1 Numerično računanje

1.1 Predstavitev števil

Predstavljivo število x v sistemu $P(b, t, L, U)$ je zapisano kot $x = \pm m \cdot b^e$, kjer je b baza, e eksponent v mejah $L \leq e \leq U$ in

$$m = 0.c_1 \dots c_t, \qquad 0 \leq c_i \leq b-1, \quad i = 1, \dots, t,$$

mantisa. Pri tem zahtevamo, da je $c_1 \neq 0$, razen kadar je $e = L$. Predstavljiva števila s $c_1 \neq 0$ imenujemo normalizirana, ostala so denormalizirana.

Naloga 1.1. Zapišite vsa normalizirana števila v sistemu $P(2, 3, -1, 3)$. Katera ležijo na intervalu $(0, 1)$? Koliko je denormaliziranih števil?

Rešitev. Normalizirana števila v sistemu $P(2, 3, -1, 3)$ so

$$\pm 0.100_2 \cdot 2^e \quad \pm 0.101_2 \cdot 2^e, \quad \pm 0.110_2 \cdot 2^e, \quad \pm 0.111_2 \cdot 2^e$$

za $e \in \{-1, 0, 1, 2, 3\}$ oziroma

± 0.2500,	± 0.5000,	± 1.0000,	± 2.0000,	± 4.0000,
± 0.3125,	± 0.6250,	± 1.2500,	± 2.5000,	± 5.0000,
± 0.3750,	± 0.7500,	± 1.5000,	± 3.0000,	± 6.0000,
± 0.4375,	± 0.8750,	± 1.7500,	± 3.5000,	± 7.0000

v desetiškem zapisu. Na intervalu $(0, 1)$ ležijo števila s pozitivnim predznakom pri $e = -1$ in $e = 0$. Denormalizirana števila so določena z mantisami 0.001, 0.010 in 0.011 ter najmanjšim eksponentom $e = -1$. Torej jih je vsega skupaj šest (tri pozitivna in tri negativna).

Naloga 1.2. V Matlabu generirajte vsa predstavljiva števila iz množice $P(5, 4, -5, 5)$ in jih uredite po velikosti od najmanjšega do največjega. Nato poiščite odgovore na spodnja vprašanja.

1. Kakšen je delež denormaliziranih števil?
2. Koliko normaliziranih števil je manjših od π?
3. Kakšen je povprečni razmik med zaporednimi predstavljivimi števili, ki se od π absolutno razlikujejo za manj kot 1?

Rešitev. Najprej sestavimo program, ki izračuna seznam predstavljivih (`X`), normaliziranih (`Xn`) in denormaliziranih (`Xdn`) števil v danem sistemu.

```
% sistem
b = 5; t = 4; L = -5; U = 5;

% mantise
c = 0:b-1;
M = zeros(b^t,1);
```

```
i = 1;
for c1 = c
    for c2 = c
        for c3 = c
            for c4 = c
                M(i) = (b.^-(1:t))*[c1; c2; c3; c4];
                i = i+1;
            end
        end
    end
end

% normalizirana števila
d = U-L+1;
bm = b^(t-1);
Xpn = zeros((b-1)*bm, d);
for i = 0:d-1
    Xpn(:,i+1) = M(bm+1:end) * b^(L+i);
end
Xpn = Xpn(:);
Xn = [-Xpn(end:-1:1); Xpn];

% denormalizirana števila
Xpdn = M(2:b^(t-1)) * b^L;
Xdn = [-Xpdn(end:-1:1); Xpdn];

% predstavljiva števila (brez 0, Inf, -Inf in NaN)
X = [Xn(1:end/2); Xdn(1:end/2); Xpdn; Xpn];
```

Za vajo poskusite začetni del zgornjega programa nadgraditi tako, da izračuna vse mantise splošne dolžine `t`.

1. Delež denormaliziranih števil izračunamo tako, da njihovo število delimo s številom vseh predstavljivih števil. Rezultat je približno 2.2%.
2. Število normaliziranih števil manjših od π lahko preštejemo z ukazom `sum(Xn<pi)`. Dobimo 8768.
3. Da dobimo predstavljiva števila, ki se od π razlikujejo za manj kot ena, uporabimo ukaz `S = X(abs(X-pi)<1)`. Nato uporabimo vgrajeni funkciji `mean` in `diff`, da izračunamo povprečni razmik `mean(diff(S))`, ki je enak 0.008.

Število x, ki ni vsebovano v danem sistemu $P(b,t,L,U)$, je nepredstavljivo. Nadomestimo ga s številom $\mathrm{fl}(x)$, ki je bodisi največje predstavljivo število, manjše od x, bodisi najmanjše predstavljivo število, večje od x. Število $\mathrm{fl}(x)$ navadno določimo z zaokroževanjem x. S tem zagotovimo, da ob pogoju, da $|x|$ leži na intervalu med najmanjšim in največjim pozitivnim predstavljivim številom, velja $\mathrm{fl}(x) = x(1+\delta)$, kjer je δ število z lastnostjo, da je $|\delta|$ manjša od osnovne zaokrožitvene napake $u = b^{1-t}/2$.

Naloga 1.3. Katero je največje število v množici $P(5, 4, -5, 5)$, ki je manjše od π, in katero je najmanjše število, ki je večje od π? Katero izmed teh dveh števil je $\mathrm{fl}(\pi)$?

Rešitev. S pomočjo programa iz naloge 1.2 lahko odgovore na vprašanja poiščemo s spodnjimi ukazi.

```
x = pi;                                % 3.1416

xl = X(find(X<pi,1,'last'));           % 3.1360
xf = X(find(X>pi,1,'first'));          % 3.1440

if xf-x <= x-xl
    flx = xl;
else
    flx = xf;                          % 3.1440
end
```

Naloga 1.4. Predstavite število $x = 47.712$ v dvojiškem zapisu in z zaokroževanjem poiščite njegovo najbližje predstavljivo število $\mathrm{fl}(x)$ v sistemu $P(2, 9, -10, 10)$. Preverite, da je relativna napaka $|\mathrm{fl}(x) - x| \,/\, |x|$ manjša od osnovne zaokrožitvene napake.

Rešitev. Dvojiški zapis celega oziroma decimalnega dela x dobimo z deljenjem oziroma z množenjem z 2,

$$\begin{aligned}
47 &= 23 \cdot 2 + 1, & 0.712 \cdot 2 &= 0.424 + 1,\\
23 &= 11 \cdot 2 + 1, & 0.424 \cdot 2 &= 0.848 + 0,\\
11 &= 5 \cdot 2 + 1, & 0.848 \cdot 2 &= 0.696 + 1,\\
5 &= 2 \cdot 2 + 1, & 0.696 \cdot 2 &= 0.392 + 1,\\
2 &= 1 \cdot 2 + 0, & 0.392 \cdot 2 &= 0.784 + 0,\\
1 &= 0 \cdot 2 + 1, & 0.784 \cdot 2 &= 0.568 + 1, \ldots .
\end{aligned}$$

Od tod sledi $47 = 101111_2$ (ostanke v levem stolpcu prepišemo od spodaj navzgor) in $0.712 = 0.101101\ldots_2$ (celi del v desnem stolpcu prepišemo od zgoraj navzdol). Torej je

$$x = 0.101111101101\ldots_2 \cdot 2^6 \quad \text{in} \quad \mathrm{fl}(x) = 0.101111110_2 \cdot 2^6.$$

Ker je

$$\begin{aligned}
|\mathrm{fl}(x) - x| &= \left|(0.101111101_2 + 2^{-9}) - (0.101111101_2 + 1.01\ldots_2 \cdot 2^{-10})\right| \cdot 2^6\\
&< \left|2^{-9} - 2^{-10}\right| \cdot 2^6\\
&= 2^{-4},
\end{aligned}$$

je relativna napaka $|\mathrm{fl}(x) - x| \,/\, |x|$ manjša od 0.0014, kar je manj od osnovne zaokrožitvene napake $2^{1-9}/2 \approx 0.0020$.

Pravimo, da je število x zapisano v enojni natančnosti, če je predstavljeno s številom $\mathrm{fl}(x)$ iz množice $P(2, 24, -125, 128)$. V računalniškem spominu je tako število shranjeno v 32 bitih. Če je normalizirano, je podano v obliki

$$\mathrm{fl}(x) = (-1)^s (1 + f) \cdot 2^{\widetilde{e}-127},$$

kjer $s \in \{0,1\}$ določa predznak (en bit), $\widetilde{e} \in \{1,2,\ldots,2^8-1\}$ eksponent (osem bitov) in $f = 0.c_2\,c_3\ldots c_{24}$ del mantise (23 bitov). Na podoben način so s 64 biti opisana števila iz $P(2,53,-1021,1024)$, ki določajo dvojno natančnost.

Naloga 1.5. Dokažite, da je

$$0.1 = \sum_{i=1}^{\infty} \left(2^{-4i} + 2^{-4i-1}\right)$$

in določite $\mathrm{fl}(0.1)$ za 0.1 v enojni natančnosti. Kako je to število v tem formatu predstavljeno v računalniku?

