Vybrané slajdy k přednášce z předmětu

NUMERICKÉ METODY (12NME1, 12ANM)

na KLFF FJFI ČVUT v Praze

5. Řešení nelineárních rovnic

Pracovní verze, 20. února 2025. Bude průběžně aktualizováno.

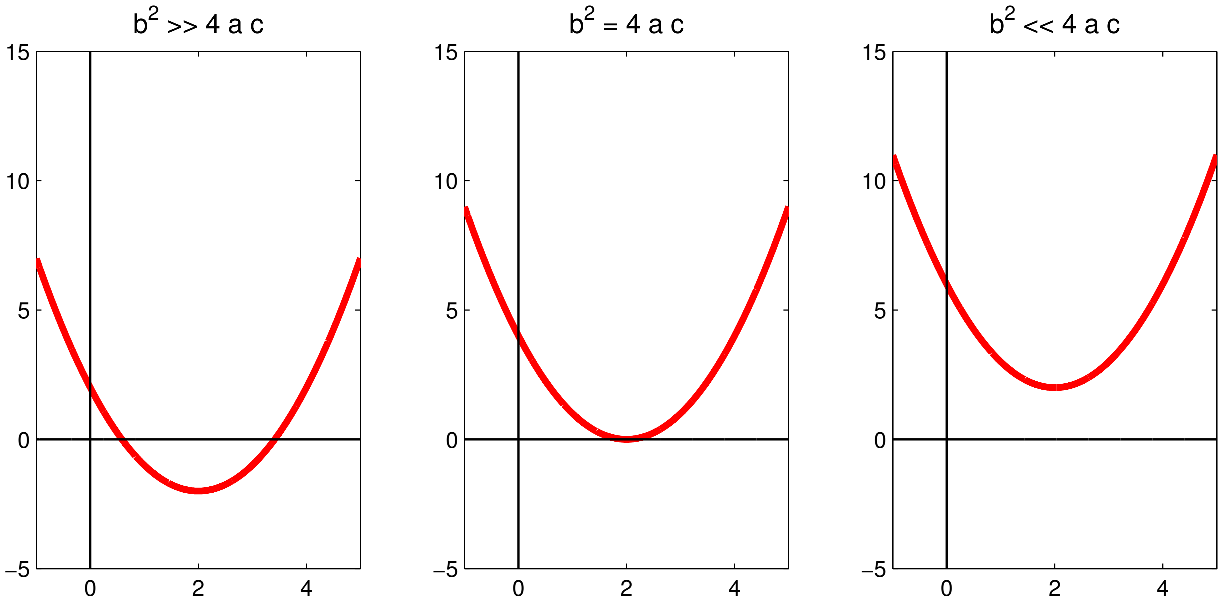
!!! TOTO NEJSOU SKRIPTA !!!

Tento dokument není náhradou přednášek, pouze doplňkem.
Neobsahuje všechna vysvětlení a odvození. Nepokrývá veškerou náplň předmětu, jejíž zvládnutí je nutné ke složení zkoušky.

Primárním zdrojem informací jsou přednášky, účast na nich důrazně doporučuji.

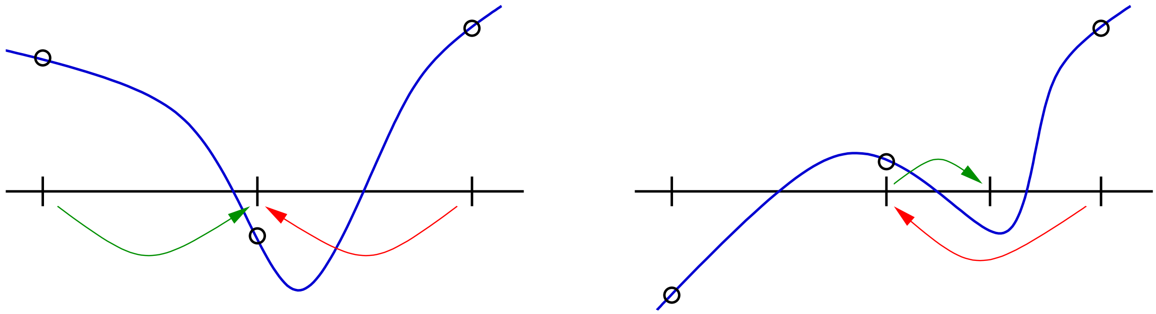
Opravy a připomínky prosím na `pavel.vachal@fjfi.cvut.cz`

Kořeny kvadratické rovnice



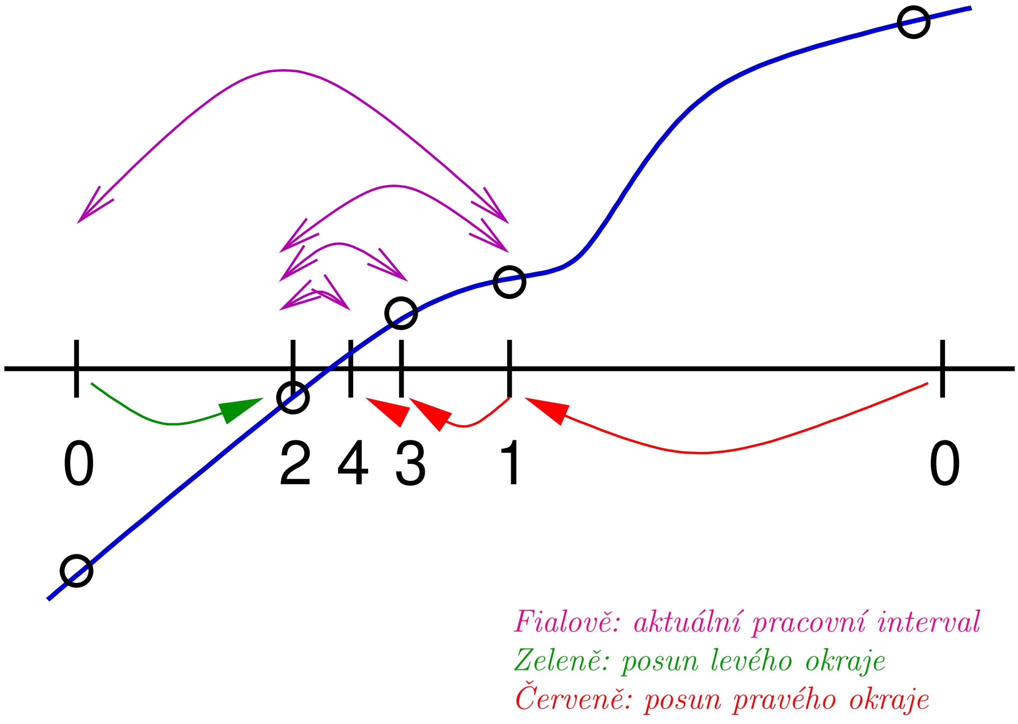
- V reálném oboru nekorektní úloha (problém pro $b^2 \approx 4\,a\,c$) $\Rightarrow$ řešíme přes hledání extrémů

Ohraničení kořene (Bracketing)



