Řešení nelineárních rovnic pomocí iterací

- též se můžete setkat s názvem metoda postupných aproximací
- rovnici typu $x = f(x)$, kde $f(x)$ je obecně nelineární funkce, či dokonce operátor, lze velmi často řešit pomocí metody prostých iterací: zvolíme počáteční odhad $x = x_0$ a iterujeme pomocí $x_{n+1} = f(x_n)$

  zda pro $n \to \infty$ konvergujeme, tj. $x_n \to x_p$, které řeší $x_p = f(x_p)$, závisí obecně jak na problému (funkci f), tak na x_0

- Věta o pevném bodě (fixed point) kontrahujícího zobrazení (Banachova věta)

  Nechť f je spojité zobrazení definované na oblasti D a nechť f zobrazuje D na sebe, tj. $f(D) \subseteq D$, a splňuje Lipšicovskou podmínku kontrahujícího zobrazení

  tj. $\exists L < 1$ takové, že pro $\forall x_1, x_2 \in D$ platí

  $$\| f(x_1) - f(x_2) \| \leq L \| x_1 - x_2 \| \quad \text{(ve vhodné normě)}$$

  Pak 1) $f(x)$ má na D právě jeden pevný bod $x_p = f(x_p)$,

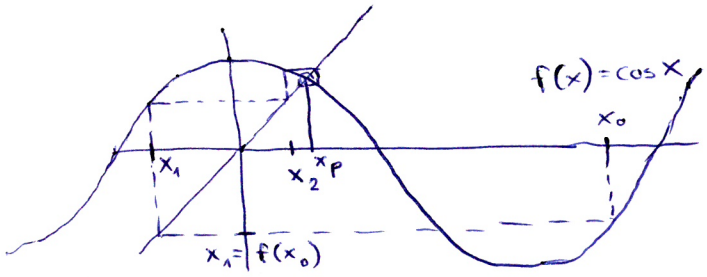
  2) posloupnost $x_n = f(x_{n-1})$ konverguje k x_p pro libovolné počáteční $x_0 \in D$, tj. $\lim_{n \to \infty} \| x_n - x_p \| = 0$,

  3) pro odhad chyby v n-té iteraci platí

  $$\| x_n - x_p \| \leq \frac{L^n}{1-L} \| x_1 - x_0 \|$$

  (důkaz viz např. J. Segethová: Základy numerické matematiky)

Př. pro $f \in C^1\langle a,b \rangle$ tedy metoda konverguje, pokud $|f'(x)| < 1$ pro $\forall x \in \langle a,b \rangle$



pro $x = \cos x$ dostáváme dokonce řešení pro lib. x_0, tj. vždy konvergujeme k $x_p \doteq 0{,}739\ldots$

Otázkou je, jak rychle?

- <u>rychlost konvergence</u> (speed of convergence) je dána tzv. <u>řádem iterační metody</u> (order of convergence)

Označíme-li chybu v n-tém kroku $e_n = x_n - x_p$, pak

$$e_{n+1} = x_{n+1} - x_p = f(x_n) - f(x_p) = f(x_p + e_n) - f(x_p) =$$

pro diferencovatelnou funkci $= f'(x_p)\, e_n + \frac{1}{2} f''(x_p)\, e_n^2 + \dots$ pro malé e_n

v závislosti na tom, kolik derivací je nulových v bodě x_p, dostáváme metody různých řádů. Tj. pro <u>lineární metody</u>

bude $f'(x_p) \neq 0$, a tedy $e_{n+1} \approx q\, e_n$, kde $q = f'(x_p)$

$\Rightarrow$ počet správných cifer roste lineárně s počtem kroků, neboť $e_n \sim e_0 q^n$ a tedy

$$\underbrace{\log|e_n| - \log|e_{n+1}|}_{\text{počet platných cifer získaných v jedné iter.}} = [n - (n+1)] \log|q| = -\log|q|$$

pro <u>kvadratické metody</u> bude $f'(x_p) = 0$ a $f''(x_p) \neq 0$

a tedy $e_{n+1} \approx \frac{1}{2} f''(x_p)\, e_n^2$

nyní $\log|e_{n+1}| = \log|q| + 2 \log|e_n|$

a počet platných cifer se v každé iteraci minimálně zdvojnásobí.