Rešitev. Vrsto izračunamo s prevedbo na geometrijsko vrsto

$$\sum_{i=1}^{\infty} \left(2^{-4i} + 2^{-4i-1}\right) = \left(1+\frac{1}{2}\right) \sum_{i=1}^{\infty} \left(2^{-4}\right)^i = \frac{3}{2} \cdot \frac{2^{-4}}{1-2^{-4}} = \frac{1}{10}$$

in s tem dokažemo, da lahko število 0.1 predstavimo v želeni obliki. Iz tega rezultata sledi, da je $0.1 = 0.0\overline{0011}_2$. Ker ima 0.1 v dvojiški bazi neskončen decimalni zapis, $\mathrm{fl}(0.1)$ dobimo z zaokroževanjem. Na podlagi

$$0.1 = 0.1100110011001100110011001\ldots_2 \cdot 2^{-3}$$

sklepamo, da je

$$\mathrm{fl}(0.1) = 0.11001100110011001100\mathbf{1}101_2 \cdot 2^{-3}$$

oziroma

$$\mathrm{fl}(0.1) = (-1)^0(1 + 0.10011001100110011001\mathbf{1}01_2) \cdot 2^{123-127},$$

kar pomeni, da število $\mathrm{fl}(0.1)$ opišemo z biti 0, 01111011 in 10011001100110011001101, ki po vrsti določajo s, $\widetilde{e}$ in f. Rezultat v Matlabu preverimo s pomočjo ukaza `single(0.1)`, ki vrne $\mathrm{fl}(0.1)$ za enojno natančnost.

```
x = 0.1;
flx = (repmat([1 1 0 0],1,6)+[zeros(1,23) 1])*2.^-(4:27)';
single(x)-x     % 1.4901161e-09
single(x)-flx   % 0
```

1.2 Napake pri računanju

Pri numerični matematiki smo soočeni z napakami v različnih fazah računanja.

1. Navadno pride do napake že pri pripravi vhodnih podatkov na začetku računanja. Napaka, ki je razlika med izvedbo računa s pravimi in dejanskimi podatki, se imenuje *neodstranljiva napaka*.

2. Pri reševanju problema smo se zaradi njegove težavnosti ali računske zahtevnosti pogosto primorani sprijazniti z njegovim približnim reševanjem. Tako namesto originalnega problema rešimo njegov bližnji problem in napaka, ki pri tem nastane, se imenuje *napaka metode*.

3. Nazadnje moramo v zakup vzeti še *zaokrožitveno napako*, ki je posledica zaokroževanja na vsakem računskem koraku izvedbe metode, saj rezultat vsake računske operacije zaokrožujemo na najbližje predstavljivo število.

Seštevek vseh treh napak je celotna napaka izračuna.

Naloga 1.6. Funkcija f je podana s predpisom $f(x) = \sqrt{1+x}$. Izračunajte vrednost $f(x)$ za $x = 1/13$ v sistemu $P(10, 5, -10, 10)$.

1. Ocenite neodstranljivo napako, ki nastane pri predstavitvi x.
2. Namesto funkcije f uporabite Taylorjev polinom funkcije f stopnje 2, ki ga dobite z razvojem okoli točke 0. Ocenite napako metode.
3. Vrednost Taylorjevega polinoma izračunajte s Hornerjevim postopkom. S pomočjo izračuna vrednosti v dvojni natančnosti ocenite zaokrožitveno napako, ki nastane zaradi računanja v dani aritmetiki.

Rešitev. Ocenimo vsako izmed napak, ki se pojavi pri izvedbi postopka.

1. Najprej ocenimo neodstranljivo napako, ki nastane zaradi predstavitve x v predpisanem sistemu. Ker je $x = 0.0769230...$, je $\overline{x} = \mathrm{fl}(x) = 0.76923 \cdot 10^{-1}$. Neodstranljiva napaka D_n je podana z $D_n = f(x) - f(\overline{x})$. Njeno absolutno vrednost lahko s pomočjo izreka o povprečni vrednosti in ocene za relativno napako predstavitve x z $\overline{x}$ v dani aritmetiki ocenimo z

$$|D_n| = |f(x) - f(\overline{x})| \leq \max_{\xi \in [0,1]} |f'(\xi)|\, |x - \overline{x}| < 0.5 \cdot 10^{1-5}/2 = 0.25 \cdot 10^{-4}.$$

2. Napaka metode nastane, ker namesto s funkcijo f računamo s približkom, ki ga dobimo s pomočjo razvoja f v Taylorjevo vrsto. Konkretno, funkcijo f zamenjamo s polinomom $g(x) = 1 + x/2 - x^2/8$. Napaka metode je podana z $D_m = f(\overline{x}) - g(\overline{x})$, njeno absolutno vrednost pa lahko ocenimo z

$$|D_m| = |f(\overline{x}) - g(\overline{x})| \leq \frac{1}{3!} \max_{\xi \in [0,1]} |f'''(\xi)|\, \overline{x}^3 < \overline{x}^3/16 < 0.29 \cdot 10^{-4}.$$

3. Označimo $g(x) = a_0 + a_1 x + a_2 x^2$. Računanje polinoma g v točki $\overline{x}$ s Hornerjevim postopkom

$$b_2 = a_2, \qquad b_i = b_{i+1}\overline{x} + a_i, \quad i = 1, 0, \qquad g(\overline{x}) = b_0,$$

v predpisanem sistemu poteka na sledeč način.

i	a_i	$b_{i+1} \cdot \overline{x}$	$c_i = \mathrm{fl}(b_{i+1} \cdot \overline{x})$	$c_i + a_i$	$b_i = \mathrm{fl}(a_i + c_i)$
2	-0.125				$-0.12500 \cdot 10^0$
1	0.5	-0.009615375	$0.96154 \cdot 10^{-2}$	0.4903846	$0.49038 \cdot 10^0$
0	1	0.03772150074	$0.37722 \cdot 10^{-1}$	1.037722	$0.10377 \cdot 10^1$

Z računanjem v dvojni natančnosti dobimo, da je $g(\overline{x})$ približno 1.0377219, torej je zaokrožitvena napaka D_z po absolutni vrednosti manjša od $0.22 \cdot 10^{-4}$.

Iz obravnave napak sledi, da je celotna napaka manjša od 10^{-4}.

Naloga 1.7. Dani sta diferenčni enačbi

$$a_n = \tfrac{5}{2}a_{n-1} - a_{n-2}, \qquad n = 2, 3, \ldots, \qquad a_0 = 1,\ a_1 = \tfrac{1}{2},$$
$$b_n = \tfrac{10}{3}b_{n-1} - b_{n-2}, \qquad n = 2, 3, \ldots, \qquad b_0 = 1,\ b_1 = \tfrac{1}{3}.$$

1. Z nastavkoma $a_n = \lambda^n$, $\lambda \in \mathbb{R}$, in $b_n = \mu^n$, $\mu \in \mathbb{R}$, poiščite točni rešitvi diferenčnih enačb.
2. V Matlabu generirajte seznama $\boldsymbol{a} = (a_0, a_1, \ldots, a_{50})$ in $\boldsymbol{b} = (b_0, b_1, \ldots, b_{50})$ ter z ukazom `scatter` narišite točke (n, a_n) in (n, b_n), $n = 0, 1, \ldots, 50$. Ali se elementi seznamov ujemajo s točnimi vrednostmi? Pojasnite, zakaj da oziroma ne.
3. Omilite napake, ki nastanejo pri izračunu elementov v seznamu $\boldsymbol{b}$ tako, da elemente generirate v obratnem vrstnem redu pri začetnih podatkih $b_{50} = 0$ in $b_{49} = 1$ ter jih na koncu skalirate s konstanto, ki zagotovi, da bo $b_0 = 1$. Primerjajte dobljene vrednosti s točnimi.

Rešitev.

1. Uporabimo nastavka za $a_n = \lambda^n$ in $b_n = \mu^n$. Ker sta enačbi dvočlenski, dobimo v obeh primerih kvadratni enačbi z rešitvama $\lambda_1 = 1/2$, $\lambda_2 = 2$ in $\mu_1 = 1/3$, $\mu_2 = 3$. Od tod sledi, da sta splošni rešitvi oblike

$$a_n = A\left(\tfrac{1}{2}\right)^n + B\,2^n, \qquad b_n = C\left(\tfrac{1}{3}\right)^n + D\,3^n,$$

   kjer so A, B, C, D konstante, ki jih določimo iz začetnih pogojev. Dobimo $a_n = 1/2^n$ in $b_n = 1/3^n$. Vrednosti a_n in b_n z naraščajočim n torej padata proti 0.

2. Elemente seznamov izračunamo na podlagi rekurzivnih formul, ki določata diferenčno enačbo.

```
% seznam a
a = [1 0.5 zeros(1,49)];
for n = 3:51
    a(n) = 5*a(n-1)/2 - a(n-2);
end

% seznam b
b = [1 1/3 zeros(1,49)];
for n = 3:51
    b(n) = 10*b(n-1)/3 - b(n-2);
end
```

   Iz diagramov na slikah 1a in 1b je razvidno, da vrednosti a_n padajo proti 0, kot je pričakovano z obzirom na točno rešitev diferenčne enačbe. Po drugi strani pa so izračunane vrednosti b_n pri večjih n povsem napačne (rastejo v pozitivno ali negativno smer, odvisno od vrstnega reda operacij pri generiranju seznama $\boldsymbol{b}$). Razlog, da v prvem primeru dobimo točne rezultate, v drugem pa napačne, se skriva v tem, da je v prvem primeru rezultat vsake računske operacije predstavljivo število v dvojni

natančnosti, medtem ko v drugem primeru operiramo z nepredstavljivimi števili, ki jih vseskozi zaokrožujemo na predstavljiva. Tako že za začetni podatek namesto $b_1 = 1/3$ uporabimo $\widetilde{b}_1 = \mathrm{fl}(b_1) = b_1(1+\delta)$, kjer je δ sicer po absolutni vrednosti majhno število, a točna rešitev $\widetilde{b}_n$ diferenčne enačbe z začetnima podatkoma b_0 in $\widetilde{b}_1$ je

$$\widetilde{b}_n = \left(1 - \tfrac{\delta}{8}\right)\left(\tfrac{1}{3}\right)^n + \tfrac{\delta}{8}3^n$$

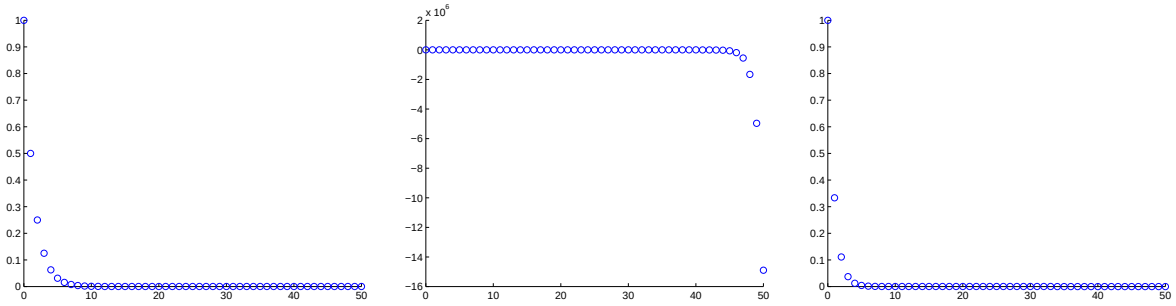
in vpliv faktorja 3^n se pri večjih vrednostih n močno pozna.

