

Uvod v numerične metode

Bor Plestenjak

Jadranska 21, soba 4.04

govorilne ure: po dogovoru

bor.plestenjak@fmf.uni-lj.si

asistent: Jan Grošelj

Literatura:

- zapiski,
- spletna učilnica FMF,
- B. Plestenjak, [Razširjen uvod v numerične metode](#), DMFA-založništvo, 2015.

Režim

- 2 sklopa domačih nalog - 20% pisne ocene
 - Matlab
 - reševanje kvizov iz spletne učilnice v računalniški učilnici
- 3 pisni izpiti - 80% pisne ocene
 - za pozitivno pisno oceno je potrebno:
 - a) iz domačih nalog in pisnega izpita zbrati skupaj vsaj 50%
 - b) pisati pozitiven pisni izpit
- ustni izpit
 - pogoj za pristop je pozitivna pisna ocena
 - pozitivna pisna ocena velja do konca izpitnega obdobja (10.9.2021)

1.1 Uvod

Ukvarjali se bomo z numeričnim reševanjem raznih matematičnih problemov.

Numerično reševanje: iščemo rešitev v numerični obliki,
npr. namesto $\sqrt{3}$ iščemo 1.73205....

Numerična metoda: postopek, ki iz danih numeričnih podatkov s končnim zaporedjem elementarnih operacij izračuna numerični rezultat

Elementarne operacije: odvisne od okolja, za nas bodo to

$$+, -, /, * \text{ in } \sqrt{}.$$

Uporabljali bomo program [Matlab](#), ki je na voljo vsem študentom FMF.

Kdaj potrebujemo numerične metode

Ko drugih možnosti ni, npr.:

- iskanje ničel polinoma pete stopnje: $x^5 + 3x - 1 = 0$,
- računanje določenega integrala: $\int_0^1 e^{x^2} dx$.

Kadar so udobnejše oz. manj zahtevne od analitičnih rešitev, npr.:

- računanje inverza matrike velikosti 100×100 ,
- Cardanove formule za ničle kubičnega polinoma.

Numerično računanje se razlikuje od točnega

Računamo s števili, ki so predstavljena v premični piki.

- Vzamemo kalkulator in izračunamo

$$100 * (100/3 - 33) - 100/3.$$

Rezultat je (ne)pričakovan, npr. v Matlabu dobimo

$$2.3448 \cdot 10^{-13}.$$

- Za množenje naj bi veljala asociativnost. A če vzamemo npr.

$$x = 0.1234567890$$

$$y = 0.0987654321$$

$$z = 0.9911991199,$$

v Matlabu dobimo

$$x * y * z - z * y * x = 1.7347 \cdot 10^{-18}.$$

Numerična matematika

Ukvarjali se bomo z numeričnim reševanjem naslednjih problemov:

- *uvod v numerično računanje*: premična pika, občutljivost in stabilnost.
- *sistem linearnih enačb*: reši sistem $Ax = b$ za $A \in \mathbb{R}^{n \times n}$ in $b \in \mathbb{R}^n$.
- *nelinearne enačbe*: reši enačbo $f(x) = 0$, kjer je $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$.
- *linearni problem najmanjših kvadratov*: poišči $x \in \mathbb{R}^n$, ki minimizira $\|Ax - b\|_2$ za $A \in \mathbb{R}^{m \times n}$ in $b \in \mathbb{R}^m$, kjer je $m > n$.
- *lastne vrednosti*: izračunaj lastne vrednosti in vektorje matrike $A \in \mathbb{R}^{n \times n}$.
- *interpolacija*: poišči polinom, ki gre skozi točke $(x_0, f(x_0)), \dots, (x_n, f(x_n))$.
- *integriranje*: izračunaj integral $\int_a^b f(t)dt$.
- *diferencialne enačbe*: reši začetni problem $y' = f(x, y)$, $y(x_0) = y_0$.

Še več: Numerična linearna algebra, predmeti iz skupine M4 na drugi stopnji.

Absolutna in relativna napaka

Pri numeričnem računanju vedno dobimo le numerični približek za točno rešitev.

Razlika med približkom in točno vrednostjo je **napaka približka**.

absolutna napaka = približek – točna vrednost,

relativna napaka = $\frac{\text{absolutna napaka}}{\text{točna vrednost}}$.

Naj bo x točna vrednost, $\hat{x}$ pa približek za x .

- Če je $\hat{x} = x + d_a$, potem je $d_a = \hat{x} - x$ absolutna napaka.
- Če je $\hat{x} = x(1 + d_r)$ oziroma $d_r = \frac{\hat{x} - x}{x}$, potem je d_r relativna napaka.

1.2 Premična pika

V računalniku so števila zapisana v premični piki kot

$$x = \pm m \cdot b^e,$$

kjer je $m = 0.c_1c_2 \dots c_t$ *mantisa* in

- b : *baza* (2, lahko tudi 10 ali 16),
- t : *dolžina mantise*,
- e : *eksponent* v mejah $L \leq e \leq U$,
- c_i : *števke* v mejah od 0 do $b - 1$.

Če je $c_1 \neq 0$, potem je število *normalizirano*, sicer pa *subnormalizirano*.

