \right]\right), \\ K_{i,4} &= f\left(x_i + \frac{12}{13}h, y_i + h \left[\frac{1932}{2197} K_{i,1} - \frac{7200}{2197} K_{i,2} + \frac{7296}{2197} K_{i,3} \right]\right), \\ K_{i,5} &= f\left(x_i + h, y_i + h \left[\frac{439}{216} K_{i,1} - 8K_{i,2} + \frac{3680}{513} K_{i,3} - \frac{845}{4104} K_{i,4} \right]\right), \\ K_{i,6} &= f\left(x_i + \frac{h}{2}, y_i + h \left[-\frac{8}{27} K_{i,1} + 2K_{i,2} - \frac{3544}{2565} K_{i,3} + \frac{1859}{4104} K_{i,4} - \frac{11}{40} K_{i,5} \right]\right). \end{aligned} \quad (3.12)$$

Ligikaudseks sammul tekkiva vea suuruseks saame

$$E_{i+1} \approx |w_{i+1} - y_{i+1}| = h \left| \frac{1}{360} K_{i,1} - \frac{128}{4275} K_{i,3} - \frac{2197}{75240} K_{i,4} + \frac{1}{50} K_{i,5} + \frac{2}{55} K_{i,6} \right| \quad (3.13)$$

Märgime, et meetod ei ole ressursse liialt "raiskav", kuna tegelikkuses 5. järku meetodi jaoks tuleb kasutada minimaalselt 6 funktsiooni $f(x, y)$ arvutamist (teoreetiliselt ei ole võimalik vähemaga läbi ajada) ja meil siin ongi vaja arvutada 6 funktsiooni $f(x, y)$ väärtust. Tõsi, viimane on ikkagi rohkem kui „vaja oleks“. Kui täpsus sobib, siis lähislahendi väärtuseks punktis x_{i+1} võib võtta w_{i+1} (kuna see on leitud kõrgemat järku meetodiga).

Märkus 3.6

Runge-Kutta-Fehlbergi skeem on kaasatud mitmetesse arvutusmatemaatika programmipakettidesse (tüüpiliselt sisaldab vastav käsk või parameeter ühendit "RKF45").

Tuntumatest võib veel nimetada Dormand-Prince'i 4/5 järku paari (vt. [3]), mida kasutab näiteks MatLab'i tarkvara käsus „ode45“. Kõik need skeemid esituvad üsna pikkade ja keeruliste valemite kaudu.

3.4 Jäigad diferentsiaalvõrrandid

Anneme mitteformaalse jäiga diferentsiaalvõrrandi (*stiff differential equation*) mõiste veidi hiljem. Alustame näitega.

Näide 3.5 Vaatleme algtingimusega ülesannet

$$y' = 10(1 - y), \quad y(0) = \frac{1}{2}. \quad (3.14)$$

Fikseerime sammu pikkuse h ja rakendame Euler'i meetodit,

$$y_{i+1} = y_i + h f(x_i, y_i) = y_i + 10h(1 - y_i) = (1 - 10h)y_i + 10h.$$

Märgime, et ülesande täpne lahend on $y(x) = 1 - \frac{1}{2}e^{-10x}$ ja $y \rightarrow 1$ protsessis $x \rightarrow \infty$ (tegelikult on y peaaegu 1 juba üsna väikese x korral). Antud juhul me võime Euler'i meetodi kirja panna kui iteratsioonimeetodi