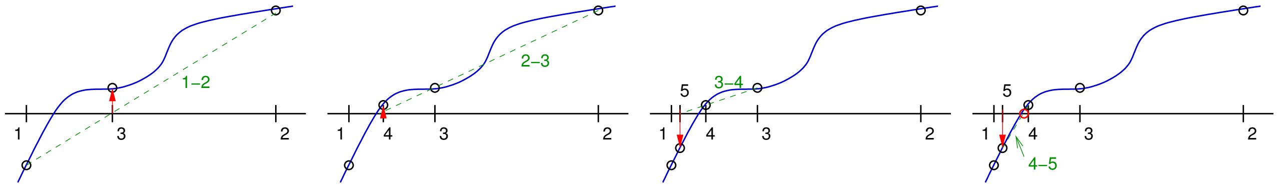
- Hledáme takový interval $[x_1, x_2]$, aby platilo $f(x_1)\, f(x_2) < 0$

- Při zkracování intervalu udržujeme $f(a)\,f(b) < 0$



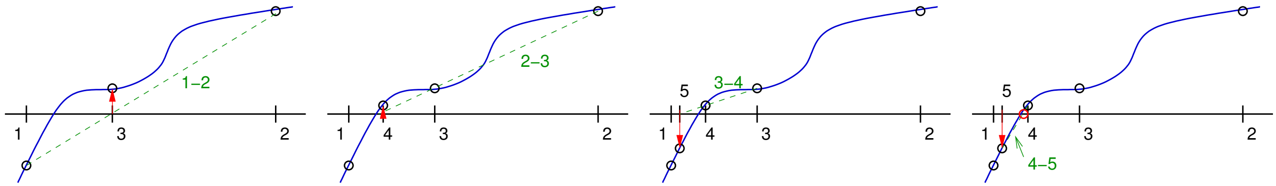
Metoda sečen (sekantová)

- Pro další iteraci vždy použijeme dva nejnovější body



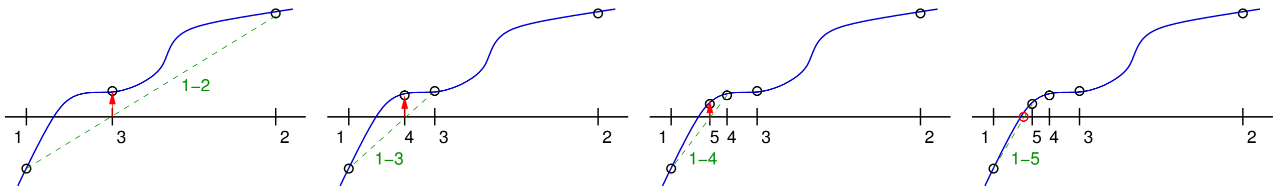
Metoda sečen (sekantová)

▶ Pro další iteraci vždy použijeme dva nejnovější body



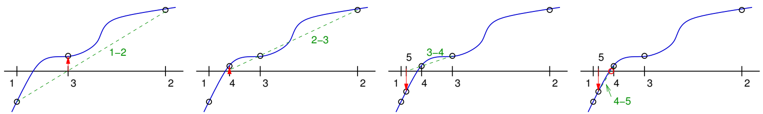
Regula falsi (metoda tětiv, False position method)

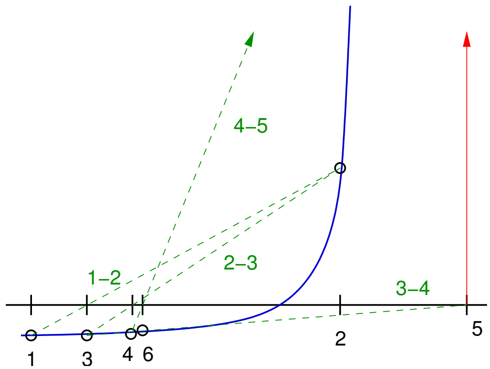
▶ V každé iteraci udržujeme $f(a)\,f(b) < 0$



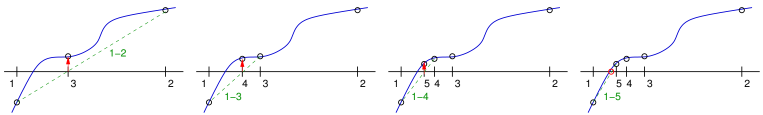
Nežádoucí chování superlineárních metod daleko od kořene

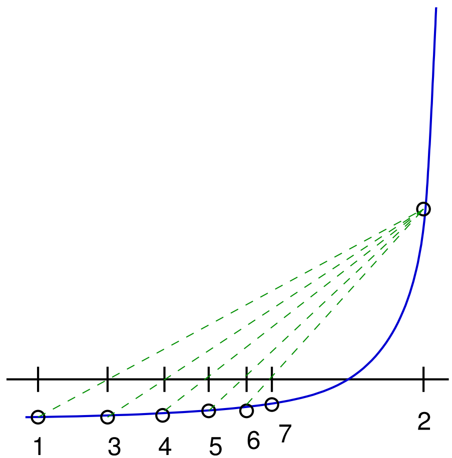
Metoda sečen (sekantová)





Regula falsi (metoda tětiv, False position method)





Odvození Brentovy metody

- Provádíme *inverzní kvadratickou interpolaci* $x = f(y)$:
- Mějme tři body $(a, f_a),\ (b, f_b),\ (c, f_c)$, které interpolujeme Lagrangeovým polynomem

$$x = \frac{(y-f_b)\,(y-f_c)\,a}{(f_a-f_b)\,(f_a-f_c)} + \frac{(y-f_c)\,(y-f_a)\,b}{(f_b-f_c)\,(f_b-f_a)} + \frac{(y-f_a)\,(y-f_b)\,c}{(f_c-f_a)\,(f_c-f_b)}$$

- V bodu $y = 0$ máme

$$\begin{aligned} x(y=0) \ &= \frac{f_b\,f_c\,a}{(f_a-f_b)\,(f_a-f_c)} \ + \ \frac{f_c\,f_a\,b}{(f_b-f_c)\,(f_b-f_a)} \ + \ \frac{f_a\,f_b\,c}{(f_c-f_a)\,(f_c-f_b)} \\ &= \frac{(R-1)(S-1)(T-1)\,b + S\,\Big(T(R-T)(c-b)-(1-R)(b-a)\Big)}{(R-1)\,(S-1)\,(T-1)}, \end{aligned}$$

kde jsme označili $R \equiv f_b/f_c,\ S \equiv f_b/f_a,\ T \equiv f_a/f_c.$ Označíme-li dále

$$\begin{aligned} P &= S\,\Big((R-1)(b-a)+T(R-T)(c-b)\Big)\ , \\ Q &= (R-1)(S-1)(T-1), \end{aligned}$$

má Lagrangeův polynom tvar

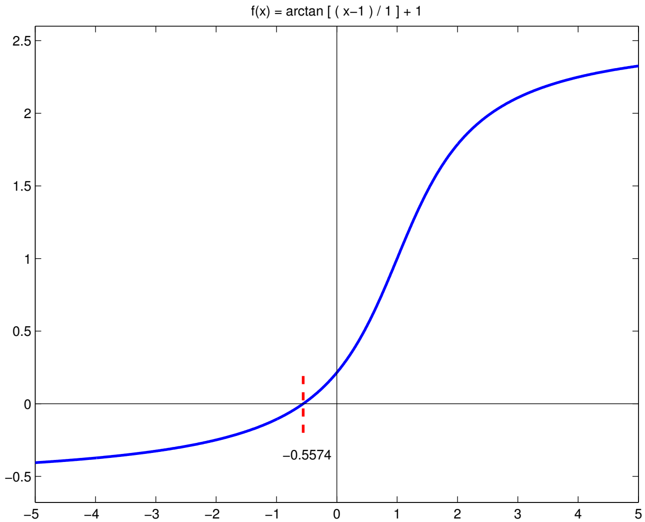
$$x(y=0) \ = b + \frac{P}{Q}.$$

- Při iteraci je b momentální odhad a P/Q kvadratická korekce.