Množico predstavljenih števil označimo s $P(b, t, L, U)$.

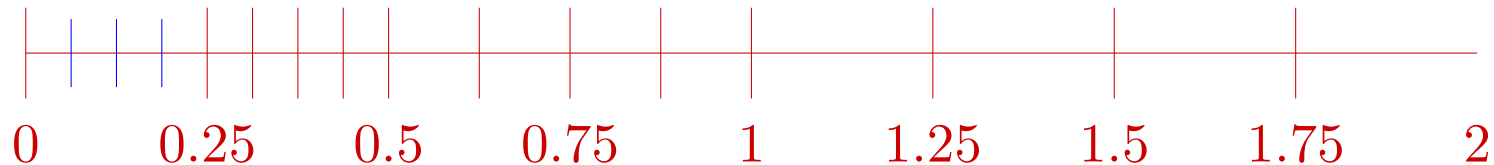
$$x = \pm(c_1b^{-1} + c_2b^{-2} + \dots + c_tb^{-t}) \cdot b^e$$

Npr. $0.1101 \cdot 2^2 = (2^{-1} + 2^{-2} + 2^{-4}) \cdot 2^2 = 3.25$.

Zgled

Vsa normalizirana pozitivna predstavljiva števila iz množice $P(2, 3, -1, 1)$ so:

$0.100_2 \cdot 2^{-1} =$	0.2500	$0.100_2 \cdot 2^0 =$	0.500	$0.100_2 \cdot 2^1 =$	1.00
$0.101_2 \cdot 2^{-1} =$	0.3125	$0.101_2 \cdot 2^0 =$	0.625	$0.101_2 \cdot 2^1 =$	1.25
$0.110_2 \cdot 2^{-1} =$	0.3750	$0.110_2 \cdot 2^0 =$	0.750	$0.110_2 \cdot 2^1 =$	1.50
$0.111_2 \cdot 2^{-1} =$	0.4375	$0.111_2 \cdot 2^0 =$	0.875	$0.111_2 \cdot 2^1 =$	1.75

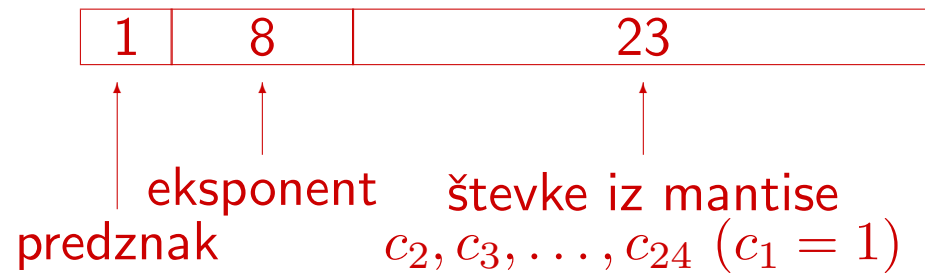


Subnormalizirana števila (možna le pri najmanjšem eksponentu) so:

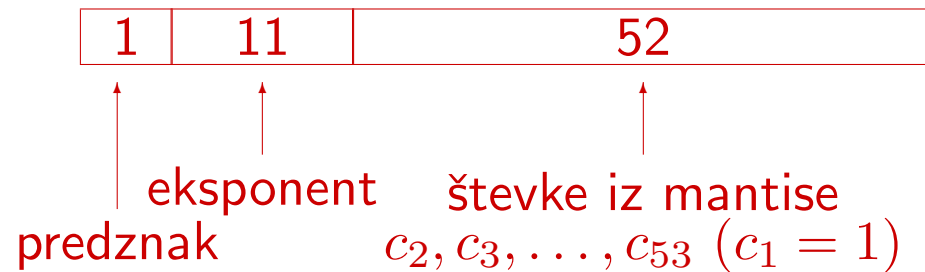
$0.011_2 \cdot 2^{-1} =$	0.1875
$0.010_2 \cdot 2^{-1} =$	0.1250
$0.001_2 \cdot 2^{-1} =$	0.0625

Standard IEEE

- *single*: $P(2, 24, -125, 128)$, število je shranjeno v 32 bitih,



- *double*: $P(2, 53, -1021, 1023)$, število je shranjeno v 64 bitih,



- standard IEEE pozna še števila 0, ∞ , $-\infty$ in NaN.

Zaokrožanje

Naj bo x pozitivno število z neskončnim zapisom

$$x = 0.d_1d_2 \dots d_t d_{t+1} \dots \cdot b^e,$$

kjer je $L \leq e \leq U$. Kandidata za predstavljen približek, ki ga označimo s $\text{fl}(x)$, sta najbližji predstavljeni števili z leve in z desne

$$\begin{aligned}x_- &= 0.d_1d_2 \dots d_t \cdot b^e, \\x_+ &= (0.d_1d_2 \dots d_t + b^{-t}) \cdot b^e.\end{aligned}$$

Pri standardu IEEE uporabimo **zaokrožanje** in za $\text{fl}(x)$ izberemo predstavljlivo število, ki je najbližje x .