$$g(x) = (1 - 10h)x + 10h. \quad (3.15)$$

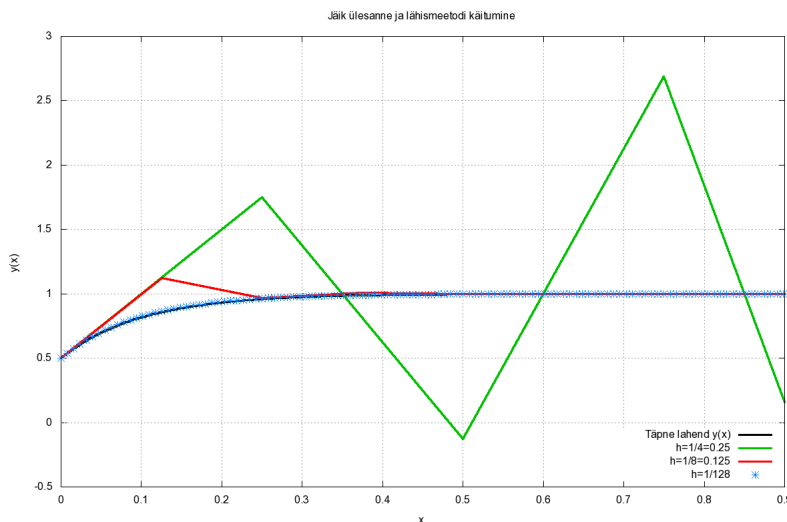
Märkus 3.7

Kui meil on antud eeskiri $x_{n+1} = g(x_n)$, $x_n \in [a, b]$ ($n = 0, 1, 2, \dots$), $g \in C^1[a, b]$ ja kehtib võrratus

$$|g'(x)| \leq \beta < 1, \quad x \in [a, b], \quad (3.16)$$

siis iteratsioonimeetod koondub geomeetrilise progressiooni kiirusega, mille teguriks on β (vt. [5]).

Arvestades märkust 3.4, siis näeme, et $g(x)$ koondub punktis x juhul, kui $|g'(x)| = |1 - 10h| < 1$. Selle võrratuse lahendamisel saame, et iteratsioonimeetod koondub kui $0 < h < 0.2$. Iga suurema sammu pikkuse h korral ei pruugi iteratsioonimeetod koonduda.

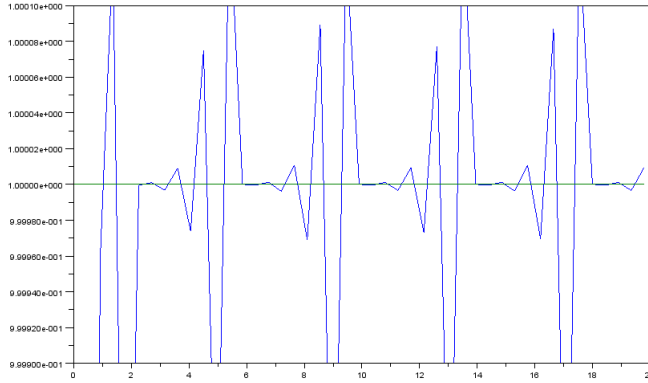


Toome joonisel Euler'i meetodiga leitud lahendid $h = \frac{1}{4} = 0.25$, $h = \frac{1}{8} = 0.125$ ja $h = \frac{1}{128}$ korral. Näeme, et $h > 0.2$ korral käitub lähilähend väga halvasti, kuid $h < 0.2$ korral juba piisavalt hästi. Antud joonis iseloomustab jäikade võrrandite lähilähendeid, kus tüüpiliselt annab lähilähendi halba käitumist parandada, vähendades sammu pikkust h vajaliku väärtuseni (iseasi, kas see on mõistlik ja kuidas ära tunda, milline see väärtus olema peab).

◇ ◇ ◇

Märkus 3.8

Niimoodi ei käitu ainult Euler'i meetod, vaid näiteks ka tavatingimustes „suurepärase“ Runge-Kutta-Fehlberg'i paari korral (siin leitud SciLab'i „ode“ käsu ja „rkf“ parameetriga) näeme graafikut suurendades, et lähislahend käitub väga rahutult:

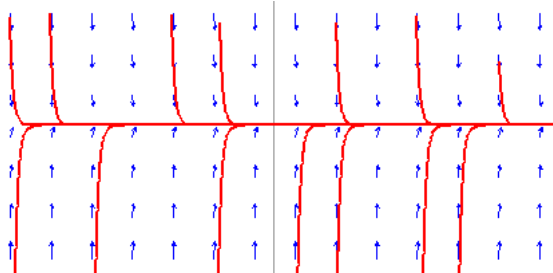
**Definitsioon 3.1**

Jäikuse mõistet ei anta edasi täpselt, see on rohkem intuiitiivne. Me ütleme, et diferentsiaalvõrrand $y' = f(x, y)$ on jäik (*stiff*), kui tema lahend läheb kiirest langusest (tõusust) järsult üle vähe muutuvaks lahendiks.