a pod. pro metody vyšších řádů

Obecně <u>řád metody</u> je nenulové α, pro nějž existuje konečná, nenulová limita

$$\lim_{n\to\infty} \frac{|x_{n+1} - x_p|}{|x_n - x_p|^\alpha} = C < \infty \qquad \left(= \lim_{n\to\infty} \frac{|e_{n+1}|}{|e_n|^\alpha} \right)$$

(říkáme též, že iterační metoda má řád konvergence α v bodě x_p)

- C samozřejmě závisí na $f(x)$
- α je jednoznačně určeno díky $C \neq 0$ a $\alpha \geq 1$

- existují i obecnější metody (např. vícekrokové $F_n(x_n, x_{n-1}, \dots, x_{n-s+1}) = x_{n+1}$ a pokud $F_n = F$, jde o stacionární metodu)

a vždy konvergentní metody fungující pro lib. $f(x)$ a x_0, ale jsou typicky pomalejší a používají se jako startovací

Př. Hledání řešení rovnice $x = \cos x$ prostou iterací je lineární metoda ($\alpha = 1$), neboť $\frac{d}{dx}\cos x\Big|_{x=x_p} \neq 0$

přesněji $\frac{d\cos x}{dx}\Big|_{x=x_p} = -\sin x_p \doteq -0{,}6736\ldots \approx -\frac{2}{3}$

a tedy $e_{n+1} \approx -\frac{2}{3} e_n$, z čehož vidíme, že zhruba po šesti iteracích se zmenší chyba o jeden řád $\left(\left(\frac{2}{3}\right)^6 \sim \frac{1}{10}\right)$

Urychlování konvergence lineárních metod

1) Aitkenova Δ^2 metoda - pro lin. metody lze pro dostatečně velké n psát chybu jako $e_n \simeq c q^n$ a pro tři po sobě jdoucí aproximace máme (alespoň přibližně)

$$x_n - x_p \simeq c q^n$$
$$x_{n+1} - x_p \simeq c q^{n+1}$$
$$x_{n+2} - x_p \simeq c q^{n+2}$$

což jsou 3 rovnice pro 3 neznámé x_p, c a q

Vyjádříme
$$q = \frac{x_{n+1} - x_p}{x_n - x_p} = \frac{x_{n+2} - x_p}{x_{n+1} - x_p}$$

z čehož
$$x_p \simeq \frac{x_n x_{n+2} - x_{n+1}^2}{x_n - 2x_{n+1} + x_{n+2}} = \underbrace{x_n - \frac{(x_{n+1} - x_n)^2}{x_{n+2} - 2x_{n+1} + x_n}}$$

toto je lepší z hlediska zaokrouhlovacích chyb ($x_n x_{n+2}$ a x_{n+1}^2 můžou být blízká)

neboli
$$x_p \approx x_n' = x_n - \frac{(\Delta x_n)^2}{\underbrace{\Delta^2 x_n}_{\text{odtud název metody}}}$$
, kde Δ je operátor dopředné diference

pozor! výsledek je opět jen přibližný $\Rightarrow$ nová posloupnost x_n', pro kterou lze ukázat, že $\lim_{n\to\infty} \frac{x_n' - x_p}{x_n - x_p} = 0$

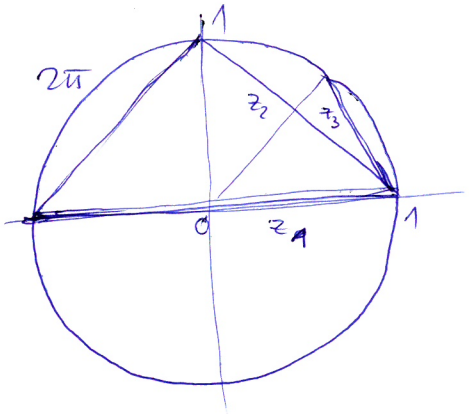
a tedy konverguje rychleji k x_p než x_n

obecně však jde opět o lineární metodu, ovšem $q' < q$

Pozn: později se setkáme s podobnou „extrapolací" ze známého chování chyby u numerické derivace a integrace