3. Iz rekurzivne zveze za vrednosti b_n izrazimo b_{n-2} in z zamikom indeksov dobimo

$$b_n = -b_{n+2} + \tfrac{10}{3}b_{n+1}.$$

Vzamemo $b_{50} = 0$ in $b_{49} = 1$ ter elemente seznama $\boldsymbol{b}$ generiramo v obratnem vrstnem redu. Na koncu vse elemente seznama delimo z b_0 in s tem zagotovimo, da je v rezultatu $b_0 = 1$. Vrednosti tega seznama se zelo dobro ujemajo s točnimi, kar potrjuje diagram na sliki 1c.

```
% seznama b generiran v obrantem vrstnem redu
rb = [zeros(1,49) 1 0];
for n = 49:-1:1
    rb(n) = -rb(n+2) + 10*rb(n+1)/3;
end
b = rb/rb(1);
```



(a) (b) (c)

Slika 1: Prikaz rezultatov pri rekurzivnem računanju števil v nalogi 1.7.

1.3 Stabilnost izračunov

Pri numeričnem računanju stabilnost obravnavamo v različnih kontekstih. V osnovi nas zanima razlika med točno vrednostjo in izračunanim približkom: to je *direktna napaka*. Če je ta pri vseh začetnih podatkih majhna, pravimo, da je metoda izračuna direktno stabilna. Ocenjevanje direktne napake je ponavadi težavno, zato si pri analizi pomagamo z *obratno napako*: to je razlika med dejanskimi vhodnimi podatki in vhodnimi podatki, pri katerih bi izračunani približek pri dejanskih podatkih predstavljal točno vrednost. Če je obratna napaka majhna pri vseh vhodnih podatkih, je metoda izračuna obratno stabilna. Z obratno stabilnostjo lahko dokažemo direktno stabilnost metode, kadar je problem neobčutljiv.

Naloga 1.8. Naj bosta x in y predstavljivi realni števili. Vrednost $z = x^2 - y^2$ izračunajte po standardu IEEE na dva načina:

1. $z = x^2 - y^2$,
2. $z = (x - y)(x + y)$.

V obeh primerih ocenite relativno napako ter obravnavajte direktno in obratno stabilnost izračuna. Predpostavite, da pri računanju ne pride do prekoračitev.

Rešitev. V prvem primeru lahko izračun interpretiramo kot računanje skalarnega produkta vektorjev (x, y) in $(x, -y)$, zato direktna stabilnost ni zagotovljena. V drugem primeru imamo produkt dveh predstavljivih števil in z obzirom na to pričakujemo, da je izračun direktno stabilen. Utemeljimo obe opazki bolj natančno.

1. Pri kvadriranju členov x in y dobimo $a_1 = x^2(1 + \alpha_1)$ in $a_2 = y^2(1 + \alpha_2)$, pri čemer je $|\alpha_1| \leq u$ in $|\alpha_2| \leq u$ ter u označuje osnovno zaokrožitveno napako. Ko člena odštejemo, namesto vrednosti z dobimo

$$\overline{z} = (a_1 - a_2)(1 + \beta) = x^2(1 + \alpha_1)(1 + \beta) - y^2(1 + \alpha_2)(1 + \beta)$$

   za $|\beta| \leq u$ oziroma

$$\overline{z} = x^2(1 + \delta_1) - y^2(1 + \delta_2)$$

   za neki konstanti δ_1 in δ_2 z lastnostjo

$$(1 - u)^2 \leq 1 + \delta_i \leq (1 + u)^2, \quad i = 1, 2.$$

   Ker je vrednost u majhna, lahko na podlagi tega ocenimo, da je $|\delta_i| \leq 2u$. Vrednost $\overline{z} = (x\sqrt{1 + \delta_1})^2 + (y\sqrt{1 + \delta_2})^2$ torej ustreza vrednosti z pri malo zmotenih podatkih, zato je izračun obratno stabilen. Po drugi strani pa direktna stabilnost ni zagotovljena, saj je

$$\frac{|\overline{z} - z|}{|z|} = \frac{|\delta_1 x^2 - \delta_2 y^2|}{|x^2 - y^2|}$$

   in v primeru, ko sta δ_1 in δ_2 nasprotno predznačeni, velja

$$\frac{|\overline{z} - z|}{|z|} \geq \min\{|\delta_1|, |\delta_2|\} \frac{x^2 + y^2}{|x^2 - y^2|}.$$

   Torej je lahko relativna napaka za $x \approx y$ velika.

2. Namesto vsote oziroma razlike števil x in y izračunamo $(x+y)(1+\alpha_1)$ in $(x-y)(1+\alpha_2)$, pri čemer sta vrednosti $|\alpha_1|$ in $|\alpha_2|$ manjši ali enaki osnovni zaokrožitveni napaki u. Z množenjem teh dveh členov namesto z dobimo

$$\overline{z} = (x + y)(x - y)(1 + \alpha_1)(1 + \alpha_2)(1 + \beta)$$

   za $|\beta| \leq u$. Sledi, da je $\overline{z} = (x\sqrt{1 + \delta})^2 - (y\sqrt{1 + \delta})^2$ za neko število δ z lastnostjo $|\delta| \leq 3u$, kar pomeni, da je izračun obratno stabilen. Poleg tega pa je tudi direktno stabilen, saj je

$$\frac{|\overline{z} - z|}{|z|} \leq \frac{|(x^2 - y^2)\delta|}{|x^2 - y^2|} = |\delta| \leq 3u.$$

Naloga 1.9. Dan je polinom

$$p(x) = a_n x^n + a_{n-1}x^{n-1} + \ldots + a_1 x + a_0 = a_n(x - x_n)(x - x_{n-1}) \ldots (x - x_1),$$

kjer so koeficienti $a_n, a_{n-1}, \ldots, a_1, a_0$ in ničle $x_n, x_{n-1}, \ldots, x_1$ predstavljiva realna števila. Obravnavajte relativni napaki pri računanju vrednosti polinoma p v točki x s Hornerjevim postopkom ter z množenjem faktorjev, ki jih določajo ničle polinoma. Ali je kateri izmed postopkov direktno stabilen?

Rešitev. Pri Hornerjevem postopku izračun vrednosti polinoma p poteka v zaporedju

$$p(x) = ((\ldots((a_n x + a_{n-1})\, x + a_{n-2})\, x + \ldots)\, x + a_1)\, x + a_0.$$

Pri vsaki operaciji seštevanja in množenja pride do zaokrožitvene napake, ki je relativno manjša od osnovne zaokrožitvene napake u. Zato namesto $y = p(x)$ izračunamo

$$\widehat{y} = a_n x^n(1 + \delta_n) + a_{n-1}x^{n-1}(1 + \delta_{n-1}) + \ldots + a_1 x(1 + \delta_1) + a_0(1 + \delta_0),$$

kjer za števila δ_i, $i = 0, 1, \ldots, n$, ocenjujemo, da so po absolutni vrednosti manjša od $2nu$ (pri vrednostih δ_i, $i = 0, 1, \ldots, n-1$, smo lahko natančnejši, velja $|\delta_i| \leq (2i + 1)u$). To pomeni, da za relativno napako izračuna velja

$$\frac{|\widehat{y} - y|}{|y|} \leq \frac{2nu\,(|a_n|\,|x^n| + |a_{n-1}|\,|x^{n-1}| + \ldots + |a_1|\,|x| + |a_0|)}{|a_n x^n + a_{n-1}x^{n-1} + \ldots + a_1 x + a_0|}.$$

Čeprav smo napako omejili navzgor in je lahko ta ocena pregroba, vseeno jasno odraža, da je za točke x blizu ničle polinoma relativna napaka lahko velika, če so koeficienti polinoma različno predznačeni. Postopek izračuna torej ni direktno stabilen.

Če obstaja faktorizacija polinoma na linearne faktorje, lahko izračun vrednosti opravimo stabilneje z množenjem faktorjev. Najprej izračunamo $x - x_i$, $i = 1, 2, \ldots, n$, pri čemer zaradi zaokroževanja dobimo $(x - x_i)(1 + \alpha_i)$ za neka števila α_i, ki so po absolutni vrednosti manjša od u. Nato vsako izmed n množenj botruje še k relativni napaki, manjši od u. Torej namesto $y = p(x)$ izračunamo

$$\widetilde{y} = a_n(x - x_n) \ldots (x - x_1)(1 + \delta) = y(1 + \delta),$$

pri čemer je $|\delta| \leq 2nu$. Torej je relativna napaka

$$\frac{|\widetilde{y} - y|}{|y|} = \frac{|y\delta|}{|y|} = |\delta| \leq 2nu$$

in izračun je direktno stabilen.

2 Reševanje nelinearnih enačb

2.1 Navadna iteracija

Eden izmed splošnih pristopov za iskanje ničle funkcije f oziroma reševanje nelinearne enačbe $f(x) = 0$ je prepis enačbe v ekvivalentno obliko $x = g(x)$. Funkcijo g imenujemo iteracijska funkcija, saj rešitev enačbe iščemo z iteracijo

$$x_{r+1} = g(x_r), \quad r = 0, 1, \ldots.$$

Če je g na določenem intervalu, ki vsebuje x_0, skrčitev, zaporedje $(x_r)_r$ po Banachovem skrčitvenem načelu konvergira k negibni točki g, ki ustreza ničli funkcije f.

