SJÄLVSTÄNDIGA ARBETEN I MATEMATIK

MATEMATISKA INSTITUTIONEN, STOCKHOLMS UNIVERSITET

En studie av iterationsmetoder

av

Sibel Cicek

2018 - No K13

En studie av iterationsmetoder

Sibel Cicek

Självständigt arbete i matematik 15 högskolepoäng, grundnivå

Handledare: Yishao Zhou

2018

En studie av iterationsmetoder

Sammanfattning

I det här arbetet har vi försökt att förstå de olika iterativa metoderna som finns för att lösa de icke-linjära ekvationerna samt deras konvergens och konvergenshastighet. Vi undersöker även frågan om hur snabba konvergensmetoder kan konstrueras och tillämpningar i egenvärdeproblem och linjära ekvationssystem samt matrisinvers.

Abstract

The aim of this project work is to understand different iterative methods to solve the nonlinear equations and their convergence and convergence speed. We also study how high order convergent methods can be constructed. Furthermore, we apply iterative methods to eigenvalue problems of a type of matrices, and systems of linear equations and computation of matrix inverse.

Innehåll

1. Inledning

Rötter till en icke-linjär ekvation $f(x) = 0$ kan i allmänhet inte uttryckas i sluten form. Även om detta är möjligt är uttrycket ofta så komplicerat att det är opraktiskt att använda. För att lösa icke-linjära ekvationer är vi därför hänvisade till approximativa metoder, vanligen grundade på idén om successiv approximation eller linearisering. Dessa metoder ger, utgående från ett eller flera närmevärden en talföljd $x_0, x_1, x_2, ...$ som förutsätts konvergera mot det sökande nollstället. Vid vissa metoder är det för alla reella nollställen tillräckligt för konvergens att känna ett intervall $[a, b]$, som innehåller nollstället. Andra metoder kräver en startapproximation, som ligger nära det sökta nollstället, men har i gengäld snabbare konvergens. Ofta är det därför lämpligt att i början av räkningarna använda en grov metod samt i slutskedet byta till en snabbare konvergent metod.

I den har studien skall vi besvara följande frågor:

(1) Hur konstrueras en iteration?
(2) Under vilka villkor är talföljden $x_0, x_1, x_2, ...$ genererad av iterationen konvergent?
(3) Hur snabbt konvergerar talföljden $x_0, x_1, x_2, ...$?

Grundläggande problem. Det grundläggande problemet är att söka en lösning (rot) till $f(x) = 0$, $x \in [a, b] \subset \mathbb{R}$ där $f : \mathbb{R} \to \mathbb{R}$ och $\mathbb{R}$ är mängden av alla reella tal. Lite mer generellt behövs det ofta hitta lösning av ett kvadratiskt system $F(x) = 0 \Leftrightarrow f_i(x_1, ..., x_n) = 0$, $i = 1, 2, ..., n$. Här är $F = (f_1, ..., f_n)$ och $x_i \in \mathbb{R}, i = 1, 2, ..., n$ samt n ett positivt heltal. Ofta finns det inte sluten lösning, så vi behöver approximera och rätt ofta med iterativa metoder. Allmänt börjar vi för heltal $m \geq 0$ initiala gissningar $x^{(0)}, x^{(1)}, ..., x^{(m-1)}$ itererar vi vidare med formeln

$$x^{(k+1)} = \phi(x^{(k)}, x^{(k-1)}, ..., x^{(k-m+1)})$$

där ϕ är en lämplig avbildning $\mathbb{R}^m \to \mathbb{R}$ och kallas *iterationsfunktion*. Första två av de ovanstående frågorna kan formuleras på följande sätt: Hur konstrueras iterationsfunktionen ϕ? Under vilka värden på $x^{(0)}, x^{(1)}, ..., x^{(m)}$ är talföljden $x_0, x_1, x_2, ...$ genererad av iterationsfunktionen ϕ konvergent?

Grundläggande begrepp. För vår studie behövs det följande definitioner och begrepp.

Definition. (*konvergensordning*) Antag att $f(x) = 0$ har en rot p och att följden $\{x^{(k)}\}$ konvergerar mot α. Vi säger att konvergens är ordning $\nu \geq 1$ om

$$\frac{|e^{(k+1)}|}{|e^{(k)}|^\nu} \to C \text{ då } k \to \infty, \quad e^{(k)} = x^{(k)} - \alpha$$

där $C \neq 0$ kallas *asymptotisk felkonstant*.

Notera att en metod åtminstone skall konvergera linjärt dvs felet måste åtminstone reduceras med en konstant faktor efter en iteration, t ex antalet korrekta siffror ökar med en iteration. Vi behöver också komma ihåg att en bra metod för rotsökning konvergerar kvadratiskt dvs antalet korrekta siffror fördubblas med en iteration. Vi exemplifierar detta med beräkning av $\sqrt{2}$. Det är det samma som att lösa ekvationen $f(x) := x^2 - 2 = 0$. Med intervallhalvering får vi efter 20 steg

```
2., 1. , 1.5, 1.25, 1.375, 1.4375,
1.40625, 1.42188, 1.41406, 1.41797,
1.41602, 1.41504, 1.41455, 1.41431,
1.41418, 1.41425, 1.41422, 1.4142,
1.41421, 1.41421
```

då $a = 0, b = 4$. Men med Newton-Raphsons metod får vi efter 5 steg med startvärdet 1

```
1, 1.5, 1.41667, 1.41422, 1.41421, 1.41421.
```

Definition. (*Lokal vs global konvergens*)

En metod har *lokal konvergens* om den konvergerar till en given rot α för alla initiala gissningar tillräckligt nära α (omgivningen av α).

En metod har *global konvergens* om den konvergerar mot roten för alla initiala gissningar.

En god initialgissning är extremt viktigt för icke-linjära problemlösningar. Om α ligger i ett givet intervall $[a, b]$ så ska initialgissning $x^{(0)}$ ligga mellan a och b. Allmänt kan vi säga att global konvergens kräver långsammare (noggrannare) metod men säkrare. Bästa strategin är att kombinera globalt konvergent metod med en snabb konvergent metod som ofta är lokalt konvergent; dvs. vi får en bra initialgissning genom en globalt konvergent metod.