Porovnání Brentovy metody a bisekce

Funkce: $f(x) = \arctan(x-1) + 1$

Analytický kořen: $x = 1 + \tan(-1) \approx -0.55740772465490$



Numericky:

	$tol = 1e-3$			$tol = 1e-10$		
	x_0	$f(x_0)$	kroků	x_0	$f(x_0)$	kroků
BRENT	**-0.557**5...	$\approx 10^{-5}$	9	**-0.557407724**6...	$\approx 10^{-13}$	11
BISEKCE	**-0.557**8...	$\approx 10^{-4}$	14	**-0.557407724**7...	$\approx 10^{-11}$	37

Newton-Raphsonova (tečnová) metoda hledání kořene skalární funkce

- Využívá derivaci funkce $f(x)$, je tedy vhodná především pokud ji umíme rychle spočítat
- Nacházíme se v bodu x_i. Chceme spočítat δ tak, aby $f\left(x_i + \delta\right) = 0$.
- Taylorův rozvoj:

$$f\left(x_i + \delta\right) \; = \; f\left(x_i\right) + \delta f'\left(x_i\right) + \frac{\delta^2}{2} f''\left(x_i\right) + \dots$$

- Položíme tedy $f\left(x_i + \delta\right)$ **rovno nule** a dále **zanedbáme** členy od 2. řádu výše:

$$\underbrace{f\left(x_i + \delta\right)}_{\stackrel{!}{=}\mathbf{0}} \; = \; f\left(x_i\right) + \delta f'\left(x_i\right) \underbrace{+ \tfrac{\delta^2}{2} f''\left(x_i\right) \; + \; \dots}_{\text{zanedb.}}$$

- Vyjádříme posunutí δ jako

$$\delta \; = \; -\frac{f\left(x_i\right)}{f'\left(x_i\right)}$$

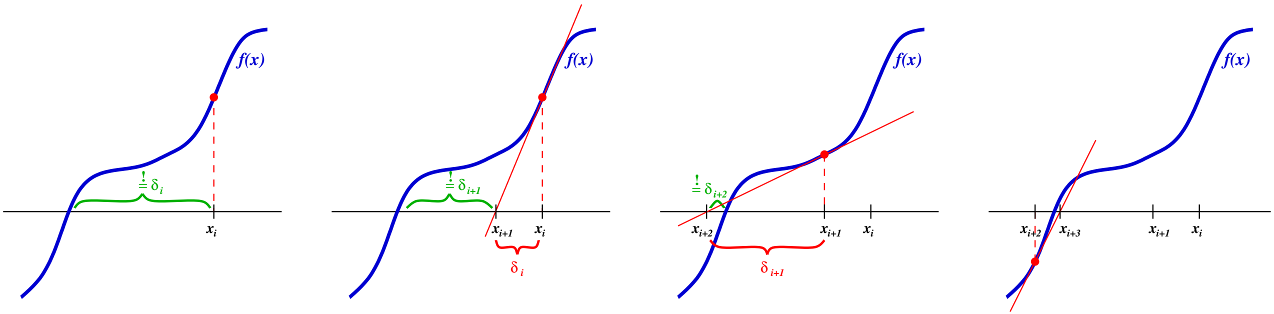
a máme tedy jednu iteraci Newton-Raphsonovy metody:

$$\boxed{x_{i+1} \; = \; x_i - \frac{f\left(x_i\right)}{f'\left(x_i\right)}}$$

- Neboli: funkci jsme nahradili lineární funkcí (tečnou v x_i) a tu jsme pak extrapolovali

Newton-Raphsonova (tečnová) metoda - jak to funguje ve skalárním případě

$$\delta_i = -\frac{f(x_i)}{f'(x_i)}, \qquad x_{i+1} = x_i + \delta_i$$



Odvození metody:

$$\underbrace{f(x_i + \delta)}_{\overset{!}{=} \mathbf{0}} = f(x_i) + \delta f'(x_i) \underbrace{+ \frac{\delta^2}{2} f''(x_i) + \ldots,}_{\text{zanedb.}}$$

$$\delta_i = -\frac{f(x_i)}{f'(x_i)}$$

- POZN.: N-R metodu pro soustavu $\vec{f}(\vec{x}) = \vec{0}$ odvodíme později - opět pomocí Taylorova rozvoje!

Řád přesnosti Newton-Raphsonovy metody

Označme přesné řešení x, přibližnou hodnotu po i-tém kroku x_i a její chybu ε_i. Máme tedy

$$x_{i+1} = x_i + \delta, \qquad x = x_i + \varepsilon_i = x_{i+1} + \varepsilon_{i+1}.$$

Chyba po kroku $i+1$ je

$$\begin{aligned}
\varepsilon_{i+1} = x - x_{i+1} &= \overbrace{x - x_i}^{\varepsilon_i} + \overbrace{x_i - x_{i+1}}^{-\delta} = \varepsilon_i - \delta = \varepsilon_i + \frac{f(x_i)}{f'(x_i)} \\
&= \frac{\varepsilon_i f'(x_i) + f(x_i)}{f'(x_i)} = \frac{\varepsilon_i f'(x - \varepsilon_i) + f(x - \varepsilon_i)}{f'(x - \varepsilon_i)} \\
&= \frac{\varepsilon_i f'(x) - \varepsilon_i^2 f''(x) + \frac{1}{2}\varepsilon_i^3 f'''(x) - \frac{1}{3!}\varepsilon_i^4 f^{(4)}(x) + \dots + \overbrace{f(x)}^{=0} - \varepsilon_i f'(x) + \frac{1}{2}\varepsilon_i^2 f''(x) - \frac{1}{3!}\varepsilon_i^3 f'''(x) + \dots}{f'(x) - \varepsilon_i f''(x) + \frac{1}{2}\varepsilon_i^2 f'''(x) + \dots} \\
&= \frac{-\frac{\varepsilon_i^2}{2} f''(x) + \left[\frac{1}{2} - \frac{1}{3!}\right] \varepsilon_i^3 f'''(x) - \left[\frac{1}{3!} - \frac{1}{4!}\right] \varepsilon_i f^{(4)}(x) + \dots}{f'(x) - \varepsilon_i f''(x) + \frac{1}{2}\varepsilon_i^2 f'''(x) + \dots} \\
&= \frac{-\frac{\varepsilon_i^2}{2} f''(x) + \frac{3-1}{3!}\varepsilon_i^3 f'''(x) - \frac{4-1}{4!}\varepsilon_i f^{(4)}(x) + \dots}{f'(x) - \varepsilon_i f''(x) + \frac{1}{2}\varepsilon_i^2 f'''(x) + \dots} \\
&\approx -\varepsilon_i^2 \, \frac{f''(x)}{2f'(x)}.
\end{aligned}$$

Metoda je tedy kvadratická $\Rightarrow$ rychlá blízko kořene

Ohraničení kořenů polynomu

$$f(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n, \quad a_n \neq 0$$

- Všechny kořeny x_i jsou v mezikruží

$$\frac{|a_0|}{|a_0| + \max\limits_{k=1,\ldots,n} |a_k|} \;\leq\; |x_i| \;\leq\; 1 + \frac{\max\limits_{k=0,\ldots,n-1} |a_k|}{|a_n|}$$

(POZOR! Jedno max jde od 1 do n, druhé od 0 do n-1.)