Osnovna zaokrožitvena napaka

Izrek 1. Če za število x velja, da $|x|$ leži na intervalu med najmanjšim in največjim pozitivnim predstavljamim normaliziranim številom, potem velja

$$\frac{|\text{fl}(x) - x|}{|x|} \leq u,$$

kjer je $u = \frac{1}{2} b^{1-t}$ osnovna zaokrožitvena napaka.

Velja

$$\text{fl}(x) = x(1 + \delta) \quad \text{za} \quad |\delta| \leq u$$

- single: $u = 2^{-24} = 6 \cdot 10^{-8}$,
- double: $u = 2^{-53} = 1 \cdot 10^{-16}$.

Računanje po standardu IEEE

Standard IEEE zagotavlja, da velja:

- $\text{fl}(x \oplus y) = (x \oplus y)(1 + \delta)$, $|\delta| \leq u$ za $\oplus = +, -, /, *$,
- $\text{fl}(\sqrt{x}) = \sqrt{x}(1 + \delta)$, $|\delta| \leq u$.

Izjema je, če pride do *prekoračitve* (overflow) ali *podkoračitve* (underflow) obsega predstavljivih števil. V tem primeru dobimo:

- prekoračitev: $\pm\infty$,
- podkoračitev: 0.

1.3 Vrste napak pri numeričnem računanju

Pri numeričnem računanju se pojavljajo tri vrste napak:

- **Neodstranljiva napaka:** pojavi se, ker začetni podatki niso točni,
- **Napaka metode:** pojavi se, ker metoda s katero računamo, ni točna,
- **Zaokrožitvena napaka:** pojavi se, ker moramo pri vseh vmesnih izračunih rezultat vedno zaokrožiti.

Izračun $\sin(\pi/10)$ z osnovnimi operacijami v $P(10, 4, -5, 5)$

točna vrednost je $\sin(\pi/10) = 0.309016994\dots$

1. Začetni podatek ni predstavljivo število.

Namesto z $x = \pi/10$ računamo z $\bar{x} = \text{fl}(\pi/10) = 0.3142 \cdot 10^0$

2. Kako z osnovnimi operacijami izračunamo vrednost funkcije $\sin$?

Namesto $\bar{y} = \sin(\bar{x})$ izračunamo $\tilde{y} = g(\bar{x})$ za $g(x) = x - x^3/6$.

3. Končni rezultat je odvisen od zaporedja operacij za izračun $g(\bar{x})$.

$$\begin{aligned} a_1 &= \text{fl}(\bar{x} \cdot \bar{x}) &= \text{fl}(0.09872164) &= 0.9872 \cdot 10^{-1} \\ a_2 &= \text{fl}(a_1 \cdot \bar{x}) &= \text{fl}(0.031017824) &= 0.3102 \cdot 10^{-1} \\ a_3 &= \text{fl}(a_2/6) &= \text{fl}(0.00517) &= 0.5170 \cdot 10^{-2} \\ \hat{y} &= \text{fl}(\bar{x} - a_3) &= \text{fl}(0.30903) &= 0.3090 \cdot 10^0 \end{aligned}$$

Absolutna napaka končnega približka je $D = y - \hat{y} = 1.7 \cdot 10^{-5}$.

Relativna napaka končnega približka je $D/y = 5.5 \cdot 10^{-5}$.

Osnovna zaokrožitvena napaka je $5 \cdot 10^{-4}$.

Celotno napako lahko razdelimo na tri dele

Neodstranljiva napaka: Namesto z x računamo s približkom $\bar{x}$ in namesto $y = f(x)$ izračunamo $\bar{y} = f(\bar{x})$. Neodstranljiva napaka je $D_n = y - \bar{y}$.

D_n je posledica napak začetnih podatkov.

Zgled: *Računanje $\sin(\pi/10)$ z osnovnimi operacijami v $P(10, 4, -5, 5)$*

Namesto z $x = \pi/10$ računamo z $\bar{x} = 0.3142 \cdot 10^0$

$$D_n = y - \bar{y} = \sin(\pi/10) - \sin(0.3142) = -3.9 \cdot 10^{-5}$$

Napaka metode: Namesto f računamo vrednost funkcije g , ki jo lahko izračunamo s končnim številom operacij. Namesto $\bar{y} = f(\bar{x})$ tako izračunamo $\tilde{y} = g(\bar{x})$. Napaka metode je $D_m = \bar{y} - \tilde{y}$.

Zgled: *Namesto $\sin(\bar{x})$ izračunamo $g(\bar{x})$ za $g(x) = x - x^3/6$.*

$$D_m = \bar{y} - \tilde{y} = 2.5 \cdot 10^{-5}$$

Celotna napaka

Zaokrožitvena napaka: Pri računanju $\tilde{y} = g(\bar{x})$ se pri vsaki računski operaciji pojavi zaokrožitvena napaka, tako da namesto $\tilde{y}$ izračunamo $\hat{y}$. Sama vrednost $\hat{y}$ je odvisna od vrstnega reda operacij in načina izračuna $g(\bar{x})$. Zaokrožitvena napaka je $D_z = \tilde{y} - \hat{y}$.