Archimedův výpočet čísla π

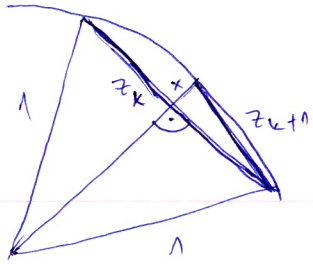
- číslo π postupně aproximujeme obvodem vepsaného mnohoúhelníku do kružnice



$$2\pi \doteq 2z_1 \quad \Rightarrow \quad \pi \doteq 2$$
$$2\pi \doteq 4z_2 \quad \Rightarrow \quad \pi \doteq 2\sqrt{2}$$
$$2\pi \doteq 8z_3 \quad \Rightarrow \quad \pi \doteq 4\sqrt{2-\sqrt{2}}$$

atd.

obecně dostaneme rekurentní vztah pro z_k:



nejprve
$$x = 1 - \sqrt{1-\left(\frac{z_k}{2}\right)^2}$$

a tedy
$$z_{k+1} = \sqrt{x^2 + \left(\frac{z_k}{2}\right)^2} =$$
$$= \sqrt{1 - 2\sqrt{1-\left(\frac{z_k}{2}\right)^2} + \left(1 - \cancel{\left(\frac{z_k}{2}\right)^2}\right) + \cancel{\left(\frac{z_k}{2}\right)^2}}$$
$$z_{k+1} = \sqrt{2}\sqrt{1-\sqrt{1-\left(\frac{z_k}{2}\right)^2}}$$

- protože $2\pi \doteq 2^k z_k = 2y_k$

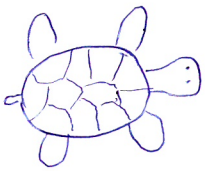
bude $\pi \doteq y_{k+1} = 2^k z_{k+1} = \frac{2^{k+1}}{\sqrt{2}}\sqrt{1-\sqrt{1-\left(\frac{y_k}{2^k}\right)^2}}$

ovšem zde je problém s odčítáním 2 blízkých čísel
$\left(\frac{y_k}{2^k} \to 0 \text{ pro } k \text{ velké}\right)$

proto výraz vynásobíme $\frac{\sqrt{1+\sqrt{\dots}}}{\sqrt{1+\sqrt{\dots}}}$ a dostaneme

$$\tilde{y}_{k+1} = \frac{2^{k+1}}{\sqrt{2}} \frac{\sqrt{\cancel{1} - \left(\cancel{1} - \left(\frac{\tilde{y}_k}{2^k}\right)^2\right)}}{\sqrt{1+\sqrt{1-\left(\frac{\tilde{y}_k}{2}\right)^2}}} = \tilde{y}_k \sqrt{\frac{2}{1+\sqrt{1-\left(\frac{\tilde{y}_k}{2^k}\right)^2}}}$$

což je stabilní výpočet (chyba cca 10^{-14} po 25 iteracích)



2) Aitkenova-Steffensenova metoda

použijeme-li opakovaně Aitkenovu Δ^2 metodu, dostaneme posloupnosti

$$x_n \to x_n' \to x_n'' \to \dots$$

$$\begin{array}{ccccc} x_0 & x_1 & x_2 & x_3 & x_4 \\ x_0' & x_1' & x_2' & & \\ x_0'' & & & & \end{array}$$

atd.

Alternativně lze uvažovat posloupnosti $x_0^{(k)}, x_1^{(k)}$ a $x_2^{(k)}$ které získáme následovně (nyní se mění horní index)

1) na začátku položíme $x_0^{(0)} = x_0$, $x_1^{(0)} = x_1 = f(x_0)$, $x_2^{(0)} = x_2 = f(x_1)$

2) pro $k = 0, 1, 2, \dots$ spočteme vždy $x_0^{(k)}$ z Aitkenovy Δ^2 metody

tj.
$$x_0^{(k+1)} = x_0^{(k)} - \frac{(x_1^{(k)} - x_0^{(k)})^2}{x_2^{(k)} - 2x_1^{(k)} + x_0^{(k)}}$$

ovšem $x_1^{(k)} = f(x_0^{(k)})$ a $x_2^{(k)} = f(x_1^{(k)})$

souhrnně lze psát jedinou posloupnost $x_0^{(k)}$ danou vztahem

$$x_0^{(k+1)} = x_0^{(k)} - \frac{[f(x_0^{(k)}) - x_0^{(k)}]^2}{f(f(x_0^{(k)})) - 2f(x_0^{(k)}) + x_0^{(k)}}$$