- Odhad maximálního kladného kořene:
  - Nechť $a_n > 0$,
  - nechť $k = \min\{j : a_{n-j} < 0\}$. *(Tzn. a_{n-k} je první záporný koef. počítáno shora.)*

  Potom pro všechny kořeny $x_i > 0$ platí

$$x_i < 1 + \sqrt[k]{\frac{\max\limits_{j:\, a_j<0} |a_j|}{a_n}}$$

Sturmova věta

- Sturmova posloupnost:

$$
\begin{array}{rcl}
f_0(x) & = & f(x) \\
f_1(x) & = & f'(x) \\
 & \cdots & \\
f_i(x) & = & -1 \cdot \left(\text{zbytek po dělení polynomů } \dfrac{f_{i-2}(x)}{f_{i-1}(x)} \right) \\
 & \cdots & \\
f_k(x) & = & \text{const.} \quad \text{ukončí posloupnost } (k \leq n)
\end{array}
$$

- **Sturmova věta:**

  Má-li algebraická rovnice pouze jednoduché kořeny, potom počet reálných kořenů na intervalu $[a, b]$ je roven rozdílu znaménkových změn ve Sturmově posloupnosti $f_0, f_1, \ldots, f_k$ v krajních bodech a, b.

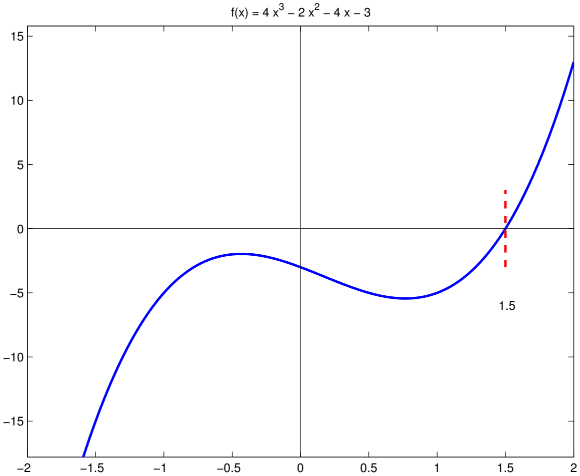
- Má-li algebraická rovnice násobné kořeny, bude $f_k = 0$.
  Vydělíme ji polynomem f_{k-1} a použijeme Sturmovu větu.
  Odtud dostaneme počet reálných kořenů na daném intervalu (bez násobnosti).

Sturmova věta - příklad 1

$$f(x) \equiv 4x^3 - 2x^2 - 4x - 3 \quad = \quad 0$$

Anal. řešení: $x_1 = \frac{3}{2}$, $x_{2,3} = \frac{1}{2}(-1 \pm i)$

$$\begin{aligned} f_0 &= 4x^3 - 2x^2 - 4x - 3 \\ f_1 &= 3x^2 - x - 1 \\ f_2 &= 26x + 29 \\ f_3 &= -1 \end{aligned}$$

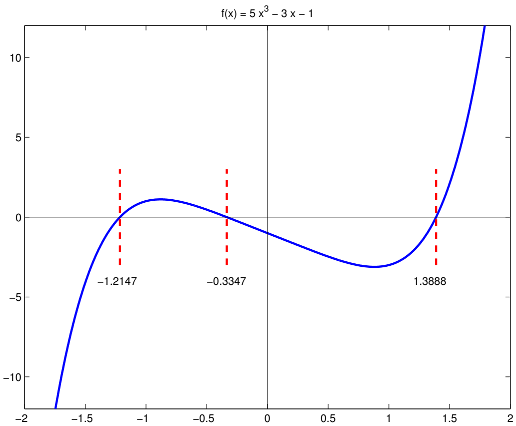


x	$-\infty$	0	1	2	$+\infty$
$\mathrm{sgn} f_0(x)$	-	-	-	+	+
$\mathrm{sgn} f_1(x)$	+	-	+	+	+
$\mathrm{sgn} f_2(x)$	-	+	+	+	+
$\mathrm{sgn} f_3(x)$	-	-	-	-	-
zn. změn	2	2	2	1	1

Sturmova věta - příklad 2

$$f(x) \equiv x^5 - 3x - 1 \quad = \quad 0$$

Anal. řešení: $x_1 = -1.21465$, $x_2 = -0.334734$, $x_3 = 1.38879$, $x_{4,5} = 0.0802951 \pm 1.32836i$



$$\begin{aligned} f_0 &= x^5 - 3x - 1 \\ f_1 &= 5x^4 - 3 \\ f_2 &= \frac{1}{5}(12x + 5) \\ f_3 &= \frac{59083}{20736} \end{aligned}$$

x	$-\infty$	-2	0	2	$+\infty$
$\operatorname{sgn} f_0(x)$	-	-	-	+	+
$\operatorname{sgn} f_1(x)$	+	+	-	+	+
$\operatorname{sgn} f_2(x)$	-	-	+	+	+
$\operatorname{sgn} f_3(x)$	+	+	+	+	+
zn. změn	3	3	1	0	0

Po zjemnění intervalů:

x	-2	-1.5	-1	-0.5	0	0.5	1	1.5	2
$\operatorname{sgn} f_0(x)$	-	-	+	+	-	-	-	+	+
$\operatorname{sgn} f_1(x)$	+	+	+	-	-	-	+	+	+
$\operatorname{sgn} f_2(x)$	-	-	-	-	+	+	+	+	+
$\operatorname{sgn} f_3(x)$	+	+	+	+	+	+	+	+	+
zn. změn	3	3	2	2	1	1	1	0	0

Mullerova metoda

- V aktuální (k-té) iteraci vezmeme tři poslední body (odhady) $x^{(k-2)}$, $x^{(k-1)}$, $x^{(k)}$ a jejich funkční hodnoty $\quad y^{(k-2)} = P(x^{(k-2)}), \quad y^{(k-1)} = P(x^{(k-1)}), \quad y^{(k)} = P(x^{(k)})$ a proložíme je Lagrangeovým polynomem 2. řádu:
- Lokálně (pro tuto iteraci) přeznačíme indexy

$$(x,y)_1 := (x,y)^{(k-2)}, \qquad (x,y)_2 := (x,y)^{(k-1)}, \qquad (x,y)_3 := (x,y)^{(k)},$$

a Lagrangeův polynom je

$$L(x) = \frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)}\, y_1 + \frac{(x-x_1)(x-x_3)}{(x_2-x_1)(x_2-x_3)}\, y_2 + \frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)}\, y_3$$

což lze zapsat jako

$$L(x) = A\,x^2 + B\,x + C,$$

kde

$$\begin{aligned}
A &= \frac{1}{(x_1-x_2)(x_1-x_3)}\, y_1 + \frac{1}{(x_2-x_1)(x_2-x_3)}\, y_2 + \frac{1}{(x_3-x_1)(x_3-x_2)}\, y_3, \\
B &= \frac{x_2\,x_3}{(x_1-x_2)(x_1-x_3)}\, y_1 + \frac{x_1\,x_3}{(x_2-x_1)(x_2-x_3)}\, y_2 + \frac{x_1\,x_2}{(x_3-x_1)(x_3-x_2)}\, y_3, \\
C &= \frac{-(x_2+x_3)}{(x_1-x_2)(x_1-x_3)}\, y_1 + \frac{-(x_1+x_3)}{(x_2-x_1)(x_2-x_3)}\, y_2 + \frac{-(x_1+x_2)}{(x_3-x_1)(x_3-x_2)}\, y_3.
\end{aligned}$$

(Všimněme si, že v každém sloupci se zlomky liší jen v čitateli.)