Zgled: D_z je odvisna je od vrstnega reda in načina računanja $g(\bar{x})$. Primer:

$$\begin{aligned} a_1 &= \text{fl}(\bar{x} \cdot \bar{x}) &= \text{fl}(0.09872164) &= 0.9872 \cdot 10^{-1} \\ a_2 &= \text{fl}(a_1 \cdot \bar{x}) &= \text{fl}(0.031017824) &= 0.3102 \cdot 10^{-1} \\ a_3 &= \text{fl}(a_2/6) &= \text{fl}(0.00517) &= 0.5170 \cdot 10^{-2} \\ \hat{y} &= \text{fl}(\bar{x} - a_3) &= \text{fl}(0.30903) &= 0.3090 \cdot 10^0 \end{aligned}$$

$$D_z = \tilde{y} - g(\bar{x}) = 3.0 \cdot 10^{-5}$$

Celotna napaka: Končna napaka je $D = D_n + D_m + D_z$. Velja

$$|D| \leq |D_n| + |D_m| + |D_z|.$$

Zgled: Celotna napaka je $D = D_n + D_m + D_z = 1.7 \cdot 10^{-5}$

1.4 Občutljivost problema

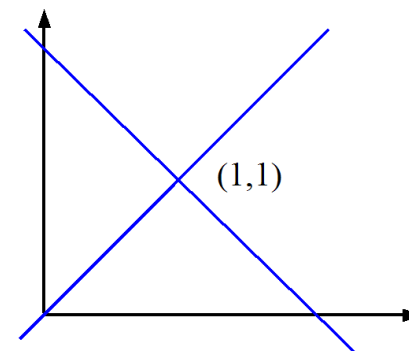
Če se rezultat pri majhni spremembi argumentov (*motnji* oz. *perturbaciji*) ne spremeni veliko, je problem *neobčutljiv*, sicer pa je *občutljiv*.

$$\text{a) } \begin{array}{l} x + y = 2 \\ x - y = 0 \end{array} \implies x = y = 1.$$

Zmotimo desno stran:

$$\begin{array}{l} x + y = 1.9999 \\ x - y = 0.0002 \end{array} \implies x = 1.00005, y = 0.99985.$$

Ta sistem je neobčutljiv.

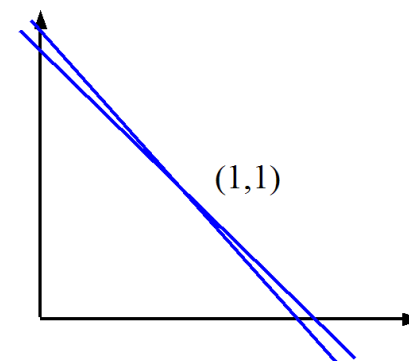


$$\text{b) } \begin{array}{l} x + 0.99y = 1.99 \\ 0.99x + 0.98y = 1.97 \end{array} \implies x = y = 1.$$

Zmotimo desno stran:

$$\begin{array}{l} x + 0.99y = 1.9899 \\ 0.99x + 0.98y = 1.9701 \end{array} \implies x = 2.97, y = -0.99.$$

Ta sistem je zelo občutljiv.



Wilkinsonov zgled

Polinom

$$p(x) = (x - 1)(x - 2) \cdots (x - 20) = x^{20} - 210x^{19} + \cdots + 20!$$

ima enostavne in lepo separirane ničle $1, 2, \dots, 20$, polinom

$$g(x) = p(x) - 2^{-23}x^{19}$$

pa ima ničle

$$x_1 = 1.00000$$

$$x_9 = 8.91725$$

$$x_2 = 2.00000$$

$$x_{10,11} = 10.09527 \pm 0.64350i$$

$$x_3 = 3.00000$$

$$x_{12,13} = 11.79363 \pm 1.65233i$$

$$x_4 = 4.00000$$

$$x_{14,15} = 13.99236 \pm 2.51883i$$

$$x_5 = 5.00000$$

$$x_{16,17} = 16.73074 \pm 2.81262i$$

$$x_6 = 6.00001$$

$$x_{18,19} = 19.50244 \pm 1.94033i$$

$$x_7 = 6.99970$$

$$x_{20} = 20.84691.$$

$$x_8 = 8.00727$$

Majhna motnja povzroči velike spremembe.

Wilkinson je s tem primerom pokazal, da računanje lastnih vrednosti matrike preko karakterističnega polinoma ni stabilno.

Stopnja občutljivosti

Stopnjo občutljivosti merimo z razmerjem med velikostjo spremembe rezultata in velikostjo spremembe podatkov.

Zgled: Naj bo $f : \mathbb{R} \rightarrow \mathbb{R}$ zvezna in odvedljiva funkcija. Zanima nas razlika med $f(x)$ in $f(x + \delta x)$, kjer je δx majhna motnja.

Velja

$$|f(x + \delta x) - f(x)| \approx |f'(x)| \cdot |\delta x|,$$

torej je $|f'(x)|$ *absolutna občutljivost* f v točki x .