Vi fokuserar i denna rapport på iterationsmetoder för enkelrotsökning till ekvationen $f(x) = 0$. Vi börjar med att undersöka Newton-Raphsons metod, sekantmetoden, och intervallhalvering för att få idéer och insikt på begrepp om konvergens och snabb konvergens. En allmän teori för konvergens av iterationsmetoder kommer att presenteras. Vi skall även undersöka hur konvergens kan accelereras genom extrapolation och genom kombination av olika snabba metoder. Därefter studerar vi, i detalj, en del resultat för att lösa polynomekvationer, egenvärdeproblem och uppskattning av rötter till polynomekvationer. I detta sammanhang demonstrerar användning av Sturms polynomföljder. Till sist visar vi att iterationsmetoder kan användas för lösandet av linjära ekvationssystem och matrisinvers. Observera att vi inte kommer att undersöka fel som påverkas av numeriska beräkningar i denna studie. Vi diskuterar inte heller komplexitet av metoderna. De flesta ämnen i denna studie är tagna ur boken [7] kompletterad med [9]. Men vi använder oss av matematik på nivån motsvarande Matematik I och Matematik II. Vi har syfte till att en lärarstudent kan följa resonemang och bevis presenterade i denna rapport. I rapporten används decimalpunkten i förekommande fall med hänsyn till datorprograms standard .

2. Tre klassiska iterationsmetoder

Detta avsnitt beskriver tre klassiska iterationsmetoder och dess konvergensegenskaper. Mer precis skall vi svara på de tre frågorna i Inledningen.

2.1. Intervallhalvering.

Idén bakom intervallhalverings metoden är att lokalisera en rot genom tecken. Antag att $f(x)$ är kontinuerlig på $[a, b]$. Då vet vi att om $f(a)f(b) < 0$ måste roten p ligga mellan a och b. Metoden kan sammanfattas så här: Låt $a_1 = a$, $b_1 = b$.

(1) Bestäm tecken av $f(a_1)$ och $f(b_1)$. Antag, utan att göra inskränkning, att $f(a_1) < 0$.
(2) Halvera $[a_1, b_1]$. Sätt $p_1 = \frac{a_1+b_1}{2}$. Bilda ett nytt intervall

$$[a_2, b_2] = \begin{cases} [p_1, b_1] & \text{om } f(p_1) < 0 \\ [a_1, p_1] & \text{om } f(p_1) > 0 \end{cases}$$

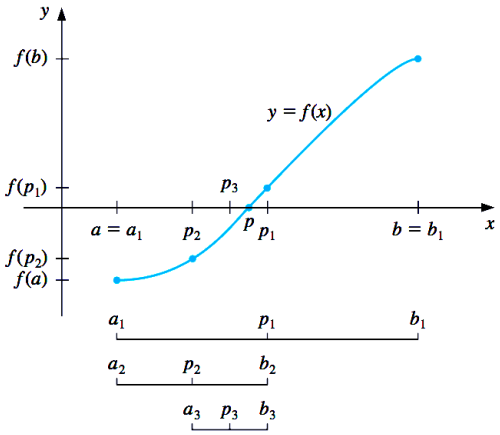
Antag nu $f(p_1) \neq 0$ (annars är p_1 en rot).

(3) Fortsätt på samma sätt att halvera intervallen:
$p_k = \frac{a_k+b_k}{2}$. Bilda nytt intervall

$$[a_{k+1}, b_{k+1}] = \begin{cases} [p_k, b_k] & \text{om } f(p_k) < 0 \\ [a_k, p_k] & \text{om } f(p_k) > 0 \end{cases}$$

Bilden till höger illustrerar metoden. Observera att varje halvering behöver tecken till $f(p_k)$, men bara tecken som gäller. Konstruktionen ger

$$b_k - a_k = \frac{b_{k-1} - a_{k-1}}{2} = \ldots = \frac{b_1 - a_1}{2^k}$$



Konvergensen hos intervallhalveringsmetoden är långsam. *För varje steg vinner vi en binär siffra i noggrannhet.* Eftersom $10^{-1} \approx 2^{-3,3}$ vinner vi alltså i genomsnitt en decimal siffra på $3,3$ steg. Däremot är konvergenshastigheten helt oberoende av funktionen $f(x)$. Dessutom har vi

$$|p_k - p| \leq |b_k - a_k| \leq \frac{b-a}{2^{k+1}} \to 0$$

då $k \to \infty$. Alltså är konvergens global.

2.2. **Newton-Raphsons metod.** Idén bakom Newton-Raphsons metod för lösning av en ekvation $f(x) = 0$, är att approximera kurvan $y = f(x)$ med tangenten i punkten $(p_0, f(p_0))$. Låt p_1 vara abskissan för skärningspunkten mellan x-axeln och tangenten. Se bilden till höger. Fortsätter vi på så sätt kan vi få en talföljd $p_0, p_1, p_2, \ldots$

Att approximera kurvan $y = f(x)$ med tangenten i $(p_k, f(p_k))$ är detsamma som att ersätta funktionen med förstaordningstermerna i dess Taylorutveckling omkring $x = p_k$:

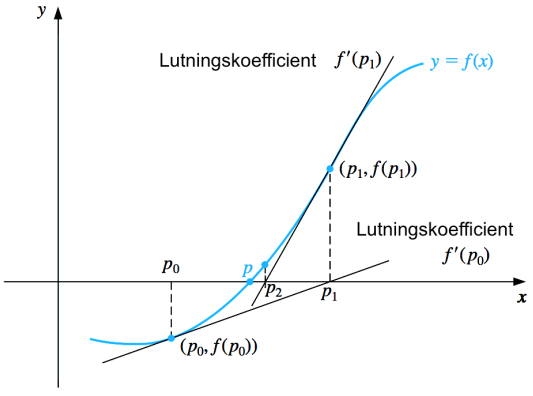
$$f(x) \approx f(p_k) + f'(p_k)(x - p_k)$$

Vi bestämmer p_{k+1} genom, för ett lämpligt val av $x^{(0)}$,

$$f(p_k) + f'(p_k)(p_{k+1} - p_k) = 0$$

$$\iff \quad p_{k+1} = p_k - \frac{f(p_k)}{f'(p_k)} \qquad \text{(NR)}$$

vilket är Newton-Raphsonsmetod, där vi antar att $f'(p_k) \neq 0$.



Normalt kommer p_1 att bli ett mycket bättre närmevärde till roten p än p_0. Det är vanligt att p_1 får nästan dubbelt så många riktiga siffror som p_0, men om p_0 är ett mycket dåligt närmevärde, så kan det hända att p_1 blir sämre än p_0.