Naloga 2.1. Funkcija f je podana s predpisom $f(x) = x^5 - 10x + 1$. Njeno ničlo iščemo z iteracijsko funkcijo $g(x) = (x^5 + 1)/10$.

1. Utemeljite, da ima funkcija f na intervalu $[0, 0.2]$ natanko eno ničlo.
2. Dokažite, da začetni približek $x_0 = 0$ zagotavlja konvergenco iteracijskega zaporedja $x_{r+1} = g(x_r)$ k ničli funkcije f na intervalu $[0, 0.2]$.
3. Ocenite, v koliko prvih decimalkah se približek $x_2 = g(g(0))$ ujema z ničlo funkcije f na intervalu $[0, 0.2]$.

Rešitev.

1. Funkcija f je za $x = 0$ enaka 1, za $x = 0.2$ pa $0.2^5 - 1 < 0$. Ker je zvezna, ima na intervalu $[0, 0.2]$ vsaj eno ničlo. Poleg tega je funkcija f na intervalu $[0, 0.2]$ padajoča, saj je $f'(x) < 0$ za vsak x s tega intervala. To potrjuje, da je ničla ena sama.
2. Odvod iteracijske funkcije g je enak $x^4/2$, zato je po absolutni vrednosti manjši od 1 natanko tedaj, ko je $|x| < \sqrt[4]{2}$. To zagotavlja konvergenco za vsak začetni približek z intervala $(-\sqrt[4]{2}, \sqrt[4]{2}) \approx (-1.1892, 1.1892)$.
3. Naj bo $\alpha \in (0, 0.2)$ ničla funkcije f ($f(\alpha) = 0$) oziroma negibna točka funkcije g ($g(\alpha) = \alpha$). Opazimo, da za vsak $x \in [0, \alpha]$ velja

   $$g(x) - \alpha \leq \frac{\alpha^5 + 1}{10} - \alpha = \frac{f(\alpha)}{10} = 0,$$

   kar pomeni, da se vsi členi iteracijskega zaporedja nahajajo na intervalu $[0, \alpha]$. Po izreku o povprečni vrednosti lahko razliko med zaporednima približkoma x_r in x_{r+1}, $r \in \mathbb{N}$, ocenimo z

   $$|x_{r+1} - x_r| = |g(x_r) - g(x_{r-1})| < g'(0.2)\,|x_r - x_{r-1}| = 8 \cdot 10^{-4}\,|x_r - x_{r-1}|\,.$$

   Ker začnemo iteracijo z $x_0 = 0$, v naslednjih dveh korakih dobimo $x_1 = 1/10$ in $x_2 = 1/10 + 1/10^6$. Ocenimo

   $$|x_2 - \alpha| \leq |x_2 - x_3| + |x_3 - x_4| + \ldots < \left(8 \cdot 10^{-4} + \left(8 \cdot 10^{-4}\right)^2 + \ldots\right) |x_1 - x_2|$$

   in od tod sklepamo

   $$|x_2 - \alpha| < \frac{8 \cdot 10^{-4}}{1 - 8 \cdot 10^{-4}} 10^{-6} \approx 8 \cdot 10^{-10}.$$

   To pomeni, da se x_2 z ničlo funkcije f gotovo ujema v prvih devetih decimalkah.

Naloga 2.2. Iteracijska funkcija g je podana s predpisom $g(x) = -x^2 + 8x - 12$.

1. Določite negibne točke iteracije $g(x) = x$ in opredelite, katere so privlačne in katere odbojne.
2. Za katere začetne približke v okolici negibnih točk konvergenčni izrek zagotavlja konvergenco?
3. Določite, za katere začetne približke je navadna iteracija konvergentna. Kam konvergirajo zaporedja?

Rešitev.

1. Negibne točke dobimo z reševanjem kvadratne enačbe $g(x) = x$. Negibni točki sta torej dve, prva je 3, druga pa 4. Z odvajanjem iteracijske funkcije dobimo, da je $g'(3) = 2$ in $g'(4) = 0$, kar pomeni, da je 3 odbojna, 4 pa privlačna točka.
2. Konvergenčni izrek zagotavlja konvergenco v okolici negibne točke $x = 4$ za začetne približke x_0, za katere velja $|g'(x_0)| < 1$. To je res natanko tedaj, ko je $|-2x_0 + 8| < 1$ oziroma $x_0 \in (3.5, 4.5)$.
3. Na podlagi grafa iteracijske funkcije g in simetrale lihih kvadrantov domnevamo, da iteracija konvergira k 4 pri začetnih približkih z intervala $(3, 5)$, pri vseh drugih začetnih približki pa divergira ali obstane v 3. Opazimo, da za približek x_r na r-tem koraku iteracije velja

$$x_r - 4 = g(x_{r-1}) - 4 = -(x_{r-1} - 4)^2 = \ldots = -(x_0 - 4)^{2^r},$$

kar potrjuje, da za vsak $x_0 \in (3, 5)$ velja $\lim_{r\to\infty} x_r = 4$. Od tod sledi tudi, da se za vsak $x_0 \in (-\infty, 3) \cup (5, \infty)$ približki x_r zmanjšujejo brez meje, pri začetnih približkih $x_0 = 3$ in $x_0 = 5$ pa velja $x_r = 3$ za vsak $r \in \mathbb{N}$.

Naloga 2.3. V Matlabu sestavite funkcijo, ki izvede navadno iteracijo.

1. Vhodni podatki naj bodo iteracijska funkcija, začetni približek, toleranca in maksimalno število korakov.
2. Izhodni podatki naj bodo izračunan približek po končani iteraciji, seznam vseh približkov tekom iteracije in število opravljenih korakov.
3. Funkcija naj izvaja iteracijo, dokler se zadnja dva približka absolutno ne razlikujeta za manj, kot je predpisana toleranca, ali število korakov ne preseže danega maksimalnega števila korakov.

Za test implementacije uporabite iteracijsko funkcijo iz naloge 2.2.

Rešitev. Funkcijo, ki izvede navadno iteracijo, poimenujemo `iteracija`.

```
function [x,X,k] = iteracija(g,x0,tol,N)
% funkcija
%  [x,X,k] = iteracija(g,x0,tol,N)
% izvede navadno iteracijo z dano iteracijsko funkcijo in
% začetnim približkom.
%
```

```
% Vhodni podatki:
%  g    iteracijska funkcija,
%  x0   začetni približek,
%  tol  toleranca absolutnega ujemanja dveh zaporednih
%       približkov,
%  N    maksimalno število korakov iteracije
%
% Izhodni podatki:
%  x    zadnji približek izračunan z navadno iteracijo,
%  X    seznam vseh izračunanih približkov,
%  k    število opravljenih korakov iteracije

X = x0;
k = 0;
while (k == 0 || abs(X(end)-X(end-1)) >= tol) && k < N
    X(k+2) = g(X(k+1));
    k = k+1;
end
x = X(k+1);

end
```

Test kaže, da se računski rezultati ujemajo s teoretičnimi izpeljavami iz naloge 2.2.

```
g = @(x)-x^2+8*x-12; tol = 1e-10; N = 1e3;
[x,~,k] = iteracija(g,3.5,tol,N)     % x = 4, k = 7
[x,~,k] = iteracija(g,4.5,tol,N)     % x = 4, k = 7
[x,~,k] = iteracija(g,3,tol,N)       % x = 3, k = 1
[x,~,k] = iteracija(g,2,tol,N)       % x = -Inf, k = 1000
```

V praksi je pomembno, kako hitro iteracijsko zaporedje $(x_r)_r$ konvergira k negibni točki α. Pravimo, da je red konvergence enak p, če obstaja taka konstanta $C > 0$, da je

$$\lim_{r\to\infty} \frac{|x_{r+1} - \alpha|}{|x_r - \alpha|^p} = C.$$

Če je funkcija g dovoljkrat zvezno odvedljiva, lahko red p enostavno določimo z odvajanjem: veljati mora $g^{(k)}(\alpha) = 0$, $k = 1, 2, \ldots, p-1$, in $g^{(p)}(\alpha) \neq 0$ ob dodatni predpostavki, da je $|g'(\alpha)| < 1$, če je $p = 1$.

Naloga 2.4. Za iskanje ničle funkcije $f(x) = x^2 - x - 2$ uporabite štiri različne iteracijske funkcije:

1. $g_1(x) = x^2 - 2$,
2. $g_2(x) = \sqrt{x+2}$,
3. $g_3(x) = 1 + 2/x$,
4. $g_4(x) = (x^2+2)/(2x-1)$.

Za vsako funkcijo analizirajte konvergenco v okolici ničel -1 in 2 ter določite njen red. Ugotovitve preverite v Matlabu s pomočjo funkcije, implementirane v nalogi 2.3. Narišite grafe števila korakov iteracije v odvisnosti od začetnih približkov na intervalu $[-2, 4]$.

Rešitev. Premislimo, kako se iteracijske funkcije obnašajo v okolici ničel funkcije f.

1. Ker je $g_1'(x) = 2x$, sta tako -1 kot 2 odbojni negibni točki iteracijske funkcije g_1. Ničlo funkcije f torej dobimo le v posebnih primerih, ko za začetni približek izberemo -2, -1, 0, 1 ali 2.

2. Iz $g_2'(x) = 1/\sqrt{x+2}$ sledi, da je 2 privlačna negibna točka, v okolici katere je red konvergence enak 1. Enostavno je dokazati, da iteracijsko zaporedje konvergira k 2 za vsak začetni približek, ki je večji ali enak -2. Na drugi strani -1 ni negibna točka funkcije g_2, zato te ničle funkcije f z g_2 ne moremo poiskati.