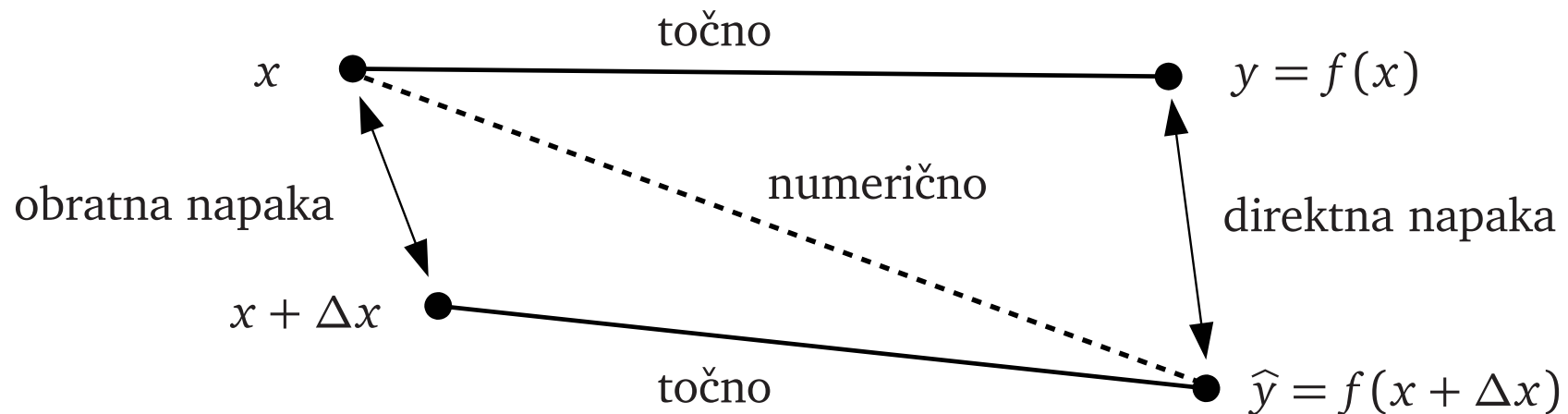
Za oceno relativne napake dobimo

$$\frac{|f(x + \delta x) - f(x)|}{|f(x)|} \approx \frac{|f'(x)| \cdot |x|}{|f(x)|} \cdot \frac{|\delta x|}{|x|},$$

torej je $\frac{|f'(x)| \cdot |x|}{|f(x)|}$ *relativna občutljivost* f v točki x .

1.5 Obratna in direktna stabilnost metode

Metoda iz x namesto $y = f(x)$ izračuna $\hat{y}$



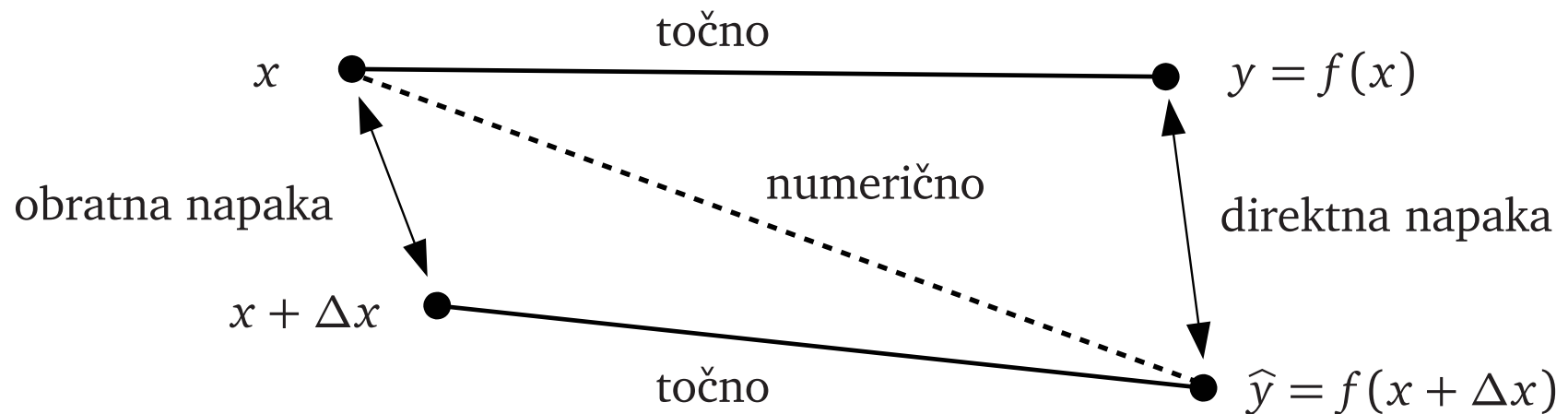
- **direktna stabilnost:** Če je za vsak x razlika med y in $\hat{y}$ majhna (absolutno oz. relativno), je metoda direktno stabilna (absolutno oz. relativno).

Direktno stabilna metoda vedno vrne bližnji odgovor.

- **obratna stabilnost:** Vprašamo se, za koliko moramo spremeniti argument x v $x + \Delta x$, da bo $f(x + \Delta x) = \hat{y}$. Če je za vsak x razlika Δx majhna (absolutno oz. relativno), je metoda obratno stabilna (absolutno oz. relativno).

Obratno stabilna metoda vedno vrne odgovor na bližnje vprašanje.