Nu skall konvergensegenskaperna hos Newton-Raphsons metod studeras. Vi antar att funktionen $f(x)$ är två gånger kontinuerligt deriverbar samt att den sökta roten p är en enkelrot. Då är $f'(p) \neq 0$ och således $f'(x) \neq 0$ för alla x i en viss omgivning av roten p.

Taylorutveckling med resttermen ger

$$0 = f(p) = f(p_k) + (p - p_k)f'(p_k) + \frac{1}{2}(p - p_k)^2 f''(\xi),$$

för något ξ mellan p_k och p. Skriv om ekvationen under förutsättningen att $f'(p_k) \neq 0$

$$\left(p_k - \frac{f(p_k)}{f'(p_k)}\right) - p = \frac{1}{2}(p - p_k)^2 \frac{f''(\xi)}{f'(p_k)}$$

Låt $e_k := p_k - p$. Då

$$e_{k+1} = p_{k+1} - p = \frac{1}{2}e_k^2 \frac{f''(\xi)}{f'(p_k)}$$

$$\Longrightarrow \quad \frac{|e_{k+1}|}{|e_k|^2} = \frac{1}{2}\frac{|f''(\xi)|}{|f'(p_k)|} \to \frac{1}{2}\frac{|f''(p)|}{|f'(p)|} \tag{Q}$$

då $k \to \infty$. Antag nu att Newton-Raphsons metod ger en talföljd sådan att $\lim_{k\to\infty} p_k = p$. Då enligt ovan

$$\lim_{k\to\infty} \frac{|e_{k+1}|}{|e_k|^2} = C, \quad C = \frac{1}{2}\frac{|f''(p)|}{|f'(p)|}$$

och $C \neq 0$ om $f''(p) \neq 0$. Per definition är *Newton-Raphsons metod kvadratiskt konvergent.* Detta visar att Newton-Raphsons metod för enkelrötter i allmänhet ger en andra ordningens talföljd.

Antag nu att I är en omgivning av roten p sådan att

$$\frac{1}{2}\left|\frac{f''(y)}{f'(x)}\right| \leq M, \quad \text{för alla } x \in I, y \in I.$$

Om $p_k \in I$, så följer det då av (Q) att

$$|e_{k+1}| \leq Me_k^2 \quad \Longleftrightarrow \quad |Me_{k+1}| \leq (Me_k)^2$$

Antag att $|Me_0| < 1$ och att intervallet $[p - |e_0|, p + |e_0|]$ är en delmängd av I. Genom induktion kan vi visa att

$$|Me_{k+1}| \leq |Me_k|^2 \leq (M|e_0|)^{2^k}$$

Härav följer att Newton-Raphsons metod konvergerar under förutsättning att p_0 väljes tillräckligt nära roten p, dvs om

$$|Me_0| = M|p_0 - p| < 1.$$

Följande kommentar illustrerar den snabba konvergensen.

Kommentar. Även om $M|e_0|$ *endast ligger något under* 1 får vi efter några steg ett mycket snabbt avtagande av felet. Om t ex $M|e_0| = 0,9$ och $M = 1$ blir $|e_k|$ för $k = 1, 2, 3, ...$ begränsad av

$$0,81; 0,656; 0,44; 0.19; 0,036; 0,0013; 0,000016, ...$$

För $k \geq 6$ fördubblas approximativt antal signifikanta decimaler för varje iteration. Således är Newton-Raphsons metod en snabbt konvergent metod.

För att få en klarare bild av konvergensegenskaper hos Newton-Raphsons metod räknar vi några konkreta exempel.

Exempel. (i) Vi skall studera konvergens av Newton-Raphsons metod för att lösa ekvationen $x^2 = c > 0$, dvs $f(x) = x^2 - c = 0$.

$$p_{k+1} = p_k - \frac{p_k^2 - c}{2p_k} = \frac{1}{2}(p_k + c/p_k)$$

Vi kan dessutom bevisa att den konvergerar för alla $p_0 > 0$. Först notera att

$$p_{k+1} - \sqrt{c} = \frac{1}{2}(p_k + c/p_k) - \sqrt{c} = \frac{1}{2p_k}(p_k^2 - 2p_k\sqrt{c} + c) = \frac{1}{2p_k}(p_k - \sqrt{c})^2 \geq 0$$

dvs $p_{k+1} \geq \sqrt{c}$ för alla $k \geq 0$ och

$$p_{k+1} - p_k = -\frac{1}{2p_k}(p_k^2 - c) \leq 0$$

Så följden $\{p_k\}$ är strängt avtagande och nedåt begränsad, $p_1 \geq \cdots \geq \sqrt{c}$. Därför konvergerar iterationen för alla $p_0 > 0$.

Notera att om $p_0 < \sqrt{c}$ så kommer $p_1 > \sqrt{c}$ efter en direkt uträkning.

(ii) Låt $f(x) = \arctan x$. Då $p = 0$ är en lösning till $f(x) = 0$. Newton-Raphsons interation är

$$p_{k+1} = p_k - (1 + p_k^2)\arctan p_k.$$

Om vi väljer p_0 sådant att

$$\arctan |p_0| > \frac{2|p_0|}{1 + p_0^2}$$

då divergerar talföljden $\{|p_k|\}$: $\lim_{k\to\infty} |p_k| = \infty$.

(iii) Det kan kanske ge intryck att Newton-Raphsons metod konvergerar om man har goda startvärden. Det är tyvärr inte alltid så.

Om vi använder Newton-Raphsons metod för lösandet av roten till $\sqrt[3]{x} = 0$ blir vi ganska besvikna. Här är iterationen

$$p_{k+1} = p_k - \sqrt[3]{p_k}(\frac{1}{3}\sqrt[3]{p_k^2}) = p_k - 3p_k = -2p_k$$

Det är en linjär iteration och vi inser direkt att talföljden genererad av denna iteration oscillerar och divigerar för alla $p_0 \neq 0$.