Mullerova metoda (pokračování)

- Hledáme kořen původního polynomu $P(x)$. Předpokládáme, že je blízko kořene aproximujícího polynomu $L(x)$, tedy položíme $L(x) = 0$ a řešíme kvadratickou rovnici pro x:

$$x_{a,b} = \frac{-B \pm \sqrt{B^2 - 4\,A\,C}}{2\,A}$$

- Z kořenů vybereme ten, který je blíže k $x^{(k)}$ $\quad (=$ k našemu $x_3)$:

$$x^{(k+1)} = \begin{cases} x_a & \text{pokud } |x_a - x_3| < |x_b - x_3| \\ x_b & \text{jinak} \end{cases} \quad ,$$

  vypočítáme funkční hodnotu $y^{(k+1)} = P(x^{(k+1)})$ a jdeme na další iteraci s body

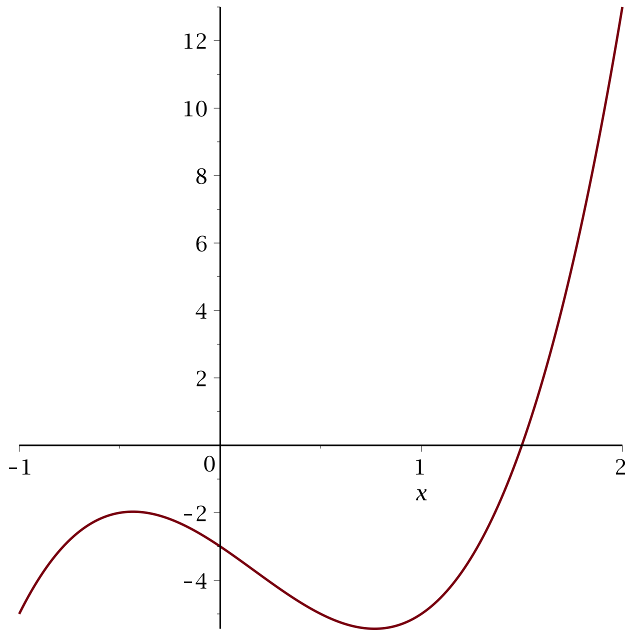
$$(x,y)_1 := (x,y)^{(k-1)}, \qquad (x,y)_2 := (x,y)^{(k)}, \qquad (x,y)_3 := (x,y)^{(k+1)}$$

- Inicializace $x^{(-1)}$, $x^{(0)}$ a $x^{(1)}$: vezmeme libovolné tři body, třeba ekvidistantní reálná čísla

Mullerova metoda - příklad

Hledáme kořen polynomu $P(x) = 4\,x^3 - 2\,x^2 - 4\,x - 3$

Analytické řešení: jeden reálný kořen $x = \frac{3}{2}$, dva komplexní $x = -\frac{1}{2} \pm \frac{1}{2}\,i$



Laguerrova metoda - algoritmus

Hledáme kořen polynomu $P_n(x)$ stupně n.

- Zvolíme počáteční odhad $x^{(0)}$
- V k-té iteraci postupně spočítáme

$$G = \frac{P_n'(x^{(k)})}{P_n(x^{(k)})},$$

$$H = G^2 - \frac{P_n''(x^{(k)})}{P_n(x^{(k)})},$$

$$a = \frac{n}{G \pm \sqrt{(n-1)\,(n\,H - G^2)}},$$

  kde znaménko volíme tak, abychom maximalizovali abs. hodnotu jmenovatele
- $x^{(k+1)} = x^{(k)} - a$
- Opakujeme, dokud a není dostatečně malé nebo iterací není příliš mnoho

Laguerrova metoda - příklad: $P_3(x) = 4x^3 + 3x^2 + 2x + 1,$ $x^{(0)} = -1.0$

- První iterace:

$$x^{(0)} = -1.0$$

$$G = \frac{P_3'(x^{(0)})}{P_3(x^{(0)})} = \left.\frac{12x^2 + 6x + 2}{4x^3 + 3x^2 + 2x + 1}\right|_{x^{(0)}} = -4$$

$$H = G^2 - \frac{P_3''(x^{(0)})}{P_3(x^{(0)})} = (-4)^2 - \left.\frac{24x + 6}{4x^3 + 3x^2 + 2x + 1}\right|_{x^{(0)}} = 16 - 9 = 7,$$

$$a = \frac{3}{G \pm \sqrt{(3-1)\,(3H - G^2)}} = \left\{ \begin{array}{c} \cancel{\frac{3}{-0.837722339831620}} \\ \boxed{\frac{3}{-7.162277660168380}} \end{array} \right\} = -0.418861169915810$$

$$x - a = -1 - (-0.418861169915810) = -0.581138830084190$$

$$\Rightarrow x^{(1)} = -0.581138830084190$$

- Rychle konverguje k reálnému kořenu:

k	$x^{(k)}$	G	H	a
0	-1.0	-4	7	$\approx -4.1886 \times 10^{-1}$
1	-0.581138830084190	$\approx 3.8974 \times 10^{1}$	$\approx 1.6397 \times 10^{3}$	$\approx 2.4704 \times 10^{-2}$
2	-0.605843146337280	$\approx -7.3747 \times 10^{4}$	$\approx 5.4384 \times 10^{9}$	$\approx -1.3560 \times 10^{-5}$
3	-0.605829586188266	$\approx 4.6193 \times 10^{14}$	$\approx 2.1338 \times 10^{29}$	$\approx 2.1648 \times 10^{-15}$
4	-0.605829586188268	$\ldots$		

Laguerrova metoda - jak to funguje

- Pro polynom ve tvaru $P_n(x) = C \prod_{i=1}^{n} (x - x_i)$ platí

$$P_n'(x) = C\left[1\,(x-x_2)\,(x-x_3)\ldots(x-x_n) + (x-x_1)\,1\,(x-x_3)\ldots(x-x_n) + \ldots\right] = P_n(x) \sum_{i=1}^{n} \frac{1}{x-x_i}$$

$$P_n''(x) = P_n(x) \sum_{i=1}^{n} \frac{1}{x-x_i} \sum_{k=1,k\neq i}^{n} \frac{1}{x-x_k}$$

$$\frac{\mathrm{d}}{\mathrm{d}x} \ln|P_n(x)| = \frac{\mathrm{d}}{\mathrm{d}x}\left(\sum_{i=1}^{n} \ln|x-x_i| + \ln C\right) = \sum_{i=1}^{n} \frac{1}{x-x_i}, \qquad \frac{\mathrm{d}^2}{\mathrm{d}x^2} \ln|P_n(x)| = \sum_{i=1}^{n} \frac{1}{(x-x_i)^2}$$

- Definujeme

$$G \equiv \frac{\mathrm{d}}{\mathrm{d}x} \ln|P_n(x)| = \frac{P_n'}{P_n}, \qquad H \equiv \frac{\mathrm{d}^2}{\mathrm{d}x^2} \ln|P_n(x)| = \left[\frac{P_n'}{P_n}\right]^2 - \frac{P_n''}{P_n}$$

- “Drastické předpoklady”: $x - x_1 = a, \quad x - x_i = b, \quad 2 \leq i \leq n, \quad a < b$

- Potom máme

$$G = \frac{1}{a} + \frac{n-1}{b}, \qquad H = \frac{1}{a^2} + \frac{n-1}{b^2} \quad \Rightarrow \quad a = \frac{n}{G \pm \sqrt{(n-1)\,(nH - G^2)}}.$$

Znaménko volíme tak, abychom maximalizovali abs. hodnotu jmenovatele.