3. Najprej opazimo, da sta negibni točki funkcije g_3 tako -1 kot 2, vendar iz $g_3'(-1) = -2$ in $g_3'(2) = -1/2$ sklepamo, da je le 2 privlačna negibna točka. Za približke v okolici -1 torej ni pričakovati konvergence k -1, v -1 z iteracijo končamo le, če v -1 z njo začnemo. Po drugi strani vrednost odvoda g_3 v točki 2 zagotavlja konvergenco k -2 za začetne približke v okolici -2. S podrobnejšo analizo iteracijske funkcije lahko dokažemo, da iteracijsko zaporedje konvergira k -2 za vse začetne približke, razen -1. Pri začetnih približkih 0 in -2 v prvem oziroma drugem koraku iteracije delimo z 0.

4. Funkcija g_4 ima negibni točki -1 in 2, v obeh pa je vrednost odvoda g_4 enaka 0. Red konvergence v okolici -1 in 2 je torej vsaj reda 2 in za začetne približke blizu negibnih točk se lahko nadejamo hitre konvergence k eni ali drugi ničli funkcije f. Iz analize vrednost $g_4(x) - x$ sledi, da iteracijsko zaporedje konvergira k 2 za začetni približek večji od 1/2 in k -1 za začetni približek manjši od 1/2.

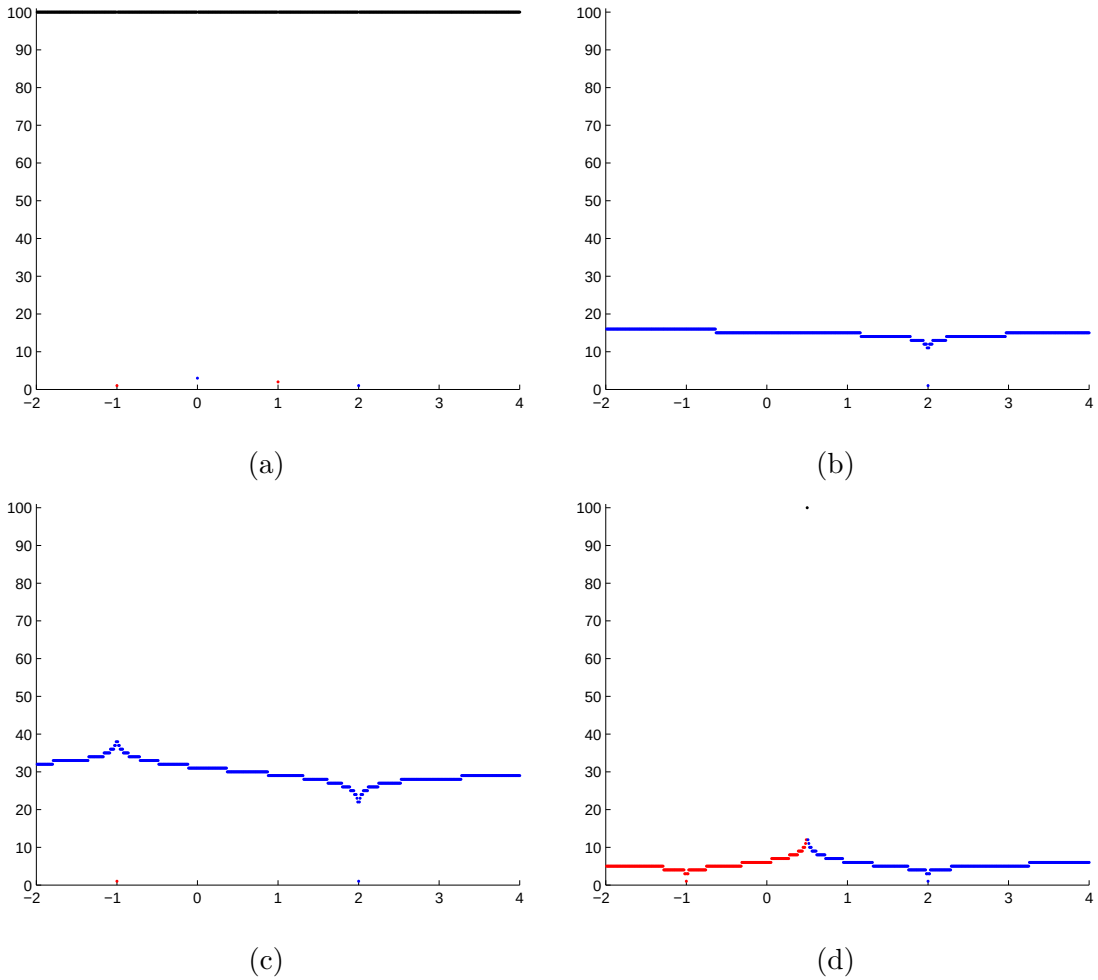
V Matlabu preverimo, kakšne rezultate dobimo z uporabo iteracijskih funkcij pri začetnih približkih $-1/2$ in 3. Iteracijo izvedemo s pomočjo funkcije `iteracija` iz naloge 2.3 pri absolutni toleranci `tol = 1e-10` in maksimalnem številu korakov `N = 100`.

```
g1 = @(x) x.^2-2;
g2 = @(x) sqrt(x+2);
g3 = @(x) 1+2./x;
g4 = @(x) (x.^2+2)./(2*x-1);

tol = 1e-10; N = 100;

[x,~,k] = iteracija(g1,-0.5,tol,N)  % x = 0.5914,  k = 100
[x,~,k] = iteracija(g1,3,tol,N)     % x = Inf,     k = 100
[x,~,k] = iteracija(g2,-0.5,tol,N)  % x = 2,       k = 19
[x,~,k] = iteracija(g2,3,tol,N)     % x = 2,       k = 18
[x,~,k] = iteracija(g3,-0.5,tol,N)  % x = 2,       k = 39
[x,~,k] = iteracija(g3,3,tol,N)     % x = 2,       k = 35
[x,~,k] = iteracija(g4,-0.5,tol,N)  % x = -1,      k = 5
[x,~,k] = iteracija(g4,3,tol,N)     % x = 2,       k = 6
```

Na sliki 2 so prikazani grafi števila korakov iteracij pri začetnih približkih iz seznama `linspace(-2,4,601)`. Modra točka označuje konvergenco k ničli 2, rdeča točka pa konvergenco k ničli -1. Črna točka pomeni, da se je iteracija končala brez konvergence po maksimalnem številu korakov. Slika 2a prikazuje rezultate za iteracijsko funkcijo g_1, kjer v eni izmed ničel končamo le pri začetnih približkih -2, -1, 0, 1 in 2. Na sliki 2b je prikazano število korakov iteracij pri funkcji g_2, ki ima za negibno točko ničlo 2. Število korakov, ki so potrebni za dosego tolerance, je manjše kot pri iteracijski funkciji g_3, kot kaže slika 2c. Iz te slike je tudi jasno razvidno, da je 1 odbojna točka iteracije z g_3, saj iteracijsko zaporedje le pri začetnem približku -1 konvergira k -1. Iteracijska zaporedja najhitreje konvergirajo pri iteracijski funkciji g_4. Slika 2d kaže, da je limita za začetne približke manjše od 1/2 enaka -1, za začetne približke večje od 1/2 pa 2. Pri začetnem približku 1/2 iteracijsko zaporedje ne konvergira k nobeni izmed ničel.



Slika 2: Število korakov v odvisnosti od začetnih približkov pri iteracijah iz naloge 2.4.

Naloga 2.5. Dokažite, da lahko kvadratni koren pozitivnega realnega števila a izračunamo z iteracijo

$$x_{r+1} = x_r \frac{x_r^2 + 3a}{3x_r^2 + a}$$

pri poljubnem začetnem približku $x_0 > 0$ ter določite red konvergence iteracijskega zaporedja v bližini $\sqrt{a}$.

Rešitev. Enostavno je preveriti, da je funkcija $g(x) = x(x^2 + 3a)/(3x^2 + a)$ iteracijska funkcija za enačbo $x(x^2 - a) = 0$. Negibne točke iteracije so $-\sqrt{a}$, 0 in $\sqrt{a}$.

Radi bi dokazali, da iteracija za $x_0 > 0$ konvergira k negibni točki $\sqrt{a}$. Najprej opazimo, da za vsak približek x_r, $r \in \mathbb{N}$, velja

$$x_r - \sqrt{a} = g(x_{r-1}) - \sqrt{a} = \frac{(x_{r-1} - \sqrt{a})^3}{3x_{r-1}^2 + a}.$$

Od tod po indukciji sledi, da je za začetni približek $x_0 < \sqrt{a}$ vsak približek x_r manjši od $\sqrt{a}$. Podobno, če je $x_0 > \sqrt{a}$, je tudi vsak približek x_r večji od $\sqrt{a}$. Nadalje, ker je

$$x_{r+1} - x_r = g(x_r) - x_r = \frac{2x_r(a - x_r^2)}{3x_r^2 + a},$$

za $x_r < \sqrt{a}$ velja $x_{r+1} > x_r$, za $x_r > \sqrt{a}$ pa $x_{r+1} < x_r$. To dokazuje, da je zaporedje približkov $(x_r)_{r=0}^{\infty}$ pri začetnem približku $x_0 > 0$ konvergentno, saj je bodisi strogo naraščajoče in navzgor omejeno s $\sqrt{a}$ bodisi strogo padajoče in navzdol omejeno s $\sqrt{a}$. Naj bo limita zaporedja označena z α. Zanjo velja

$$\alpha = \lim_{r\to\infty} x_r = \lim_{r\to\infty} x_{r+1} = \lim_{r\to\infty} g(x_r) = g(\lim_{r\to\infty} x_r) = g(\alpha),$$

zato je negibna točka funkcije g. Ker je $\sqrt{a}$ edina neigbna točka g na intervalu $(0, \infty)$, je $\alpha = \sqrt{a}$.