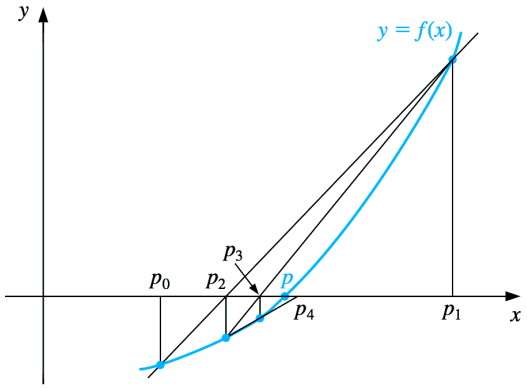
2.3. **Sekantmetoden.** Newton-Raphsons metod behöver derivatan $f'(p_k)$ i varje steg. Vi kan i stället approximera

$$f'(p_k) \approx \frac{f(p_k) - f(p_{k-1})}{p_k - p_{k-1}}$$

vilket ger

$$p_{k+1} = p_k - f(p_k)\frac{p_k - p_{k-1}}{f(p_k) - f(p_{k-1})}$$

med två initiala gissningar p_0 och p_1 och $f(p_k) \neq f(p_{k-1})$.



För enkelhets skull inför vi notationen $f_k := f(p_k)$. Nu skall vi härleda ett samband mellan fel i konsekutiva närmevärden genom Newtons interpolationsformel med resttermen (se en godtycklig litteratur om approximation eller numerisk analys t ex [7])

$$f(x) = f_k + (x - p_k)f[p_{k-1}, p_k] + (x - p_{k-1})(x - p_k)\frac{1}{2}f''(\xi) \qquad \text{(NI)}$$

där

$$f[p_{k-1}, p_k] := \frac{f_k - f_{k-1}}{p_k - p_{k-1}},$$

och ξ tillhör det minsta intervall som innehåller x, p_{k-1}, p_k. Försummas här resttermen, erhålles sekantens ekvation och p_{k+1} uppfyller därför ekvationen

$$0 = f_k + (p_{k+1} - p_k) f[p_{k-1}, p_k].$$

Sätt nu $x = p$ i Newtons interpolationsformel (NI) och subtrahera ekvationen för p_{k+1}. Då $f(p) = 0$ erhåller vi

$$(p - p_{k+1}) f[p_{k-1}, p_k] + \frac{1}{2}(p - p_{k-1})(p - p_k) f''(\xi) = 0$$

Genom att applicera medelvärdessatsen får vi

$$f[p_{k-1}, p_k] = f'(\xi')$$

där ξ' ligger i det minsta intervall som innehåller p_k, p_{k-1}. Varav följer sambandet

$$e_{k+1} = \frac{f''(\xi)}{2f'(\xi')} e_k e_{k-1}.$$

Av detta samband följer att sekantmetoden konvergerar för tillräckligt goda startvärden p_0 och p_1, om $f'(p) \neq 0$ och $f(x)$ är två gånger kontinuerligt deriverbar. Observera att om vi låter $p_{k-1} \to p_k$ övergår formeln för ett steg med sekantmetoden i formeln för ett steg med Newton-Raphsons metod och vi återfinner formeln (Q) i föregående avsnittet.

Antag nu att sekantmetoden konvergerar. När k är stort gäller det att $\xi \approx p$, $\xi' \approx p$, och

$$|e_{k+1}| \approx C|e_k| \cdot |e_{k-1}|,$$

dä $C = \frac{1}{2}\frac{|f''(p)|}{|f'(p)|}$. Nu försöker vi bestämma ordningen för sekantmetoden med en heuristisk betraktelse utan ett strikt bevis med syfte på att ge insikt. Vi ansätter på försök

$$|e_{k+1}| \approx K|e_k|^\nu, \quad |e_k| \approx K|e_{k-1}|^\nu.$$

Insättning av detta i föregående ekvationen ger

$$K|e_k|^\nu \approx C|e_k| K^{-1/\nu} |e_k|^{1/\nu}$$

Detta gäller endast om $\nu = 1 + 1/\nu$, dvs, $\nu = \frac{1}{2}(1 \pm \sqrt{5})$, och $C = K^{1+1/\nu} = K^\nu$. Man kan visa att det till beloppet mindre roten $\frac{1}{2}(1 - \sqrt{5})$ kan lämnas utan avseende, och att

$$|e_{k+1}| \approx C^{1/\nu}|e_k|^\nu, \quad \nu = \frac{1}{2}(1 + \sqrt{5}) \quad (k \gg 1).$$

3. Fixpunkts iteration och allmän teori för iterationsmetoder

Newton-Raphsons metod och sekantmetoden kan uppfattas som specialfall av följande mycket allmänna iterationsmetoder som nämndes i första avsnittet. Alltså är p_{k+1} bestämd av

$$p_{k+1} = \phi(p_k, p_{k-1}, ..., p_{k-m+1})$$

givet de första m start gissningarna $p_0, ., , , .p_{m-1}$. Det kallas även m-punkts iterationsmetod.

Exempel. (i) Newton-Raphsons metod är en-punkts iterationsmetod då

$$\phi_{NR}(x) := x - \frac{f(x)}{f'x}$$

(ii) Sekantmetoden är en två-punkts iterationsmetod då

$$\phi_S(x_1, x_2) := x_1 - f(x_1)\frac{x_1 - x_2}{f(x_1) - f(x_2)}$$

Den allmänna teorin för iterationsmetoder är enklaste för $m = 1$. I detta fall har vi

$$p_{k+1} = \phi(p_k).$$

Anmärkning. Vi poängterar här att synsätt på en- eller flerpunkts iterationer är relativt. Alla m-punkts iterationsmetoder kan formuleras som ett system av enpunkts iterationsmetod om vi inför tillståndsvektor på följande sätt. Sätt $x^{(k)} = \begin{pmatrix} p_{k-m+1} & \cdots & p_{k-1} & p_k \end{pmatrix}^T$. Då gäller att

$$x^{(k+1)} = \varphi(x^{(k)}) := \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{pmatrix} x^{(k)} + \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \phi(x^{(k)}) \end{pmatrix}$$