- Tedy iterace $x_1^{(k+1)} = x_1^{(k)} - a$

Syntetické dělení polynomů

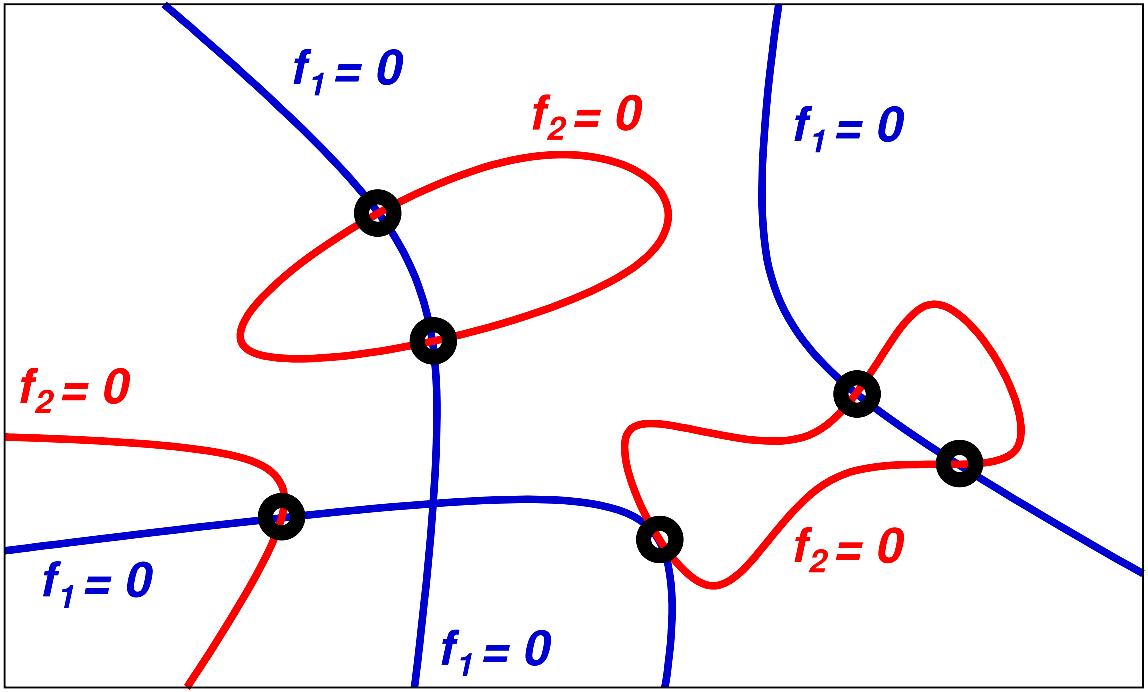
$$\frac{a_n\,x^n + a_{n-1}\,x^{n-1} + \cdots + a_0}{b_m\,x^m + b_{m-1}\,x^{m-1} + \cdots + b_0} = \overbrace{c_{n-m}\,x^{n-m} + \cdots + c_0}^{\text{podíl}} + \frac{\overbrace{d_{m-1}\,x^{m-1} + \cdots + d_0}^{\text{zbytek po dělení}}}{b_m\,x^m + b_{m-1}\,x^{m-1} + \cdots + b_0}$$

- Jednoduchý postup, tzv. *long division* analogický ručnímu dělení čísel
- Syntetické dělení postupuje efektivněji:
  - nepoužívá x, pouze koeficienty polynomů
  - potřebuje méně operací
- Postupně dostáváme koeficienty podílu (c_i) a zbytku po dělení (d_i) dvou polynomů
- Implementováno např. v proceduře POLDIV v Numerical Recipes (dle Knutha), kde koeficienty podílu dostáváme pomocí vztahů

$$\begin{aligned} c_{n-m} &= \frac{a_n}{b_m}, \\ c_{n-m-1} &= \frac{a_{n-1} - c_{n-m}\,b_{m-1}}{b_m}, \\ &\dots \end{aligned}$$

a zároveň počítáme koeficienty zbytku $d_{m-1}, \dots$.

Řešení 2 nelineárních rovnic o 2 neznámých



Řešení soustavy nelineárních rovnic

Speciální případ: převedení úlohy na hledání extrému

- Máme soustavu $\vec{f}(\vec{x}) = \vec{0}$, neboli

$$
\begin{aligned}
f_1\left(x_1, x_2, \ldots, x_n\right) &= 0 \\
f_2\left(x_1, x_2, \ldots, x_n\right) &= 0 \\
&\cdots \\
f_n\left(x_1, x_2, \ldots, x_n\right) &= 0.
\end{aligned}
$$

- Pokusíme se vyjádřit f pomocí potenciálu V:

$$f_i = -\frac{\partial V(\vec{x})}{\partial x_i}, \qquad i = 1 \ldots n.$$

- Jestliže se nám to podaří, nahradíme řešení soustavy $\vec{f}(\vec{x}) = \vec{0}$ hledáním extrému $V(\vec{x})$.

Řešení nelineárních soustav prostou iterací

- Předpokládejme, že soustavu $\vec{f}(\vec{x}) = \vec{0}$ umíme vyjádřit ve tvaru $\vec{x} = \vec{\varphi}(\vec{x})$, tedy

$$
\begin{aligned}
x_1 &= \varphi_1(x_1, x_2, \ldots, x_n) \\
x_2 &= \varphi_2(x_1, x_2, \ldots, x_n) \\
&\ldots \\
x_n &= \varphi_n(x_1, x_2, \ldots, x_n) .
\end{aligned}
$$

  Tato soustava má stejná řešení jako soustava původní.

- Iterujeme: $\quad \vec{x}^{(k+1)} = \vec{\varphi}\left(\vec{x}^{(k)}\right)$.

Postačující podmínka konvergence

- Nechť $\vec{\varphi}(\vec{x})$ je *kontrahující zobrazení.*

  Tzn.: Nechť v nějakém okolí G řešení $\vec{\xi}$ platí:

  (1) $(\forall \vec{x} \in G)\,(\vec{\varphi}(\vec{x}) \in G)$

  (2) $(\exists q \in (0,1))\,(\forall \vec{x_1}, \vec{x_2} \in G)\,(\|\vec{\varphi}(\vec{x_1}) - \vec{\varphi}(\vec{x_2})\| \leq q\,\|\vec{x_1} - \vec{x_2}\|)$.