Red konvergence določimo z odvajanje. Ker je

$$g'(x) = \frac{3(x^2 - a)^2}{(3x^2 + a)^2}, \qquad g''(x) = \frac{48xa}{(3x^2 + a)^3}(x^2 - a),$$

je $g'(\sqrt{a}) = g''(\sqrt{a}) = 0$ in red konvergence je vsaj kubičen. Red v resnici je kubičen, saj je $g'''(\sqrt{a}) \neq 0$, kot sledi iz

$$g'''(x) = \left(\frac{48xa}{(3x^2 + a)^3}\right)'(x^2 - a) + \frac{48xa}{(3x^2 + a)^3}2x.$$

Naloga 2.6. Določite parametre α, β in γ v iterativni formuli

$$x_{r+1} = \alpha x_r + \beta\frac{a}{x_r^2} + \gamma\frac{a^2}{x_r^5}, \quad r = 0, 1, \ldots,$$

za računanje $\sqrt[3]{a}$ tako, da bo konvergenca iteracijskega zaporedja za začetni približek v okolici $\sqrt[3]{a}$ kubična.

Rešitev. Obravnavajmo iteracijsko funkcijo $g(x) = \alpha x + \beta a/x^2 + \gamma a^2/x^5$. Če želimo, da je $\sqrt[3]{a}$ negibna točka g, mora veljati $\alpha + \beta + \gamma = 1$. Da bo red konvergence vsaj kubičen, mora biti $g'(\sqrt[3]{a}) = 0$ in $g''(\sqrt[3]{a}) = 0$, kar je ekvivalentno zahtevama $\alpha - 2\beta - 5\gamma = 0$ in $6\beta + 30\gamma = 0$. Z reševanjem sistema enačb za dane parametre ugotovimo, da mora biti $\alpha = 5/9$, $\beta = 5/9$ in $\gamma = -1/9$. Ker je pri teh parametrih $g'''(\sqrt[3]{a}) = 10/\sqrt[3]{a^2} \neq 0$, je red konvergence kubičen.

2.2 Metode za reševanje nelinearnih enačb

Za iskanje ničel funkcije so na voljo številni recepti, po katerih lahko pripravimo ustrezno iteracijsko funkcijo za izvedbo navadne iteracije. Eden takih je tangentna (ali Newtonova oziroma Newton–Raphsonova) metoda. Približek x_{r+1} za ničlo odvedljive funkcije f dobimo iz približka x_r z zvezo

$$x_{r+1} = x_r - \frac{f(x_r)}{f'(x_r)}.$$

Geometrijsko x_{r+1} predstavlja presečišče abscisne osi in tangente funkcije f v točki x_r.

Naloga 2.7. Babilonska metoda za računanje kvadratnega korena $\sqrt{a}$ pozitivnega števila a temelji na iteraciji

$$x_{r+1} = \frac{1}{2}\left(x_r + \frac{a}{x_r}\right), \quad r = 0, 1, \ldots.$$

Za izvedbo enega koraka iteracije so potrebne le tri osnovne računske operacije.

1. Preverite, da babilonska metoda ustreza tangentni metodi za funkcijo $f(x) = x^2 - a$.
2. Kakšen je red konvergence babilonske metode v okolici negibne točke $\sqrt{a}$?
3. Dokažite, da zaporedje približkov $(x_r)_{r=0}^{\infty}$ pri babilonski metodi konvergira k $\sqrt{a}$ za vsak začetni približek $x_0 > 0$.

Rešitev.

1. V formulo, ki določa približek po tangentni metodi, vstavimo $f(x) = x^2 - a$. Z nekaj preurejanja dobimo babilonsko metodo.
2. Red konvergence babilonske metode v okolici $\sqrt{a}$ lahko določimo z odvajanjem iteracijske funkcije $g(x) = (x + a/x)/2$. Izračunamo

   $$g'(x) = \frac{1}{2}\left(1 - \frac{a}{x^2}\right), \qquad g''(x) = \frac{a}{x^3}$$

   in iz $g'(\sqrt{a}) = 0$ in $g''(\sqrt{a}) = 1/\sqrt{a} \neq 0$ sklepamo, da je red konvergence v okolici $\sqrt{a}$ kvadratičen.
3. Odvod funkcije g je po absolutni vrednosti manjši od 1 za vsak $x \in (\sqrt{a/3}, \infty)$, na podlagi česar lahko sklepamo, da zaporedje $(x_r)_{r=0}^{\infty}$ konvergira k $\sqrt{a}$ za vsak začetni približek x_0 s tega intervala. Potrebno je dokazati še, da to velja tudi v primeru, ko je $x_0 \in (0, \sqrt{a/3}]$. Opazimo, da je

   $$g(x_0) - \sqrt{a} = \frac{x_0}{2} + \frac{a}{2x_0} - \sqrt{a} = \frac{1}{2x_0}\left(x_0 - \sqrt{a}\right)^2,$$

   iz česar sledi, da za vsak začetni približek $x_0 > 0$ približek $x_1 = g(x_0)$ leži na intervalu $[\sqrt{a}, \infty)$; tu pa je g dokazano skrčitev, zato je limita iteracijskega zaporedja negibna točka $\sqrt{a}$.

Naloga 2.8. Naj bo f funkcija z m-kratno ničlo α. Če je ničla enostavna ($m = 1$), je red konvergence tangentne metode v njeni bližini vsaj kvadratičen. Kaj pa če je $m > 1$?

1. S pomočjo razvoja f v Taylorjevo vrsto okoli α dokažite, da za iteracijsko funkcijo $g(x) = x - f(x)/f'(x)$ tangentne metode velja

$$\lim_{x\to\alpha} g'(x) = 1 - \frac{1}{m}.$$

   Kaj lahko na podlagi tega sklepate o hitrosti konvergence tangentne metode?

2. Kako bi popravili tangentno metodo, da bi bil red konvergence v okolici m-kratne ničle vsaj kvadratičen?

Rešitev.

1. Limito funkcije

$$g'(x) = \left(x - \frac{f(x)}{f'(x)}\right)' = 1 - \frac{f'(x)^2 + f(x)f''(x)}{f'(x)^2} = \frac{f(x)f''(x)}{f'(x)^2}$$

   obravnavamo s pomočjo razvoja f v Taylorjevo vrsto okoli točke α. Ker je α m-kratna ničla f, je

$$f(x) = \frac{1}{m!}f^{(m)}(\alpha)(x-\alpha)^m + \mathcal{O}\left((x-\alpha)^{m+1}\right)$$

   in zato

$$f'(x) = \frac{1}{(m-1)!}f^{(m)}(\alpha)(x-\alpha)^{m-1} + \mathcal{O}\left((x-\alpha)^m\right),$$
$$f''(x) = \frac{1}{(m-2)!}f^{(m)}(\alpha)(x-\alpha)^{m-2} + \mathcal{O}\left((x-\alpha)^{m-1}\right).$$

   Izračunamo

$$g'(x) = \frac{\frac{1}{m!}\frac{1}{(m-2)!}\left(f^{(m)}(\alpha)\right)^2(x-\alpha)^{2m-2} + \mathcal{O}\left((x-\alpha)^{2m-1}\right)}{\frac{1}{(m-1)!^2}\left(f^{(m)}(\alpha)\right)^2(x-\alpha)^{2m-2} + \mathcal{O}\left((x-\alpha)^{2m-1}\right)}$$

   in od tod sklepamo, da je

$$\lim_{x\to\alpha} g'(x) = \frac{\frac{1}{m!}\frac{1}{(m-2)!}}{\frac{1}{(m-1)!^2}} = 1 - \frac{1}{m}.$$

   To pomeni, da je red konvergence v okolici večkratne ničle linearen in tudi, da je hitrost konvergence tem manjša, čim večja je kratnost ničle.

2. Funkcijo g bi radi zamenjali s sorodno funkcijo h, za katero velja $\lim_{x\to\alpha} h'(x) = 0$. Ker je $m\lim_{x\to\alpha} g'(x) = m - 1$, poskusimo s funkcijo

$$h(x) = x - m\frac{f(x)}{f'(x)}.$$

Njen odvod je

$$h'(x) = 1 - m\frac{f'(x)^2 + f(x)f''(x)}{f'(x)^2} = 1 - m + mg'(x)$$

in je v limiti, ko gre x proti α, enak 0. Seveda lahko metodo z iteracijsko funkcijo h izkoristimo le, če vnaprej poznamo kratnost ničle funkcije f.

Naloga 2.9. V odvisnosti od začetnih približkov analizirajte konvergenco tangentne metode za $f(x) = x^3 - x$.

1. Razvijte f v Taylorjevo vrsto okoli trenutnega približka in jo izvrednotite v 1. S pomočjo dobljene formule dokažite, da za vsak začetni približek iz $(1/\sqrt{3}, \infty)$ metoda konvergira k 1. S podobnim sklepom utemeljite, da za začetni približek iz $(-\infty, -1/\sqrt{3})$ metoda konvergira k -1.
2. Opišite, do kakšnih težav pride pri začetnih približkih $\pm 1/\sqrt{3}$ in $\pm 1/\sqrt{5}$.
3. Kako se obnaša zaporedje približkov pri začetnih približkih iz $(-1/\sqrt{5}, 1/\sqrt{5})$ in ali metoda konvergira k 0?
4. Ugotovite, kam konvergira zaporedje približkov pri začetnih pogojih z intervalov $(-1/\sqrt{3}, -1/\sqrt{5})$ in $(1/\sqrt{5}, 1/\sqrt{3})$?

Rešitev. Iteracijska funkcija je podana s predpisom

$$g(x) = x - \frac{x^3 - x}{3x^2 - 1} = \frac{2x^3}{3x^2 - 1}.$$

Vemo, da so negibne točke iteracije ničle funkcije f, to so -1, 0 in 1. Ker so enostavne ničle, vemo tudi, da je konvergenca v neki njihovi okolici vsaj kvadratična. Oglejmo si natančneje, kaj se dogaja z iteracijskimi zaporedji pri različnih začetnih približkih.