T ex Sekantmetoden kan iterationen skrivas på matrisformen

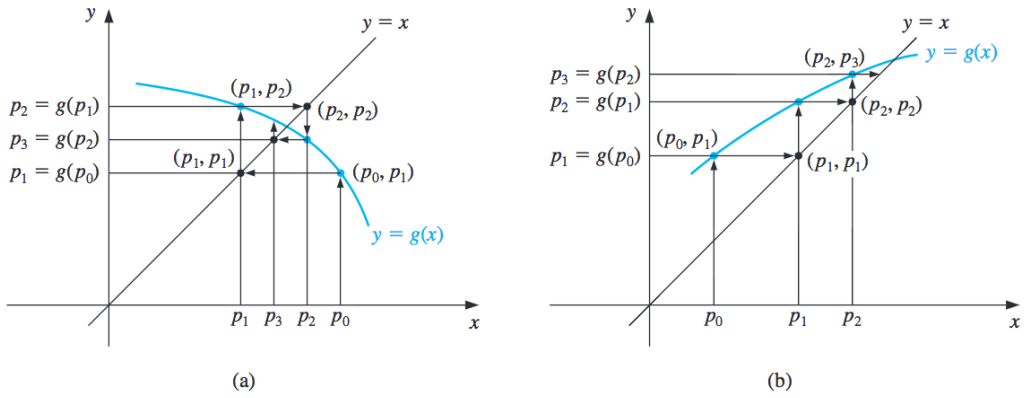
$$x^{(k+1)} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} x^{(k)} + \begin{pmatrix} 0 \\ \phi_S(x^{(k)}) \end{pmatrix}$$

Således kan flerpunkts iterationsmetoder betraktas som enpunkts iterationsmetoder för att lösa system av icke-linjära ekvationer.

Enpunkts iterationer kallas i litteratur fixpunkts iteration eftersom vi får $p = g(p)$, vilket innebär att p är en fixpunkt av avbildningen g, om talföljden konvergerar mot p. I detta avsnitt ska vi studera allmänna teorin för denna klass av iterationsmetoder:

$$p_{k+1} = g(p_k), \tag{F}$$

med startvärdet p_0. Metoden är illustrerad nedan.



Generellt får vi

- linjär konvergens ty

  $$|p_{k+1} - p| = |g(p_k) - g(p)| = |g'(\xi)||p_k - p| \text{ för något } \xi$$

  mellan $p_k$och p.
- om $|g'(x)| \leq m < 1$ för alla $x \in [a, b]$ får vi konvergens

Vi skall nu bevisa det sista påståendet. Men innan vi går vidare för strikt analys av konvergens visar vi med följande exempel hur iterationsfunkionen kan konstrueras.

Exempel. (i) Hitta den positiva lösningen till ekvationen $x^2 = c$, där $c > 0$ är ett givet tal. Vi skriver om ekvationen

$$x^2 = c \Leftrightarrow x = \frac{c}{x} \Leftrightarrow 2x = \frac{c}{x} + x \Leftrightarrow x = \frac{1}{2}\left(\frac{c}{x} + x\right)$$

Vi kan välja

$$g(x) = \frac{1}{2}\left(\frac{c}{x} + x\right)$$

Så en iterationsmetod är

$$p_{k+1} = \frac{1}{2}\left(\frac{c}{p_k} + p_k\right).$$

En snabb beräkning visar att det är Newton-Raphons metod för att lösa ekvationen $f(x) = 0$ där $f(x) = x^2 - c$. Derivering av $g(x)$ ger

$$g'(x) = \frac{1}{2}(1 - \frac{c}{x^2}) \quad \Longrightarrow \quad |g'(x)| < 1, \forall x : x^2 < c$$

Så iterationen konvergerar och den går mot $\sqrt{c}$.

Om vi istället sätter $g(x) = \frac{c}{x}$ eftersom $x = g(x)$ är precis samma som den givna ekvationen, då ger iterationen en följd, som studsar fram och tillbaka mellan p_0 (för jämna k) och $2/p_0$ (för udda k). Det vill säga talföljden konvergerar inte.

(ii) Ekvationen $x^3 - x - 5 = 0$ kan exempelvis omskrivas som $x = g(x)$. Den har en rot i närheten av 1.9. Vi gör tre olika försök.

(a) $x^3 - x - 5 = 0 \Longleftrightarrow x = x^3 - 5$. Så vi väljer $g_1(x) = x^3 - 5$;
(b) $x^3 - x - 5 = 0 \Longleftrightarrow x = \sqrt[3]{x+5}$. Vi väljer $g_2(x) = \sqrt[3]{x+5}$;
(c) $x^3 - x - 5 = 0 \Longleftrightarrow x = \frac{5}{x^2-1}$. Vi väljer $g_3(x) = \frac{5}{x^2-1}$.

Vi skall bevisa att det är bara (b) som genererar konvergent talföljd lite senare.

Dessa exempel visar att det finns många olika individuella sätt att behandla dem, vissa är bra, vissa är mindre bra eller dåliga. Det finns inte en universal metod för alla.

Nu skall vi bevisa följande sats.

Sats. ([7]) *Låt $J = \{x : |x - p| \leq \rho\}$. Antag att $x = g(x)$ har en rot p. Antag vidare att $g'(x)$ existerar i J och $|g'(x)| \leq d < 1$. Då gäller, för alla $p_0 \in J$ att*

(1) *$p_k \in J$, $k = 0, 1, 2, ...$*
(2) *$p_k \to p$ då $k \to \infty$,*
(3) *p är den enda rot i J till $x = g(x)$.*

Bevis. (1) Vi har

$$\begin{aligned} p_k - p &= g(p_{k-1}) - g(p) \quad \text{medelvärdessatsen} \\ &= g'(\xi)(p_{k-1} - p) \quad \text{för något } \xi \in J \\ \Rightarrow \quad |p_k - p| &\leq d|p_{k-1} - p| \leq d\rho < \rho \Rightarrow x^{(k)} \in J. \end{aligned}$$

(2) Induktivt kan vi upprepa olikheten $|p_k - p| \leq d|p_{k-1} - p|$ vidare:

$$|p_k - p| \leq d|p_{k-1} - p| \leq d^2|p_{k-2} - p| \leq \cdots \leq d^k|p_0 - p| \to 0$$

ty $d < 1$. $\Rightarrow p_k \to p$.