o které si ukážme, že konverguje kvadraticky k x_p

- nyní totiž neiterujeme f, ale novou funkci

$$\tilde{f}(x) = x - \frac{[f(x) - x]^2}{f(f(x)) - 2f(x) + x}$$

- označme $q = f'(x_p)$, pak

$$f(x_p + \varepsilon) = x_p + \varepsilon q + O(\varepsilon^2)$$
$$f(f(x_p + \varepsilon)) = x_p + \varepsilon q^2 + O(\varepsilon^2)$$

a dostaneme

$$\tilde{f}(x_p + \varepsilon) = x_p + \varepsilon - \frac{(\cancel{x_p} + \varepsilon q - \cancel{x_p} - \varepsilon)^2}{\cancel{x_p} + \varepsilon q^2 - 2\cancel{x_p} - 2\varepsilon q + \cancel{x_p} + \varepsilon} + O(\varepsilon^2)$$

$$= x_p + \varepsilon - \varepsilon \frac{(q-1)^2}{(q-1)^2} + O(\varepsilon^2) = x_p + O(\varepsilon^2)$$

neboli $\tilde{f}(x_p) = x_p$ (jde o pevný pod $\tilde{f}$)

a $\tilde{f}'(x_p) = 0$ $\Rightarrow$ kvadratická konvergence

Hledání kořenů algebraické rovnice $g(x)=0$

- vždy konvergentní metody pro reálné funkce na intervalu $\langle a,b\rangle$
  - zákl. předpoklad: $g(x)$ je spojitá na $\langle a,b\rangle$ a $g(a)g(b)<0$, aby měla kořen
  - standardní „klasické" metody:

    a) metoda bisekce (půlení intervalu) →

    2 posloupnosti a_n a b_n

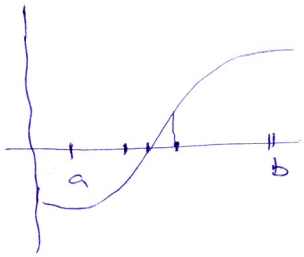
    $a_0=a$, $b_0=b$

    pro $n=0,1,2\dots$ $\quad x_n=\frac{a_n+b_n}{2}$

    je-li $g(a_n)g(x_n)\le 0$: $b_{n+1}=x_n$, $a_{n+1}=a_n$

    jinak $b_{n+1}=b_n$, $a_{n+1}=x_n$

    dokud $|g(x_n)|\ge\varepsilon$



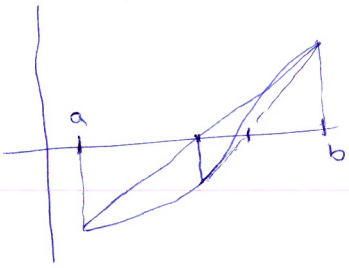
    b) metoda regula falsi (sečen)

    opět a_n a b_n, $a_0=a$, $b_0=b$

    ale nyní proložíme přímku:

    pro $n=0,1,\dots$ $\quad x_n=\frac{a_n g(b_n)-b_n g(a_n)}{g(b_n)-g(a_n)}$

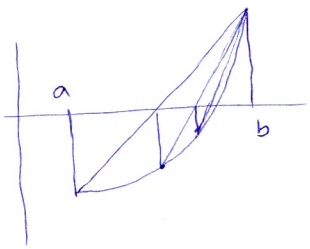
    podmínky stejné



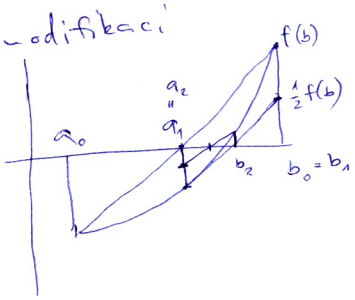
    c) modifikovaná metoda regula falsi

    pouze obrázkově

    před



    po modifikaci



    podrobnosti o konvergenci a detaily algoritmů viz J. Segethová: Základy numerické matematiky, kap. 6.3

- iterační metody vyššího řádu
  - ~~přep~~ píšeme $x=f(x)=x-\phi(x)g(x)$, kde $\phi(x)$ je vhodná funkce

    je-li x_p kořen $g(x_p)=0$, pak je též pevný bod fce $f(x)$

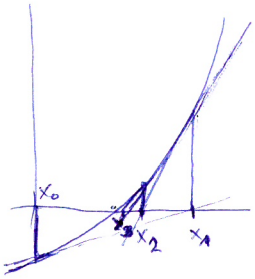
    za předpokladu, že jde o kontrahující zobr.
  - $\phi(x)$ nesmí mít kořen na intervalu, kde hledáme x_p

    a navíc $0<|\phi(x)|<\infty$, tedy musí být omezená
  - jinou možností je např. $f(x)=x-F(g(x))$, kde