1. Iz razvoja f v Taylorjevo vrsto okoli x_{r-1} izvrednotenega v ničli 1 sledi, da je

$$x_r - 1 = \frac{f''(\xi_{r-1})}{2f'(x_{r-1})}(x_{r-1} - 1)^2, \qquad r \in \mathbb{N},$$

za nek ξ_{r-1} med 1 in x_{r-1}. Ker je f na intervalu $(1/\sqrt{3}, \infty)$ strogo naraščajoča in strogo konveksna, za vsak x_{r-1} s tega intervala velja $x_r > 1$ za vsak $r \in \mathbb{N}$. V posebnem to pomeni, da za začetni približek $x_0 \in (1/\sqrt{3}, \infty)$ velja $f(x_1) > 0$, zato iz definicije tangentne metode sledi $1 < x_2 < x_1$. Po indukciji lahko ta sklep nadaljujemo do ugotovitve, da je zaporedje $(x_r)_{r=1}^{\infty}$ strogo padajoče in navzdol omejeno z 1. Ker je 1 edina ničla funkcije f na intervalu $(1/\sqrt{3}, \infty)$, smo s tem dokazali, da za vsak začetni približek s tega intervala zaporedje $(x_r)_{r=0}^{\infty}$ konvergira k 1. Podoben razmislek pripelje do zaključka, da zaporedje $(x_r)_{r=0}^{\infty}$ konvergira k -1 za vsak začetni približek $x_0 \in (-\infty, -1/\sqrt{3})$.

2. Tangentna metoda pri začetnih približkih $x_0 = \pm 1/\sqrt{3}$ propade, saj sta $\pm 1/\sqrt{3}$ pola iteracijske funkcije g. Težave lahko pričakujemo tudi pri začetnih približkih $\pm 1/\sqrt{5}$, saj sta rešitvi enačbe

$$g(x_0) = \frac{2x_0^3}{3x_0^2 - 1} = -x_0.$$

To pomeni, da za prva dva približka velja $x_2 = -x_1 = x_0$, iz česar sklepamo, da začetna približka $\pm 1/\sqrt{5}$ povzročita cikel reda 2.

3. Obravnavajmo začetne približke z intervala $(-1/\sqrt{5}, 1/\sqrt{5})$. Prepričajmo se, da zagotavljajo konvergenco k 0. Najprej iz padanja iteracijske funkcije g na obravnavanem intervalu sklepamo, da je $g(x) \in (0, 1/\sqrt{5})$ za vsak $x \in (-1/\sqrt{5}, 0)$ in $g(x) \in (-1/\sqrt{5}, 0)$ za vsak $x \in (0, 1/\sqrt{5})$. Nato opazimo še, da je $g(x) + x < 0$ za vsak $x \in (-1/\sqrt{5}, 0)$ in $g(x) + x > 0$ za vsak $x \in (0, 1/\sqrt{5})$. Od tod sledi, da členi zaporedja alternirajoče menjavajo predznak. Če je $x_0 > 0$, zaporedje sodih členov pada proti 0, zaporedje lihih členov pa raste proti 0. Za $x_0 < 0$ je situacija ravno obratna, ne glede na to pa celotno zaporedje približkov konvergira k 0.

4. Za začetne približke z intervala $(-1/\sqrt{3}, -1/\sqrt{5})$ lahko s pomočjo grafa iteracijske funkcije g in simetrale lihih kvadrantov razberemo, da zaporedje približkov konvergira bodisi k -1 bodisi k 1, ki je najbolj oddaljena ničla. Pri začetnem približku $x_0 = -1/2$ je na primer $x_1 = 1$, zato je zaradi zveznosti g pri začetnih približkih blizu $-1/2$ približek x_1 blizu 1, kar implicira konvergenco k 1. Konvergenco k -1 v resnici dobimo le za začetne približke x_0 na majhnem odseku, kjer je $g(x_0) \in (1/\sqrt{5}, 1/\sqrt{3})$. Po simetriji podobno velja za začetne približke z intervala $(1/\sqrt{5}, 1/\sqrt{3})$. Zaključimo, da je tangentna metoda za določene začetne približke lahko zelo nepredvidljiva.

Naloga 2.10. Poiščite funkcijo f, za katero tangentna metoda propade za vsak začetni približek.

Rešitev. Poskušajmo določiti tako funkcijo f, da pri tangentni metodi za vsak začetni približek x_0 velja $x_r = -x_{r-1}$, $r \in \mathbb{N}$. To pomeni, da se tekom iteracije izmenjujeta približka x_0 in $-x_0$. Ker je $x_r = x_{r-1} - f(x_{r-1})/f'(x_{r-1})$, mora biti funkcija f rešitev navadne diferencialne enačbe

$$\frac{f'(x)}{f(x)} = \frac{1}{2x}$$

in zato zadošča

$$|f(x)| = C\sqrt{|x|}$$

za neko konstanto C. Vzemimo na primer

$$f(x) = \frac{x\sqrt{|x|}}{|x|}.$$

Ker je $f'(x) = 1/(2\sqrt{|x|})$, res velja $x_r = -x_{r-1}$ za vsak $r \in \mathbb{N}$.

Naloga 2.11. Implementirajte tangentno metodo v Matlabu in z njo poiščite ničlo funkcije f, ki je podana s predpisom $f(x) = x + 4 - e^{x^2}$. Koliko korakov iteracije je potrebno izvesti pri začetnem približku $x_0 = 1$, da se zadnji izračunani približek v 10 decimalkah ujema z vrednostjo približka, ki ga dobite z vgrajeno funkcijo `fzero`?

Rešitev. Pri pripravi tangentne metode se opremo na funkcijo `iteracija` iz naloge 2.3.

```
function [x,X,k] = tangentna(f,df,x0,tol,N)
% funkcija
%  [x,X,k] = tangentna(f,df,x0,tol,N)
```

```
% izvede tangentno metodo za iskanje ničle funkcije f
%
% Vhodni podatki:
%  f    funkcija, ničlo katere iščemo,
%  df   odvod funkcije f.
%
% Ostali vhodni in izhodni podatki so enaki kot pri
% funkciji 'iteracija'.

g = @(x) x - f(x)./df(x);
[x,X,k] = iteracija(g,x0,tol,N);

end
```

Seznam približkov, ki ga dobimo s klicem `[x,X,k] = tangentna(f,df,1,1e-15,100)` za primerno definirani funkciji `f` in `df`, primerjamo z rezultatom, dobljenim z ukazom `fzero(f,0)`. Ugotovimo, da je sedmi približek tangentne metode prvi, ki se z rezultatom vgrajene metode ujema v več kot desetih decimalkah (absolutna razlika je približno $2 \cdot 10^{-15}$).

Slaba lastnost tangentne metode je, da za njeno izvedbo poleg vrednosti funkcije potrebujemo tudi vrednosti njenega odvoda. Temu se lahko izognemo z uporabo sekantne metode, ki jo geometrijsko interpretiramo na podoben način kot tangentno metodo: novi približek x_{r+1} določimo kot presečišče abscisne osi s sekanto funkcije f skozi točki x_r in x_{r-1}, kar da

$$x_{r+1} = x_r - \frac{f(x_r)(x_r - x_{r-1})}{f(x_r) - f(x_{r-1})}.$$

Sekantna metoda ni sestavljena po receptu navadne iteracije, saj za izračun novega približka namesto enega uporabimo dva prejšnja približka. Zato tudi pri začetku izvajanja iteracije potrebujemo dva začetna približka (x_0 in x_1).

Naloga 2.12. Implementirajte sekantno metodo v Matlabu in jo preizkusite s funkcijo f iz naloge 2.11 pri začetnih približkih $x_0 = 1$ in $x_1 = 1.1$. Dobljene rezultate primerjajte z rezultati tangentne metode.

Rešitev. Funkcijo `sekantna`, ki izvede sekantno metodo, implementiramo na zelo podoben način kot funkcijo `iteracija` iz naloge 2.3. Pri ten pazimo, da funkcijsko vrednost za vsak približek izračunamo le enkrat, saj izvrednotenje funkcije predstavlja največji računski zalogaj pri izvedbi metode.

```
function [x,X,k] = sekantna(f,x0,x1,tol,N)
% funkcija
%  [x,X,k] = sekantna(f,x0,x1,tol,N)
% izvede sekantno metodo za iskanje ničle funkcije f
%
% Vhodni podatki:
%  f        funkcija, ničlo katere iščemo,
%  x0, x1   začetna približka metode.
```

```
%
% Ostali vhodni in izhodni podatki so enaki kot pri
% funkciji 'iteracija'

X = [x0; x1];
k = 1;
fxk = f(X(k));
while (k == 0 || abs(X(k+1)-X(k)) >= tol) && k < N
    fxkn = f(X(k+1));
    X(k+2) = X(k+1) - fxkn*(X(k+1)-X(k))/(fxkn-fxk);
    fxk = fxkn;
    k = k+1;
end
x = X(k+1);

end
```

Z obravnavo rezultata klica `[x,X,k] = sekantna(f,1,1.1,1e-15,100)` ugotovimo, da se deveti približek sekantne metode ujema z rezultatom vgrajene metode (`fzero(f,0)`) v enajstih decimalkah, deseti približek pa v petnajstih. Torej je število korakov pri sekantni metodi v tem primeru le za odtenek večje od števila korakov pri tangentni metodi. V resnici pa je sekantna metoda učinkovitejša, saj smo na vsakem koraku opravili le eno funkcijsko evaluacijo namesto dveh pri tangentni metodi.