(3) Antag att det fanns en annan rot $p' \in J$. Då har vi

$$\begin{aligned} |p' - p| = |g(p') - g(p)| &= |g'(\xi)||p' - p| \quad \text{medelvärdessatsen} \\ \Rightarrow |p' - p| &\leq d|p' - p| < |p' - p| \Rightarrow p' = p. \end{aligned}$$

Av denna motsägelse följer påståendet (3). □

Anmärkningar:

(i) Existens av roten p behöver egentligen inte antas med lite modifiering av satsen. Vi kan lätt bevisa en mer allmän sats:

Sats. *Antag att $g(x)$ är kontinuerlig på det slutna intervallet $[a,b]$. Antag vidare att $[a,b]$ avbildas i $[a,b]$ av g; dvs, $\forall x \in [a,b]$ $g(x) \in [a,b]$. Då finns en punkt $c \in [a,b]$ så att $g(c) = c$.*

Bevis. Sätt $f(x) := x - g(x)$. Så den är kontinuerlig på $[a,b]$. Eftersom $g(x) \in [a,b]$ för alla $x \in [a,b]$, $g(b) \in [a,b]$. Då $g(b) \leq b \Leftrightarrow f(b) \geq 0$. På samma sätt $f(a) \leq 0$. Enligt satsen om mellanliggandevärden för kontinuerliga funktioner på slutna intervall måste det finnas $c \in [a,b]$ så att $f(c) = 0 \Leftrightarrow c = g(c)$. □

(ii) Om målet är att approximera p med en tolerans $\epsilon > 0$ dvs $|p_k - p| \leq \epsilon$ så kan vi se hur stort k som behövs för att uppnå målet. Vi har sedan tidigare uppskattningen

$$|p_k - p| \leq d^k |p_0 - p|, \quad k \geq 1$$

Eftersom vi inte känner till p vill vi skriva om $|p_0 - p|$ så att vi kan använda de kända talen i uppskattningen. Detta görs på följande (*standard*) sätt.

$$|p_0 - p| \leq |p_0 - p_1| + |p_1 - p| = |p_0 - p_1| + |g(p_0) - g(p)| \leq |p_0 - p_1| + d|p_0 - p|$$

$$\Leftrightarrow |p_0 - p| \leq \frac{|p_1 - p_0|}{1-d} \Rightarrow |p_k - p| \leq \frac{d^k}{1-d}|p_1 - p_0|$$

För att få $|p_k - p| \leq \epsilon$ kan vi söka K så att $\frac{d^k}{1-d}|p_1 - p_0| \leq \epsilon$, vilket ger

$$d^k \leq \frac{\epsilon(1-d)}{|p_1 - p_0|} \Rightarrow k \geq \frac{1}{\ln d} \ln \left[\frac{(1-d)\epsilon}{|p_1 - p_0|} \right] =: K,$$

Dvs, det behövs K steg för att uppnå det önskade tolerans.

(iii) På liknande sätt som görs i beviset av satsen kan vi visa att det finns ett $k > 0$ sådant att $g(p_k) \notin J$, $x \neq p$, om $|g'(p)| > 1$. Detta kan enkelt bevisa genom linearisering av iterationsfuktionen i p eftersom $|g'(p)1| > 1$ förökar fel varje steg med faktorn $|g'(p)|$. Det ger oss ett villkor för att avgöra om en fixpunkts iteration divergerar. Nu återgår vi till Exempel (ii) före satsen.

Exempel. (återbesök) Iterationsfunktionen $g_2(x) = \sqrt[3]{x+5}$ genererar en konvergent talföljd eftersom

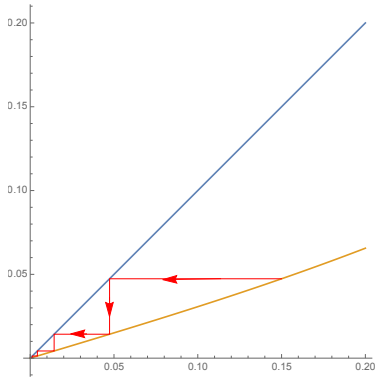
$$|g_2'(x)| = \frac{1}{3} \frac{1}{\sqrt[3]{(x+5)^2}} < 1, \forall x$$

Iterationsfunktionen $g_1(x)$ ger en divergent talföljd eftersom $g_1'(1.9) = 3 \cdot (1.9)^2 > 1$. Till sist

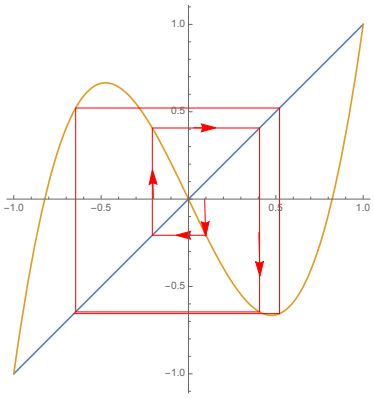
$$|g_3'(1.9)| = \frac{19}{1.9^2 - 1} > \frac{19}{(2^2-1)^2} = \frac{19}{9} > 1$$

vilket visar att talföljden genererad av g_3 divergerar.

(iv) I termer av teori för dynamiska system kallas fixpunkten p med egenskapen $|f'(p)| > 1$ och $|f'(p)| < 1$ för *hyperboliska fixpunkter*. Den första typen kallas för aysmptotiska respektive den andra för repellera. Dessa egenskaper kan undersökas grafiskt. Vi ritar upp funktionen $f_a(x) = (1-a)x + ax^3$ ([5]) för (a) $a = 0.7$ då fixpunkt 0 är en attraktion och $0 < f_a'(0) < 1$; (b) $a = 3.1$ då fixpunkt 0 är en repellor och $-1 < f_a'(0) < 0$;

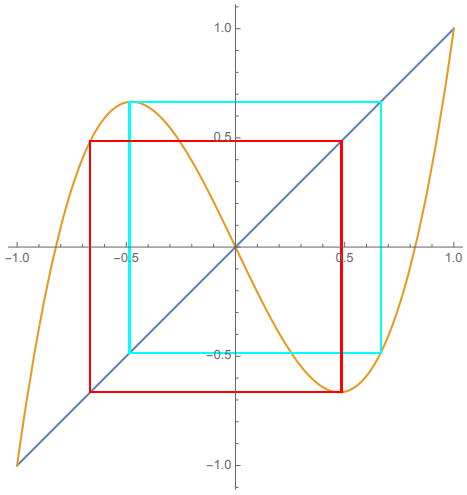


(a)



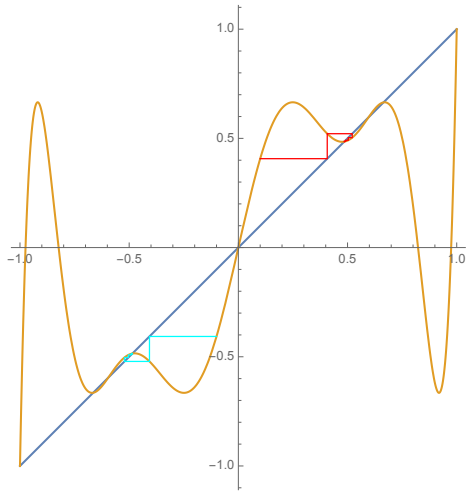
(b)

(v) Vi poängterar även att dynamiska beteenden inte är begränsad till konvergens, divergens, eller oscillation. Vi kan även se mer komplicerad beteende som kaos. Vi gör en så-kallad bifurkations diagram för iterationsfunktionen i föregående anmärkning. Bilderna är genererad med `GraphicalAnalysis.m` i Mathematica [6] för $a = 3.1$.