    $F(0)=0$, $F(y)\neq 0$ pro $y\neq 0$

a) <u>Newtonova metoda</u> (metoda tečen)

- pokud má být $x = f(x) = x - \phi(x) g(x)$ metodou 2. řádu musí být $f'(x_p) = 0 = 1 - \phi'(x_p)\underbrace{g(x_p)}_{=0} - \phi(x_p) g'(x_p)$ (x_p je kořen $g(x)=0$)



z čehož $\phi(x_p) = \dfrac{1}{g'(x_p)}$

neboli zvolíme $\phi(x) = \dfrac{1}{g'(x)}$ $\Rightarrow$ $\boxed{x_{n+1} = x_n - \dfrac{g(x_n)}{g'(x_n)}}$

a protože $f'(x_p) = \dfrac{g(x_p) g''(x_p)}{[g'(x_p)]^2} = 0$, vyžadujeme $g'(x_p) \neq 0$ a existenci $g''(x_p)$

- nekonverguje vždy!, ale pokud na určitém intervalu je $|f'(x)| < 1$, pak ano
- často funguje, pokud $g''(x)$ kolem kořene x_p nemění znaménko
- nutný „nástřel" pomocí nějaké vždy konvergentní metody a pak rychle doiterovat Newtonovou metodou

- speciálně pro $g(x) = x^2 - c = 0$ (hledáme $\sqrt{c}$) dostaneme $x_{n+1} = x_n - \dfrac{x_n^2 - c}{2x_n} = \dfrac{1}{2}\left(x_n + \dfrac{c}{x_n}\right)$

b) <u>dvoubodová metoda sečen</u>

- neznáme-li derivaci $g'(x)$, nebo je ji obtížné spočítat, lze ji aproximovat, např. $g'(x_k) = \dfrac{g(x_k) - g(x_{k-1})}{x_k - x_{k-1}}$ a tedy $\boxed{x_{n+1} = \dfrac{x_{n-1} g(x_n) - x_n g(x_{n-1})}{g(x_n) - g(x_{n-1})}}$

- lze ukázat, (viz např. Isaacson, Keller: Analysis of Numerical Methods) že pro $g(x)$ splňující $g'(x) \neq 0$ na intervalu, kde hledáme kořen a $g''(x)$ nemění znaménko

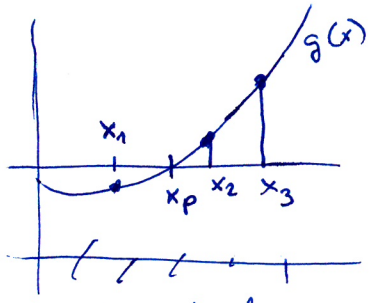
je konvergence řádu $\alpha = \dfrac{\sqrt{5}+1}{2} \simeq 1{,}618$ (zlatý řez)

(ukáže se, že $e_{n+1} \leq e_n e_{n-1}$ a pro počáteční chybu $\varrho = \max(e_0, e_1)$ dostaneme $e_2 \leq \varrho^2$, $e_3 \leq \varrho^3$, $e_4 \leq \varrho^5$ atd. exponenty tvoří Fibonacciho posloupnost)

- protože však na jednu iteraci potřebujeme vyčíslit g pouze jednou, může být rychlejší než Newton, kde potřebujeme i derivaci

- <u>kořeny polynomů</u> – lze použít výše uvedené metody
  - Newtonovu metodu i pro hledání komplexních kořenů
  - jen pro vícenásobné kořeny lineární konvergence. pokud známe násobnost kořene p: $x_{n+1} = x_n - p\dfrac{g(x_n)}{g'(x_n)}$
  - např. Laguerreova metoda (využívá 2. derivace, též Skowron, Gould:

c) Iterační metoda pomocí Padého aproximace



nebo body v komplexní rovině

- racionální aproximace jisté fce $f(x)$

$$g(x) \simeq \frac{a_0 + a_1 x + \dots + a_N x^N}{b_0 + b_1 x + \dots b_M x^M} = P_{II}^{[N/M]}(x) = \frac{P^N(x)}{Q^M(x)}$$

stále, že se obvykle položí $b_0 = 1$

($P_I^{[N/M]}(x)$ by byla Padého aproximace získaná porovnáním s Taylor. rozvojem)