Povezava med občutljivostjo, direktno in obratno napako



Direktna napaka: $\hat{y} - y$

Obratna napaka: Δx

Groba ocena:

$$|\text{direktna napaka}| \lesssim \text{občutljivost} \times |\text{obratna napaka}|.$$

Natančnost rezultata

Če je problem občutljiv, se točna rešitev bližnjega problema lahko zelo razlikuje od točne rešitve začetnega problema in izračunani rezultat je nenatančen.

Če rešimo neobčutljiv problem z obratno stabilno metodo, se izračunani rezultat ne razlikuje dosti od točnega.

Nenatančnost je lahko posledica:

- uporabe stabilnega algoritma na občutljivem problemu,
- uporabe nestabilnega algoritma.

Natančnost je zagotovljena, če neobčutljiv problem rešimo s stabilno numerično metodo.

1.5.1 Analiza zaokrožitvenih napak za produkt $n + 1$ števil

Računamo produkt $p = x_0 x_1 \cdots x_n$ predstavljivih števil $x_0, x_1, \dots, x_n$.

Eksaktni algoritem:

$$p_0 = x_0$$

$$i = 1, \dots, n$$

$$p_i = p_{i-1} x_i$$

$$p = p_n$$

Dejanski algoritem:

$$\hat{p}_0 = x_0$$

$$i = 1, \dots, n$$

$$\hat{p}_i = \hat{p}_{i-1} x_i (1 + \delta_i), \quad |\delta_i| \leq u$$

$$\hat{p} = \hat{p}_n$$

Dobimo

$$\hat{p} = p(1 + \gamma) = p(1 + \delta_1) \cdots (1 + \delta_n).$$

Velja

$$(1 - u)^n \leq 1 + \gamma \leq (1 + u)^n.$$

Ocenimo

$$(1 + u)^n = 1 + \binom{n}{1} u + \binom{n}{2} u^2 + \cdots = 1 + nu + \mathcal{O}(u^2),$$

$$(1 - u)^n \geq 1 - nu. \quad (\text{indukcija})$$

Če je $nu \ll 1$ ocenimo $|\gamma| \leq nu$. Računanje produkta $n + 1$ števil je direktno in obratno stabilno.

1.5.2 Analiza zaokrožitvenih napak za skalarni produkt

Imamo dva vektorja predstavljenih števil $x = (x_1 \cdots x_n)^T$ in $y = (y_1 \cdots y_n)^T$, računamo pa $s = y^T x = \sum_{i=1}^n x_i y_i$.

Eksaktni algoritem:

$$s_0 = 0$$

$$i = 1, \dots, n$$

$$p_i = x_i y_i$$

$$s_i = s_{i-1} + p_i$$

$$s = s_n$$

Dejanski algoritem:

$$\hat{s}_0 = 0$$

$$i = 1, \dots, n$$

$$\hat{p}_i = x_i y_i (1 + \alpha_i), \quad |\alpha_i| \leq u$$

$$\hat{s}_i = (\hat{s}_{i-1} + \hat{p}_i)(1 + \beta_i), \quad |\beta_i| \leq u$$

$$\hat{s} = \hat{s}_n$$

Obratna analiza vrne $\hat{s} = \sum_{i=1}^n x_i y_i (1 + \gamma_i)$, kjer je

$$1 + \gamma_1 = (1 + \alpha_1)(1 + \beta_2) \cdots (1 + \beta_n)$$

in

$$1 + \gamma_i = (1 + \alpha_i)(1 + \beta_i) \cdots (1 + \beta_n), \quad i = 2, \dots, n.$$

Tako dobimo ocene $|\gamma_1| \leq nu$ in $|\gamma_i| \leq (n - i + 2)u$ za $i = 2, \dots, n$. To pomeni, da je $\hat{s}$ točni skalarni produkt relativno malo zmotenih vektorjev x in y . Računanje skalarnega produkta je obratno stabilno.

Direktna analiza za izračun skalarnega produkta

Pri direktni analizi najprej izračunamo absolutno napako:

$$\hat{s} - s = \sum_{i=1}^n x_i y_i \gamma_i,$$

torej

$$|\hat{s} - s| \leq \sum_{i=1}^n |x_i| \cdot |y_i| \cdot |\gamma_i| \leq nu \sum_{i=1}^n |x_i| \cdot |y_i| = nu |y|^T |x|.$$

Dobimo

$$\left| \frac{\hat{s} - s}{s} \right| \leq \frac{|y|^T |x|}{|y^T x|} nu.$$

Če so vsi $x_i y_i$ enakega predznaka, dobimo $\left| \frac{\hat{s} - s}{s} \right| \leq nu$ in računanje je direktno stabilno, sicer pa imamo v primeru, ko sta vektorja skoraj pravokotna, lahko veliko relativno napako.

V splošnem torej računanje skalarnega produkta **ni direktno stabilno**.

1.5.3 Računanje vrednosti polinoma po Hornerju

Dan je polinom $p(x) = a_0x^n + a_1x^{n-1} + \dots + a_n$, računamo pa vrednost v točki x po Hornerju.