  Pak iterační posloupnost konverguje k řešení $\vec{\xi}$ a platí

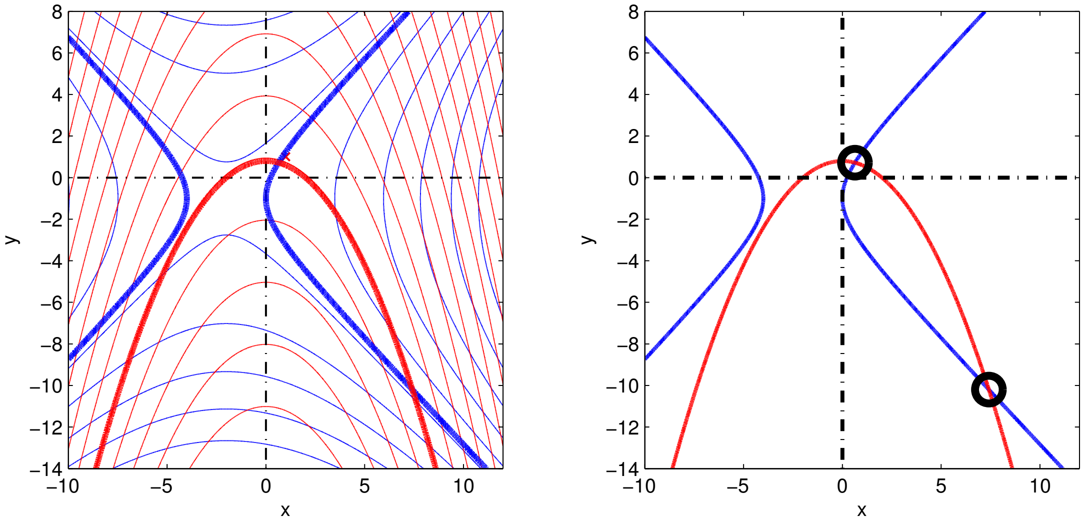
$$\left\|\vec{x}^{(k)} - \vec{\xi}\right\| \leq \frac{q}{1-q}\left\|\vec{x}^{(k)} - \vec{x}^{(k-1)}\right\| .$$

- POZN.: Pokud v okolí řešení existují všechny parciální derivace $\vec{\varphi}$, lze podmínku konvergence zapsat pomocí Jakobiánu $\mathbf{J}$ zobrazení $\vec{\varphi}(\vec{x})$ jako $\;|\mathbf{J}\vec{\varphi}(\vec{x})| \leq q < 1$.

Řešení nelineárních soustav - příklad I

demo_simpiter.m

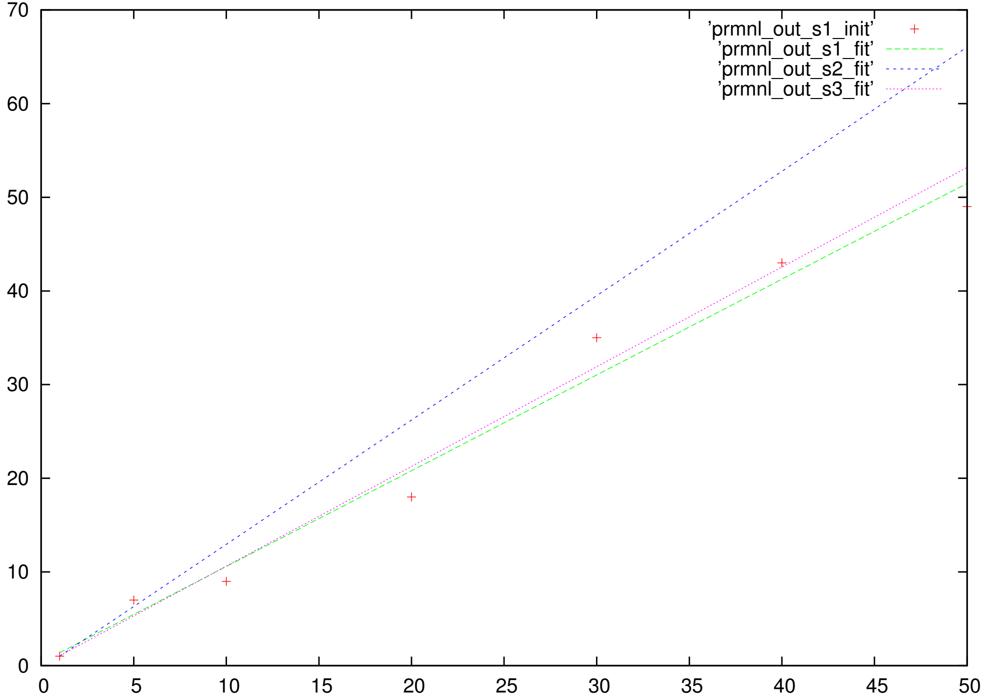
$$\begin{aligned} x^2 + 4\,x - y^2 - 2\,y - 1 &= 0 \\ x^2 + 5\,y - 4 &= 0 \end{aligned} \quad \Leftrightarrow \quad \begin{aligned} x &= \tfrac{y^2+2\,y+1-x^2}{4} \\ y &= \tfrac{4-x^2}{5} \end{aligned}$$



Dvě reálná řešení $\quad (x,y) \approx (0.637\ ,\ 0.719), \qquad (x,y) \approx (7.417\ ,\ -10.202)$

a dvě komplexní

$$(x,y) \approx (-4.027 + 0.962\,i\ ,\ -2.258 + 1.549\,i),$$
$$(x,y) \approx (-4.027 - 0.962\,i\ ,\ -2.258 - 1.549\,i).$$


'prmnl_out_s1_init'
'prmnl_out_s1_fit'
'prmnl_out_s2_fit'
'prmnl_out_s3_fit'
0
10
20
30
40
50
60
70
0
5
10
15
20
25
30
35
40
45
50

Newton-Raphsonova metoda pro soustavy nelineárních rovnic

- Stejně jako v případě 1 rovnice systém linearizujeme v bodu $\vec{x}^{(k)}$ (k-tém odhadu řešení):

$$\underbrace{f_i\left(\vec{x}^{(k)} + \vec{\delta x}^{(k)}\right)}_{\text{předpokl. } = 0,} = \underbrace{f_i\left(\vec{x}^{(k)}\right) + \overbrace{\sum_{j=1}^{n} \frac{\partial f_i}{\partial x_j} \delta x_j^{(k)}}^{\nabla f(x^{(k)}) \cdot \vec{\delta x}^{(k)}}}_{\text{položíme } = 0} + \underbrace{\mathcal{O}\left[\left(\vec{\delta x}^{(k)}\right)^2\right]}_{\text{zanedbáme}}$$

- Máme tedy soustavu n rovnic o n neznámých:

$$\underbrace{\begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{pmatrix}}_{\mathbf{J}} \cdot \begin{pmatrix} \vec{\delta x}_1^{(k)} \\ \vec{\delta x}_2^{(k)} \\ \vdots \\ \vec{\delta x}_n^{(k)} \end{pmatrix} = - \begin{pmatrix} f_1\left(\vec{x}^{(k)}\right) \\ f_2\left(\vec{x}^{(k)}\right) \\ \vdots \\ f_n\left(\vec{x}^{(k)}\right) \end{pmatrix}$$

Tím vypočteme $\vec{\delta x}^{(k)}$ a další odhad bude $\vec{x}^{(k+1)} = \vec{x}^{(k)} + \vec{\delta x}^{(k)}$

- Vektorově

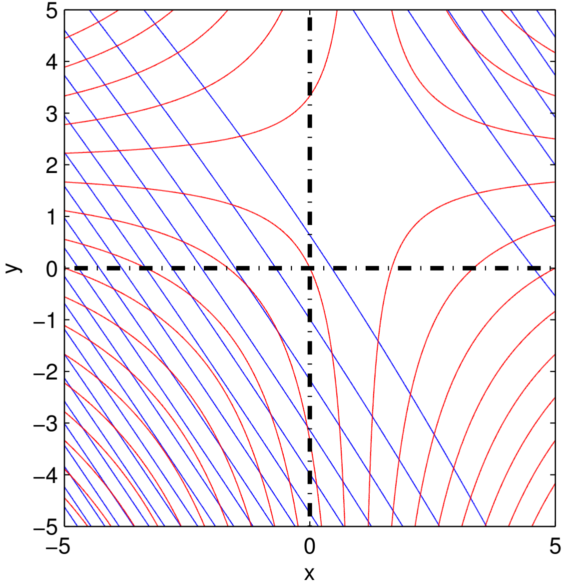
$$\vec{x}^{(k+1)} = \vec{x}^{(k)} - \left[\mathbf{J} f\left(\vec{x}^{(k)}\right)\right]^{-1} \cdot \vec{f}\left(\vec{x}^{(k)}\right)$$