Üldiselt on see seotud funktsiooni $f(x, y)$ osatuletise $\frac{\partial f(x, y)}{\partial y}$ suure negatiivse väärtusega kohal $y = c$.

Märkus 3.9

Võrrandi $y' = 10(1 - y)$ korral $\frac{\partial}{\partial y} f(x, y) = -10$ (mingis mõttes jäikuse mõõdupuu, mida negatiivsem on $f_y(x, y)$ kohal $y = c$, seda jäigem on diferentsiaalvõrrand). Joonisel on toodud võrrandi vektorväli koos erinevate lahenditega, mis kõik koonduvad konstantseks lahendiks $y(x) \rightarrow 1$, $x \rightarrow \infty$.



Märkus 3.10

Jäiga võrrandi korral aitab tihti sammu h pikkuse vähendamine, kuid kahjuks ei ole nii lihtne ette näha, kui suur või väike siis h peaks ikkagi olema. Praktikaks kasutatakse jääkade võrrandite jaoks kas siis ilmutamata meetodeid või spetsiaalselt välja töötatud meetodeid.

3.5 Tagurpidi sammuga Euler'i meetod *

Seni oleme vaadelnud nn. „ilmutatud“ meetodeid (*explicit*), kus igal sammul arvutatav y_{i+1} asub ainult võrrandi $y_{i+1} = y_i + \Delta y_i$ vasakul poolel.

Definitsioon 3.2

Võrrandit $y_{i+1} = y_i + \Delta y_i$ nimetatakse „ilmutamata“ võrrandiks (*implicit*), kui tundmatu y_{i+1} asub ka võrrandi paremal pool liikmes Δy_i .

Osutub, et ilmutamata meetodid sobivad jääkade diferentsiaalvõrrandite lahendamiseks palju paremini, kui ilmutatud meetodid.

Olgu meil antud sammu pikkus h . Siis **tagurpidi sammuga Euler'i meetodiks** nimetatakse järgmist ilmutamata meetodit:

$$y_{i+1} = y_i + h f(x_i + h, y_{i+1}), \quad i = 0, 1, 2, \dots \quad (3.17)$$

Erinevalt tavalisest Euler'i meetodist, arvutatakse vektori tõus mitte punktis x_i , vaid järgmises punktis $x_{i+1} = x_i + h$. Probleem on aga selles, et me ei tea y_{i+1} väärtust.

Näide 3.6 Vaatleme eelmist algtingimustega ülesannet

$$y' = 10(1 - y), \quad y(0) = \frac{1}{2}. \quad (3.18)$$

Fikseerime sammu pikkuse h ja rakendame tagurpidi Euler'i meetodit,

$$y_{i+1} = y_i + h f(x_i + h, y_{i+1}) = y_i + 10h(1 - y_{i+1}),$$

millest

$$y_{i+1} = \frac{y_i + 10h}{1 + 10h}.$$

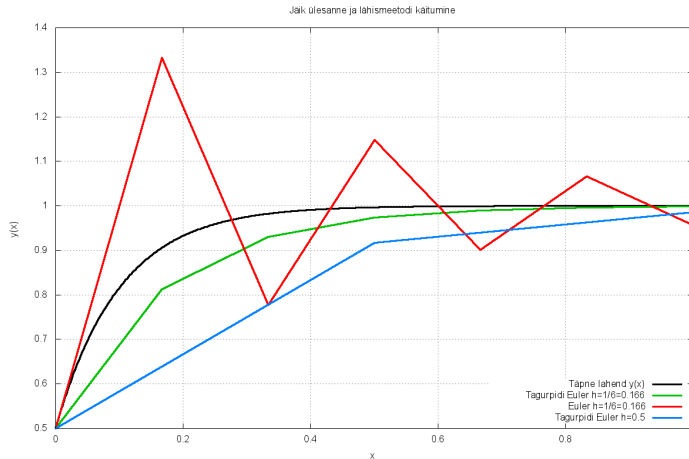
Analoogiliselt võime tagurpidi sammuga Euler'i meetodi kirja panna kui iteratsioonimeetodi

$$g(x) = \frac{x + 10h}{1 + 10h}. \quad (3.19)$$

Märkuse 3.4 järgi $g(x)$ koondub punktis x juhul, kui

$$|g'(x)| = \left| \frac{1}{1+10h} \right| < 1.$$

Viimane on aga alati tõene sõltumata sammu pikkusest $h > 0$.