Eksaktni algoritem je:

$$p_0 = a_0$$

$$i = 1, \dots, n$$

$$p_i = p_{i-1}x + a_i$$

$$p = p_n$$

Dejanski algoritem pa:

$$\hat{p}_0 = a_0$$

$$i = 1, \dots, n$$

$$\hat{p}_i = (\hat{p}_{i-1}x(1 + \alpha_i) + a_i)(1 + \beta_i)$$

$$\hat{p} = \hat{p}_n$$

Dobimo

$$\hat{p} = a_0x^n(1 + \gamma_0) + a_1x^{n-1}(1 + \gamma_1) + \dots + a_n(1 + \gamma_n),$$

kjer lahko ocenimo $|\gamma_i| \leq 2nu$ za $i = 0, \dots, n$.

Računanje vrednosti polinoma je obratno stabilno, saj smo izračunali vrednost bližnjega polinoma s koeficienti $a_i(1 + \gamma_i)$ namesto a_i .

Iz absolutne napake $\hat{p} - p = a_0x^n\gamma_0 + a_1x^{n-1}\gamma_1 + \dots + a_n\gamma_n$ sledi

$$\frac{|\hat{p} - p|}{|p|} \leq \frac{2nu(|a_0||x^n| + |a_1||x^{n-1}| + \dots + |a_n|)}{|a_0x^n + \dots + a_n|}.$$

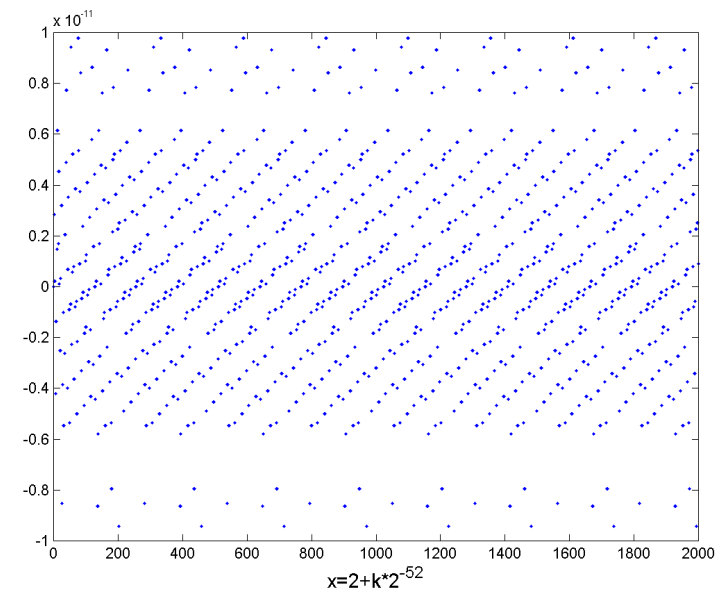
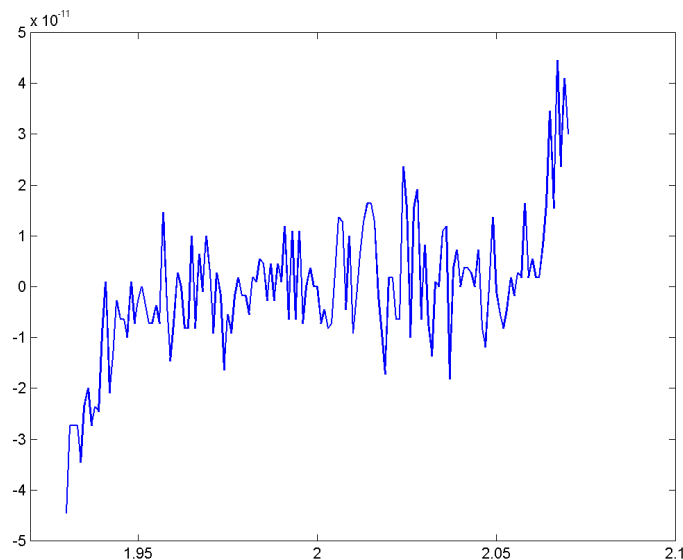
Računanje vrednosti polinoma po Hornerju ni direktno stabilno.

Računanje vrednosti polinoma po Hornerjevem algoritmu - 2.del

Za predstavo o dobljenih ocenah si pogledjmo računanje polinoma

$$(x-2)^9 = x^9 - 18x^8 + 144x^7 - 672x^6 + 2016x^5 - 4032x^4 + 5376x^3 - 4608x^2 + 2304x - 512$$

v okolici točke 2. Z Matlabom dobimo



Napake niso povsem naključne, saj v bližini 2 odkrijemo vzorec.