- k určení a_i a b_j je potřeba $N+M+1$ bodů $\Rightarrow$

soustava lin. rovnic $P^N(x_k) = g(x_k)\, Q^M(x_k)$

- použijeme-li $P_{II}^{[1/1]}(x)$ k aproximaci fce $g(x)$, jejíž kořen hledáme a určíme-li kořen jako $x_p \simeq -\frac{a_0}{a_1}$, pak opakováním tohoto postupu dostaneme iterační metodu

$$x_n = \frac{x_{n-1}\dfrac{(x_{n-2}-x_{n-3})}{F(x_{n-1})} + x_{n-2}\dfrac{x_{n-3}-x_{n-1}}{F(x_{n-2})} + x_{n-3}\dfrac{x_{n-1}-x_{n-2}}{F(x_{n-3})}}{\dfrac{x_{n-2}-x_{n-3}}{F(x_{n-1})} + \dfrac{x_{n-3}-x_{n-1}}{F(x_{n-2})} + \dfrac{x_{n-1}-x_{n-2}}{F(x_{n-3})}}$$

neboli jde o vážený průměr bodů x_{n-1}, x_{n-2} a x_{n-3} pomocí vah $w_{n-1} = \dfrac{x_{n-2}-x_{n-3}}{F(x_{n-1})}$ atd.

Řešení rovnic pro více proměnných

- Banachova věta o pevném bodě platí stále

  x je vektor, $f(x)$ je vektorová funkce

  iterujeme $x_i^{n+1} = f_i(x_1^n, \dots x_D^n)$ pro $i = 1, 2, \dots, D$

- pro chybu nyní máme

$$e^{n+1} = x^{n+1} - x_{(p)} = f(x^n) - f(x_{(p)}) = J(x_{(p)})\, e^n + O(\|e^n\|^2)$$

  kde $J(x)$ je Jakobián zobr. f, tj.

$$J = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_D} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_D}{\partial x_1} & \cdots & \frac{\partial f_D}{\partial x_D} \end{pmatrix}$$

  a pokud má být $f(x)$ kontrahující zobrazení, musí být všechna vlastní čísla J menší než 1 (v absolutní hodnotě) pro $\forall x$ z uvažované oblasti

- řád konvergence α: $\lim\limits_{n\to\infty} \dfrac{\|x^{n+1} - x_{(p)}\|}{\|x^n - x_{(p)}\|^\alpha} = C < \infty$

  a je-li $J(x_{(p)}) \neq 0$ budeme mít lineární konvergenci, atd.

- chceme-li vyřešit soustavu $g(x) = 0$ pro vekt. fci g, pak můžeme hledat pevný bod zobrazení

$$f(x) = x - A(x)\, g(x)$$

  kde $A(x)$ je nesingulární matice pro $\forall x$ z jistého okolí $x_{(p)}$ který je řešení $g(x_{(p)}) = 0$

- <u>Newtonova metoda</u> plyne z požadavku, aby byl Jakobián nulový, tj

$$0 = \frac{\partial f_i}{\partial x_j} = \delta_{ij} - \sum_k \left(\frac{\partial A_{ik}}{\partial x_j} \underbrace{g_k}_{0 \text{ pro } x = x_{(p)}} + A_{ik} \frac{\partial g_k}{\partial x_j} \right) \quad v\ x = x_{(p)}$$

  a tedy $\displaystyle\sum_{k=1}^{D} A_{ik} \frac{\partial g_k}{\partial x_j} = \delta_{ij}$

  neboli musíme iterovat

$$x^{n+1} = x^n - J_g(x^n)^{-1} g(x^n), \quad \text{kde } J_g(x)_{ij} = \frac{\partial g_i(x)}{\partial x_j}$$

  nebo zapsáno jako soustava rovnic $J_g(x^n)(x^n - x^{n+1}) = g(x^n)$

- někdy se používá tzv. relaxace, kdy na pravé straně $x = f(x)$ použijeme již modifikované složky x^n: $x_i^{n+1} = f_i(x_1^{n+1}, \dots x_{i-1}^{n+1}, x_i^n, \dots x_D^n)$