◇ ◇ ◇

Märkus 3.11

Enamasti ei õnnestu ilmutamata meetodis niisama lihtsalt avaldada tundmatut y_{i+1} . Sel juhul tekib igal sammul i mittelineaarne võrrand y_{i+1} suhtes. Antud võrrandit saab lahendada näiteks Newton'i meetodiga (vt. [1, 3, 5])

$$z_{uus} = z_{vana} - \frac{g(z_{vana})}{g'(z_{vana})}, \quad (3.20)$$

kus funktsioon g on antud ilmutamata kujul $g(x) = 0$.

Näide 3.7 Vaatleme algtingimusega ülesannet

$$y' = y + 8y^2 - 9y^3, \quad y(0) = \frac{1}{2}, \quad x \in [0, 3].$$

Ka selle ülesande lahend koondub konstantseks lahendiks $y_*(x) = 1$ protsessis $x \rightarrow \infty$. Võtame osatuletise

$$\frac{\partial}{\partial y} f(x, y) = 1 + 16y - 27y^2,$$

mille väärtus $y = 1$ korral on -10 ja sellest järeldub, et võrrand on jääk (vähemalt mingil määral).

Ülesanne. Vaatleme lõigus $[0, 20]$ diferentsiaalvõrrandit

$$y' = 6y - 6y^2, \quad y(0) = \frac{1}{2}$$

Lahendage arvuti abiga võrrand tagurpidi sammuga Euler'i meetodiga ja tavalise Euler'i meetodiga. Umbes millisest h väärtusest koondub Euler'i meetod võrrandi statsionaarseks lahendiks? Võrrelge mõlema meetodi lahendi graafikuid erineva h jaoks.

Tagurpidi Euler'i meetod näeb välja järgmine:

$$y_{i+1} = y_i + h(y_{i+1} + 8y_{i+1}^2 - 9y_{i+1}^3), \quad i = 0, 1, 2, \dots$$

Newton'i iteratsioonimeetodi jaoks moodustame funktsiooni g selliselt, et $g(z) = 0$:

$$g(z) := 9h z^3 - 8h z^2 + (1 - h)z - y_i.$$

Tuletis $g'(z) = 27h z^2 - 16h z + 1 - h$ ja Newton'i iteratsioonimeetod võtab kuju

$$z_{uus} = z - \frac{9h z^3 - 8h z^2 + (1 - h)z - y_i}{27h z^2 - 16h z + 1 - h}.$$

Alglähendina z võib kasutada kas siis Euler'i meetodiga (või mingi muu meetodiga) leitud väärtust $z = y_i + h f(x_i, y_i)$ või siis jääkade ülesannete jaoks on parem võtta $z = y_i$. Edasi paneme uue z_{uus} iteratsiooniprotsessi paremasse poolde ja leiame sama algoritmiga uuesti z_{uus} ning jätkame seni, kuni kahe järjestikulise z väärtuse vahe on küllalt väike (sel juhul võtame y_{i+1} väärtuseks viimasena leitud z_{uus} väärtuse).

◇ ◇ ◇

Viited

- [1] J. Janno. Arvutusmeetodid. Tallinna Tehnikaülikool, 2008.
- [2] W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery. Numerical Recipes in Fortran 77. The Art of Scientific Computing, II Ed. Vol. 1 of Fortran Numerical Recipes. Cambridge University Press, reprinted 2001. ISBN 0-521-43064-X.
- [3] T. Sauer. Numerical analysis. Pearson, 2012.
- [4] E. Tamme. Arvutusmeetodid II. Tallinn, 1973.
- [5] E. Tamme, L. Võhandu, L. Luht. Arvutusmeetodid I. Tallinn, 1986.