Ena izmed posplošitev tangentne metode je metoda (f, f', f''), pri kateri poleg prvega odvoda funkcije f uporabimo tudi drugi odvod. Približek x_{r+1} izračunamo na podlagi približka x_r po formuli

$$x_{r+1} = x_r - \frac{f(x_r)}{f'(x_r)} - \frac{f''(x_r)f(x_r)^2}{2f'(x_r)^3}.$$

Naloga 2.13. Naj bo f dvakrat zvezno odvedljiva funkcija z enostavno ničlo α. Z uporabo razvoja inverzne funkcije f v Taylorjevo vrsto izpeljite metodo (f, f', f'') in izračunajte red konvergence iteracijskega zaporedja v okolici α.

Rešitev. Ker je α enostavna ničla funkcije f, je $f'(\alpha) \neq 0$ in po izreku o inverzni funkciji obstaja $\delta > 0$, da je f na $(\alpha - \delta, \alpha + \delta)$ obrnljiva. Natančneje, obstaja $\varepsilon > 0$ in taka funkcija $F : (-\varepsilon, \varepsilon) \to (\alpha - \delta, \alpha + \delta)$, da je $F(f(x)) = x$ za vsak $x \in (\alpha - \delta, \alpha + \delta)$. Funkcijo F razvijemo v Taylorjevo vrsto okoli $y \in (-\varepsilon, \varepsilon)$:

$$F(z) = F(y) + F'(y)(z - y) + \frac{1}{2}F''(y)(z - y)^2 + \dots.$$

Vzemimo $z = 0$ in $y = f(x)$ za $x \in (\alpha - \delta, \alpha + \delta)$. Iz

$$F(f(x)) = x, \qquad F'(f(x))f'(x) = 1, \qquad F''(f(x))f'(x)^2 + F'(f(x))f''(x) = 0$$

izrazimo $F'(f(x)) = 1/f'(x)$ in $F''(f(x)) = -f''(x)/f'(x)^3$. Če v Taylorjevi vrsti zanemarimo člene s stopnjo višjo od 2, dobimo

$$\alpha \approx x - \frac{f(x)}{f'(x)} - \frac{f''(x)f(x)^2}{2f'(x)^3},$$

kar določa iteracijsko funkcijo

$$g(x) = x - \frac{f(x)}{f'(x)} - \frac{f''(x)f(x)^2}{2f'(x)^3}.$$

Red konvergence metode lahko določimo z odvajanjem g. Izračunamo

$$g'(x) = \frac{3f''(x)^2 - f'(x)f'''(x)}{2f'(x)^4} f(x)^2.$$

Od tod je razvidno, da je $g'(\alpha) = 0$ in $g''(\alpha) = 0$, medtem ko je vrednost $g'''(\alpha)$ v splošnem različna od nič, zato je red konvergence kubičen.

Še ena metoda, ki poseže po drugem odvodu pri računanju približka za ničlo funkcije f, je Halleyjeva metoda. Približek x_{r+1} izračunamo iz približka x_r po formuli

$$x_{r+1} = x_r - \frac{2f(x_r)f'(x_r)}{2f'(x_r)^2 - f(x_r)f''(x_r)}.$$

Halleyjeva metoda spada v razred Householderjevih metod

$$x_{r+1} = x_r + d\frac{(1/f)^{(d-1)}(x_r)}{(1/f)^{(d)}(x_r)}.$$

Dobimo jo pri $d = 2$, medtem ko tangentna metoda ustreza izbiri $d = 1$.

Naloga 2.14. Naj bo f dvakrat zvezno odvedljiva funkcija. Dokažite, da Halleyjeva metoda ustreza tangentni metodi za funkcijo $F(x) = f(x)/\sqrt{|f'(x)|}$. Kakšen je red konvergence metode v okolici ničle f?

Rešitev. Izračunamo

$$F'(x) = \frac{2f'(x)^2 - f(x)f''(x)}{2f'(x)\sqrt{|f'(x)|}}$$

in izpeljemo iteracijsko funkcijo

$$g(x) = x - \frac{F(x)}{F'(x)} = x - \frac{2f(x)f'(x)}{2f'(x)^2 - f(x)f''(x)}.$$

Z odvajanjem iteracijske funkcije določimo še red metode. Z nekaj računanja dobimo, da je

$$g'(x) = \frac{3f''(x)^2 - 2f'(x)f'''(x)}{(f(x)f''(x) - 2f'(x)^2)^2} f(x)^2,$$

od kjer sledi, da je red konvergence iteracijskega zaporedja v okolici ničle funkcije f v splošnem kubičen.

Naloga 2.15. Poenostavite Halleyjevo metodo za funkcijo f, ki je podana s predpisom $f(x) = x^2 - a$, $a > 0$. Rezultat iteracije ob ustreznem začetnem približku tedaj predstavlja približek za $\sqrt{a}$.

Rešitev. Prva odvoda funkcije $f(x) = x^2 - a$ sta $f'(x) = 2x$ in $f''(x) = 2$. Z nekaj računanja iteracijsko funkcijo g poenostavimo v

$$g(x) = x\frac{x^2 + 3a}{3x^2 + a}.$$

Zanjo smo v eni izmed prejšnjih nalog že dokazali, da določa iteracijsko zaporedje, ki konvergira k $\sqrt{a}$ za vsak začetni približek $x_0 > 0$.

Naloga 2.16. V Matlabu napravite primerjavo tangetne metode, sekantne metode, metode (f, f', f'') in Halleyjeve metode pri iskanju ničle funkcije f iz naloge 2.11. Preizkusite vse metode pri začetnih približkih x_0 iz seznama `1:0.1:10` (ter $x_1 = x_0 + 0.1$ za sekantno metodo). Pri vsaki izvedbi metode poiščite najmanjše tako število k, da se približek x_k absolutno razlikuje od rezultata `fzero(f,1)` za manj kot 10^{-10}. Nato za vsako metodo narišite graf števil k v odvisnosti od začetnih približkov ter graf števila funkcijskih evaluacij v odvisnosti od začetnih približkov. Komentirajte rezultate.

Rešitev. Implementacija tangentne metode je opisana v nalogi 2.11 (funkcija `tangentna`), implementacija sekantne metode (funkcija `sekantna`) pa v nalogi 2.12. Funkciji `fdfddf` in `halley` za izvedbo metode (f, f', f'') in Halleyjeve metode implementiramo na podoben način, pomagamo si lahko s funkcijo `iteracija` iz naloge 2.3. Za vsak začetni približek x_0 in za vsako metodo poišemo najmanjši k, da se približek x_k od vrednosti, dobljene z vgrajeno funkcijo, absolutno razlikuje za manj kot 10^{-10}.

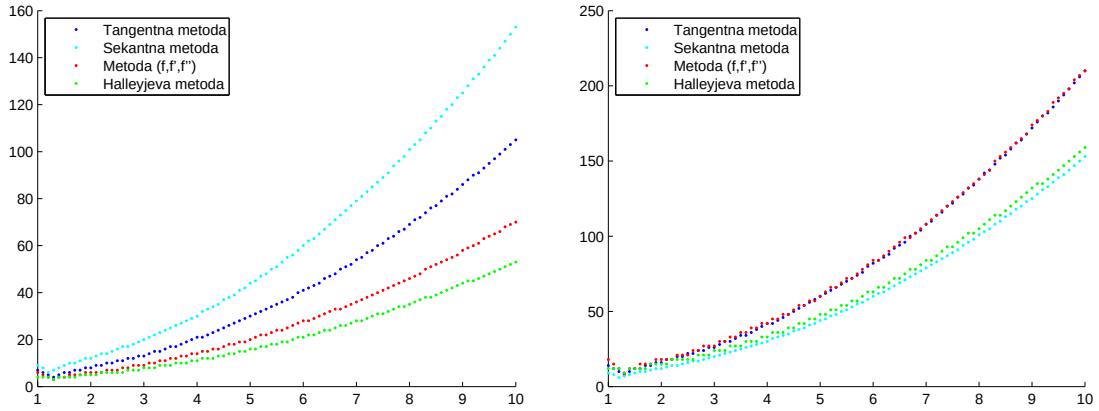
```
f = @(x) x + 4 - exp(x.^2);
df = @(x) 1 - 2*x.*exp(x.^2);
ddf = @(x) -2*exp(x.^2) - 4*x.^2.*exp(x.^2);

x = fzero(f,1);
x0 = 1:0.1:10; tol = 1e-15; N = 200; e = 1e-10;

[k1,k2,k3,k4] = deal(zeros(size(x0)));
for i = 1:length(x0)
    [~,X1] = tangentna(f,df,x0(i),tol,N);
    k1(i) = find(abs(X1-x) < e,1);
    [~,X2] = sekantna(f,x0(i),x0(i)+0.1,tol,N);
    k2(i) = find(abs(X2-x) < e,1);
    [~,X3] = fdfddf(f,df,ddf,x0(i),tol,N);
    k3(i) = find(abs(X3-x) < e,1);
    [~,X4] = halley(f,df,ddf,x0(i),tol,N);
    k4(i) = find(abs(X4-x) < e,1);
end
```

Grafi, s katerimi primerjamo učinkovitost metod, so prikazani na sliki 3. Pri pripravi grafov števila funkcijskih evaluacij upoštevamo, da pri tangentni metodi na vsakem koraku evaluiramo dve funkciji, pri sekantni eno, pri metodi (f, f', f'') in Halleyjevi metodi pa tri. Rezultati kažejo, da za izbrano funkcijo f najhitreje konvergira Halleyjeva metoda, najpočasneje pa sekantna metoda. Metoda (f, f', f'') konvergira hitreje od tangentne.

Iz grafa funkcijskih evaluacij pa je razvidno, da sta sekantna in Halleyjeva metoda učinkovitejši od tangentne metode in metode (f, f', f'').



(a) (b)

Slika 3: Primerjava metod za reševanje nelinearnih enačb na primeru iz naloge 2.16. Na levi so grafi števila korakov, na desni pa grafi števila funkcijskih evaluacij v odvisnosti od začetnih približkov z intervala $[1, 10]$.