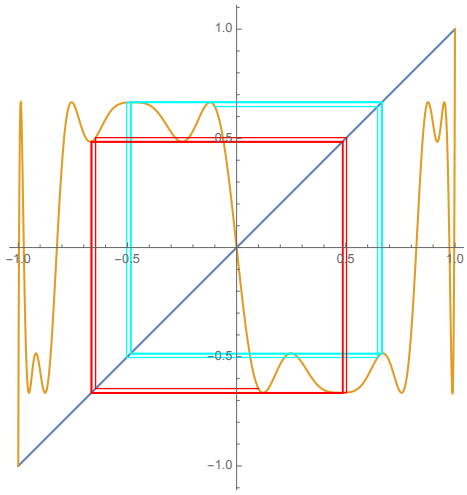


(c)

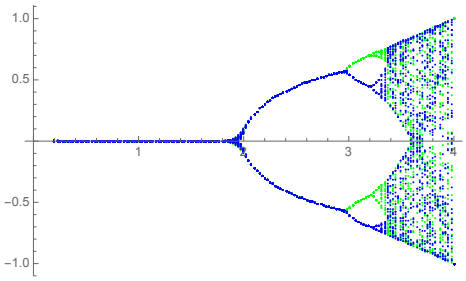
Bilden till vänster visar att följderna genererad från 0.1 respektive -0.1 har ett återkommande beteende. Om vi gör en grafisk analys för iterationen $f_a^2 := f_a(f_a(\cdot))$ respektive $f_a^3 = f_a(f_a(f_a(\cdot)))$ ser vi att de går mot fixpunkterna $x = f_a^2(x)$ (Bild (d)) respektive $x = f_a^3(x)$ (Bild (e)). Dessa punkter kallas för *periodiska punkter*. För att kunna se hur variation av a påverkar dynamiska beteende ritar vi upp ett *bifurkations diagram* (Bild (f)), vilket illustrerar att iterationen har en fixpunkt mellan $0 < a < 2$ sedan fördubblas perioden mellan $2 < a < 3$ och perioden fortsätter fördubblas tills a är nära 3.4 då är det svårt att se hur punkterna rör sig (Bild (g)). Studier av sådana beteenden utelämnas.



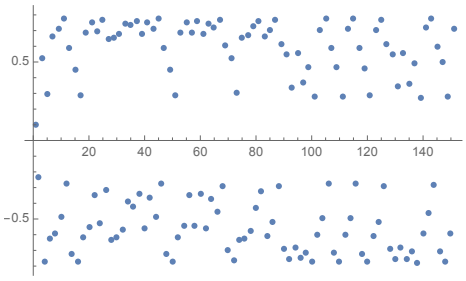
(d)



(e)



(f)



(g)

4. KONVERGENSORDNING

I beviset av konvergensegenskaper hos fixpunktsmetoden har vi

$$p_k - p = g'(\xi)(p_{k-1} - p), \text{ för något } \xi \in J,$$

vilket innebär att *de allmänna iterationsmetoderna har linjär konvergens* om inte g är speciellt konstruerad. Men vi vet att Newton-Raphsons metod är ett speciellt fall och den konvergerar kvadratiskt. Låt oss analysera närmare denna metod och försöka finna vad som ligger bakom högre konvergensordning, dvs vilka egenskaper hos $g = x - \frac{f(x)}{f'(x)}$ som påverkar konvergensordning. Först är det inte svårt att inse att

$$g'(x) = \frac{f(x)f''(x)}{(f'(x))^2} \Rightarrow g'(p) = 0 \quad (p \text{ är en rot till } f(x) = 0)$$

om $f'(x) \neq 0$ och f är två gånger kontinuerligt deriverbar. Sätt detta villkor i iterationen $p_{k+1} = g(p_k)$ och Taylorutveckla $g(p_k)$ i p

$$\begin{aligned} p_{k+1} = g(p_k) &= g(p) + g'(p)(p_k - p) + \frac{g''(\xi)}{2}(p_k - p)^2 \\ &= p + \frac{g''(\xi)}{2}(p_k - p)^2, \ (\text{ något } \xi \text{ i det minsta intervallet som har } p_k, p) \\ \Leftrightarrow \quad p_{k+1} - p = \frac{g''(\xi)}{2}(p_k - p)^2 \ &\Rightarrow \ \lim_{k\to\infty} \frac{|p_{k+1} - p|}{|p_k - p|^2} = \frac{g''(p)}{2} \neq 0 \end{aligned}$$

Alltså har Newton-Raphsons metod kvadratisk konvergens.

Observera att det är $g'(p) = 0$ som har höjt konvergensordningen eftersom termen $(p_k - p)$ försvinner i Taylorutvecklingen. Så det är inte svårt att bevisa att om $\frac{d^j g}{dx^j}(p) = 0$ för alla $j = 1, ...\nu - 1$ ger iterationsmetoden konvergensordning ν. Detta betyder att *om* $f''(p) = 0$ *så är Newton-Raphsons av minst ordningen tre.*

Man kan kanske tro att det är svårt att konstruera iterationsmetoder av godtyckligt hög ordning för lösandet av ekvationen $f(x) = 0$. Det finns faktiskt flera generella metoder för detta. Några förslag för konstruktion av g för högre ordnings konvergens finns i boken [7]. Här är ett exempel.