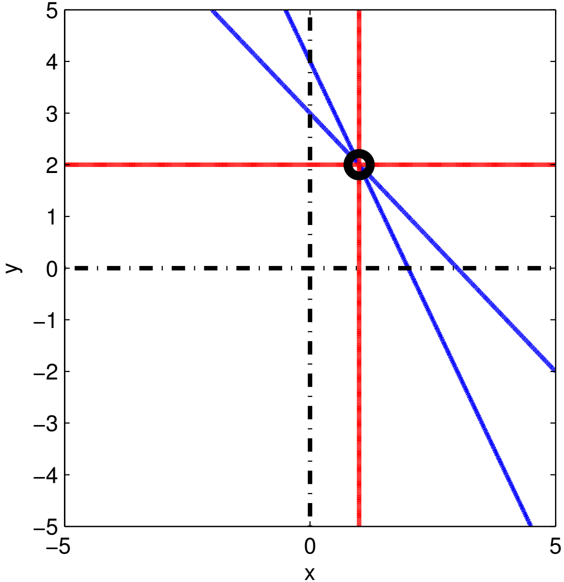
Řešení nelineárních soustav - příklad III

DMNEWTA.PAS

- Newton-Raphsonovou metodou řešíme soustavu

$$\begin{aligned} x(2x+3y-10)+y(y-7)+12 &= 0 \\ x(y-2)-y+2 &= 0 \end{aligned}$$





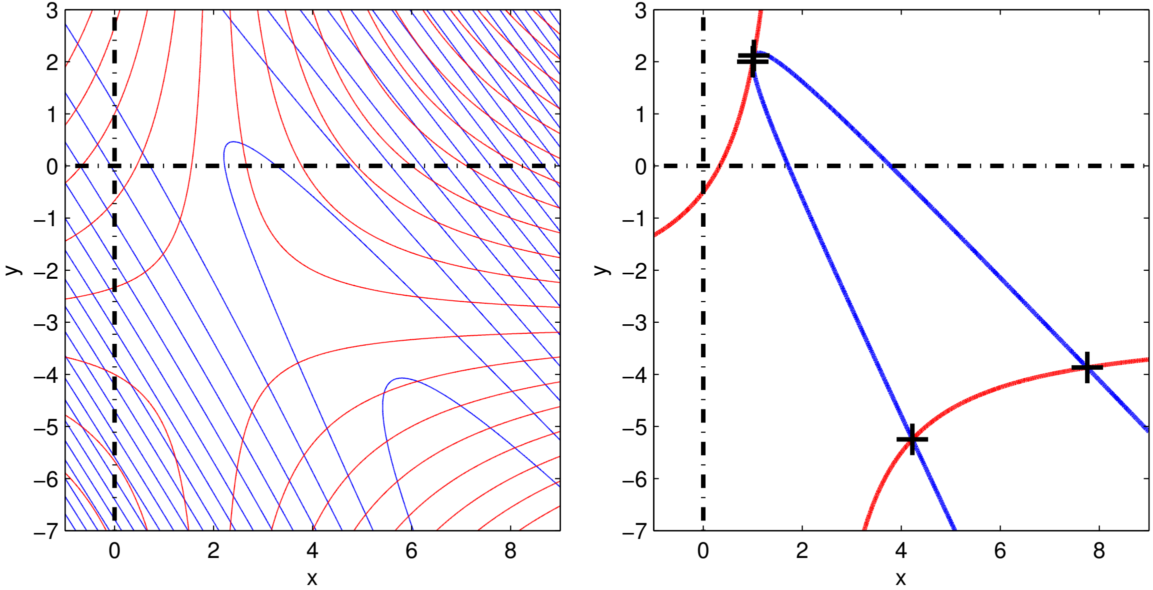
- Soustava má jen jedno řešení $(x,y)=(1,2)$, metoda k němu vždy konverguje

Řešení nelineárních soustav - příklad IV

DMNEWT2.PAS

- Newton-Raphsonovou metodou řešíme soustavu

$$\begin{aligned} x(2x+3y-11)+y(y-7)+13 &= 0 \\ 3x+y(x-2)-1 &= 0 \end{aligned}$$



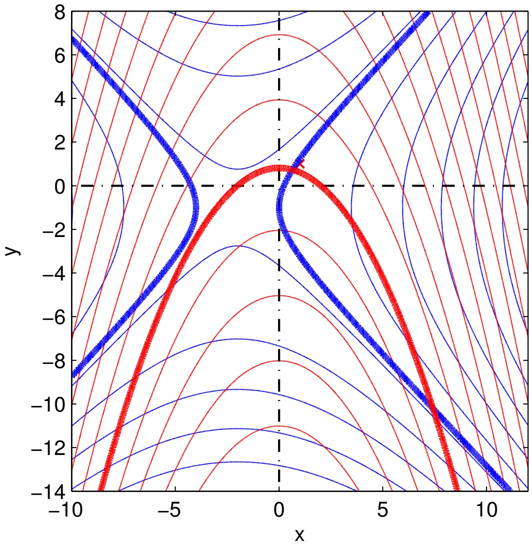
- Soustava má 4 reálná řešení, metoda konverguje k některému z nich podle počát. odhadu

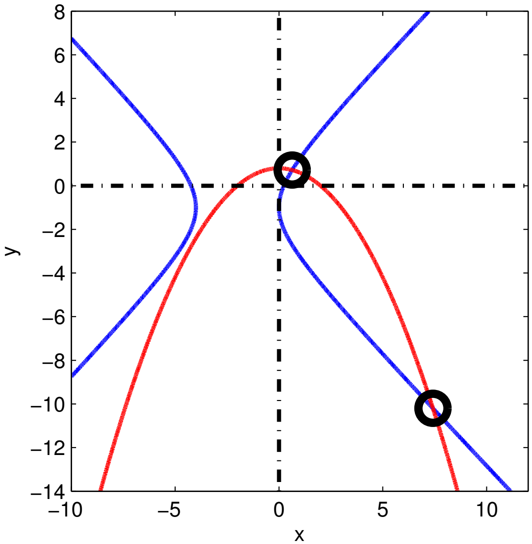
Řešení nelineárních soustav - příklad V

$\boxed{\text{demo\_newtraph.m}}$

- Newton-Raphsonovou metodou řešíme (opět) soustavu

$$
\begin{aligned}
x^2 + 4\,x - y^2 - 2\,y - 1 &= 0 \\
x^2 + 5\,y - 4 &= 0
\end{aligned}
$$





- Soustava má 2 reálná řešení, metoda konverguje k některému z nich podle počát. odhadu:
  - S počátečním odhadem $(1, 1)$ najde řešení $(x, y) \approx (0.6371, 0.7188)$,
  - S počátečním odhadem $(5, -5)$ najde řešení $(x, y) \approx (7.4169, -10.202)$.