Računanje vrednosti polinoma po Hornerjevem algoritmu - 3.del

V algoritem lahko vstavimo sprotno računanje ocene:

$$p_0 = a_0$$

$$e_0 = |a_0|$$

$$i = 1, \dots, n$$

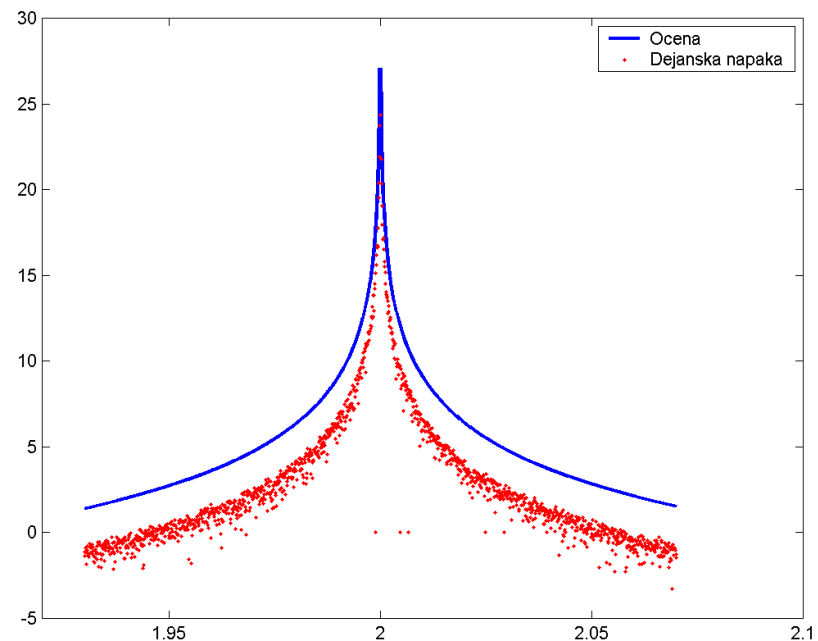
$$p_i = p_{i-1}x + a_i$$

$$e_i = e_{i-1}|x| + |a_i|$$

$$p = p_n$$

$$e = 2nu \frac{e_n}{|p|}$$

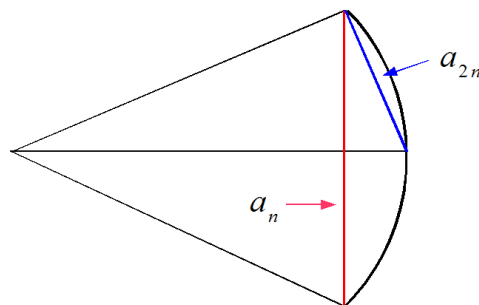
e je ocena za relativno napako.



Graf prikazuje razmerje med resnično napako in oceno (minus desetiški logaritem nam da ravno število točnih decimal). Ocena je res zgornja meja in dejanska napaka je manjša ali enaka, vendar zgled kaže, da so v bližini točke 2 napake res večje in tudi obnašajo se tako, kot napoveduje ocena.

1.6.1 Poučni primeri - računanje števila π

π je limita obsega S_n pravilnega mnogokotnika, včrtanega v krog s polmerom $r = \frac{1}{2}$. Naj bo a_n stranica pravilnega n -kotnika. Poiščimo zvezo med a_n in a_{2n} :



Velja

$$a_{2n} = \sqrt{\left(\frac{a_n}{2}\right)^2 + \left(\frac{1}{2} - \sqrt{\frac{1}{4} - \left(\frac{a_n}{2}\right)^2}\right)^2} = \sqrt{\frac{1 - \sqrt{1 - a_n^2}}{2}},$$

od tod pa iz $S_n = na_n$ sledi

$$S_{2n} = 2na_{2n} = 2n\sqrt{\frac{1 - \sqrt{1 - \left(\frac{S_n}{n}\right)^2}}{2}}.$$

Računanje števila π , 2.del

Začnemo pri $S_6 = 3$ in uporabljamo formulo $S_{2n} = 2n \sqrt{\frac{1 - \sqrt{1 - \left(\frac{S_n}{n}\right)^2}}{2}}$.

n	S_n	n	S_n
6	3.0000000	768	3.1430728
12	3.1058285	1536	3.1486604
24	3.1326292	3072	3.1374750
48	3.1393456	6144	3.1819806
96	3.1410186	12288	3.0000000
192	3.1414995	24576	4.2426405
384	3.1416743	49152	0.0000000

Formula odpove, saj pride do odštevanja skoraj enako velikih števil, napaka pa se množi z $2n$. V zgornji tabeli (enojna natančnost) so napačne decimalke rdeče.

Kadar imamo nestabilen postopek, nam ne pomaga niti računanje z večjo natančnostjo. Prava rešitev je preurediti postopek tako, da se med računanjem ne izgublja natančnost.

Računanje števila π , 3.del

Za stabilno računanje je potrebno formulo preurediti. Stabilna oblika je

$$S_{2n} = 2n \sqrt{\frac{\left(1 - \sqrt{1 - \left(\frac{S_n}{n}\right)^2}\right) \left(1 + \sqrt{1 - \left(\frac{S_n}{n}\right)^2}\right)}{2 \left(1 + \sqrt{1 - \left(\frac{S_n}{n}\right)^2}\right)}} = S_n \sqrt{\frac{2}{1 + \sqrt{1 - \left(\frac{S_n}{n}\right)^2}}}.$$

Sedaj dobimo pravilne rezultate:

n	S_n	n	S_n
6	3.0000000	768	3.1415839
12	3.1058285	1536	3.1375904
24	3.1326287	3072	3.1415920
48	3.1393502	6144	3.1415925
96	3.1410320	12288	3.1415925
192	3.1414526	24576	3.1415925
384	3.1415577	49152	3.1415925