Exempel. (En iterationsmetoder med konvegensordning 18) Vi härleder metoden utan strikt analys. Låt f vara en tillräckligt reguljär funktion och $f(x) = 0$ ha en enkelrot α. Vi delar problemet i tre steg

(i) Vi vet att Newton-Raphsons metod är en första ordnings approxmiation för f kring $x = \alpha$; dvs vi löser ut h ur ekvationen

$$0 = f(x + h) \approx f(x) + hf'(x),$$

vilket är $h = -f(x)/f'(x)$ förutsatt att $f'(x) \neq 0$ i roten α. Då har vi

$$x^{(k+1)} = \phi(x^{(k)}),$$

där $\phi(x) = x - f(x)/f'(x)$. Vi skall härleda en iterationsmetod som har kubisk konvergensordning med samma idé, dvs bestäm funktionen ϕ_1 så att $x^{(k+1)} = \phi_1(x^{(k)})$ konvergerar kubiskt.

(ii) Vi ska härleda en iterationsmetod av kubisk konvergensordning med hjälp av funktionen $g(x) = f(x)/\sqrt{|f'(x)|}$: $x^{(k+1)} = \phi_2(x^{(k)})$. Vi ska hitta om det finns någon relation mellan de här två metoderna.

(iii) Vi bevisar att konvergensordning är 18 om vi kombinerar Newton-Raphsons metod med (i) och (ii):

$$w^{(k)} = \phi(x^{(k)}) \quad y^{(k)} = \phi_2(w^{(k)}) \quad x^{(k+1)} = \phi_1(y^{(k)}).$$

Vi demonstrerar hur dessa fungerar i tre steg:

(A) Antag att f är två gånger kontinuerligt deriverbar. Newtons iteration kan ses som en första-ordnings metod (eller lineariserings metod). Det är möjligt att gå ett steg längre och skriva en extra term i Taylorutveckling:

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2}f''(x) + O(h^3).$$

Nu söker vi h så att

$$f(x+h) = 0 \approx f(x) + hf'(x) + \frac{h^2}{2}f''(x).$$

Vi tar den minsta lösningen för h (vi behöver anta att $f'(x)$ och $f''(x)$ inte är lika med noll)

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

Eftersom vi söker ett nollställe till f så kan vi betrakta $f(x)$ litet. Det är därför onödigt att använda kvadratroten. I stället kan vi Taloyutveckla den till andra-ordningen:

$$1 - \sqrt{1-\eta} = \frac{\eta}{2} + \frac{\eta^2}{8} + O(\eta^3), \text{ där } \eta = \frac{2f(x)f''(x)}{f'(x)^2}$$

$\Longrightarrow$

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

Försumma högre ordningstermer får en iterationsmetod

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

Sätt

$$\phi_1(x) = x - \frac{f(x)}{f'(x)}\left(1 + \frac{f(x)f''(x)}{2f'(x)^2}\right).$$

Denna iterationsmetod har kubisk konvergensordning eftersom $\phi_1'(\alpha) = \phi_1''(\alpha) = 0$, vilket följer från följande beräkningar

$$\phi_1'(x) = -\frac{f(x)^2\left(f^{(3)}(x)f'(x) - 3f''(x)^2\right)}{2f'(x)^4}$$
$$\phi''(x) = -\frac{f(x)\left(12f(x)f''(x)^3 + 2f^{(3)}(x)f'(x)^3 + f'(x)^2\left(f(x)f^{(4)}(x) - 6f''(x)^2\right) - 9f(x)f^{(3)}(x)f'(x)f''(x)\right)}{2f'(x)^5}$$

(B) Notera att $g(x) = f(x)/\sqrt{|f'(x)|} = 0$ och $f(x) = 0$ har samma rot. Applicera Newton-Raphsons metod på g fås

$$x^{(k+1)} = x^{(k)} - \frac{g(x^{(k)})}{g'(x^{(k)})}.$$

Så $\phi_2(x) = x - \frac{g(x)}{g'(x)}$. För att undersöka konvergensordning beräknar vi $g'(x)$

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

Insättningen av detta uttryck av $g'(x)$ och $g(x)$ i $\phi_2(x)$ ger

$$\phi_2(x) = x - \frac{2f(x)f'(x)}{2f'(x)^2 - f(x)f''(x)}$$

Metoden konvergerar kubiskt på grund av att $\phi_2'(\alpha) = \phi_2''(\alpha) = 0$, vilka fås av

$$\phi_2'(x) = \frac{f(x)^2\left(3f''(x)^2 - 2f^{(3)}(x)f'(x)\right)}{\left(f(x)f''(x) - 2f'(x)^2\right)^2}$$
$$\begin{aligned}\phi_2''(x) = 2f(x)\Big(&f(x)^2f^{(3)}(x)f''(x)^2 - 4f^{(3)}(x)f'(x)^4 + f'(x)^3\left(6f''(x)^2 - 2f(x)f^{(4)}(x)\right)\\ &+ 12f(x)f^{(3)}(x)f'(x)^2f''(x) + f(x)f'(x)\left(-2f(x)f^{(3)}(x)^2 - 12f''(x)^3 + f(x)f^{(4)}(x)f''(x)\right)\Big) /\\ &\left(2f'(x)^2 - f(x)f''(x)\right)^3\end{aligned}$$

För att hitta en relation till (i) kan vi skriva om $\phi_2(x)$ till följande form:

$$\phi_2(x) = x - \frac{f(x)}{f'(x)}\left(1 - \frac{f(x)}{f'(x)}\cdot\frac{f''(x)}{2f'(x)}\right)^{-1}$$

Applicera Taylorutvecklingen $(1-\eta)^{-1} = 1 + \eta + O(\eta^2)$ på andra termen i $\phi_2(x)$ får

$$\phi_2(x) = x - \frac{f(x)}{f'(x)}\left(1 + \frac{f(x)f''(x)}{2f'(x)^2}\right)$$

vilket är ϕ_1

Kommentar. Metoden som härleds i (i) är från Householder [4] medan den i (ii) är från Halley [8].

Kommentar. Ett annat sätt att konstruera av högre konvergensordning iteration är att skriva

$$x^{(k+1)} = x^{(k)} + h_k + a_2^{(k)}\frac{h_k^2}{2!} + a_3^{(k)}\frac{h_k^3}{3!} + \cdots$$

där $h_k = -f(x^{(k)})/f'(x^{(k)})$ är given av Newton-Raphsons metod och $(a_2^{(k)}, a_3^{(k)}, ...)$ är reella parametrar som vi kan estimera för att minimera värdet på $f(x^{(k+1)})$:

$$f(x^{(k+1)}) = f\left(x^{(k)} + h_k + a_2^{(k)}\frac{h_k^2}{2!} + a_3^{(k)}\frac{h_k^3}{3!} + \cdots\right)$$

Vi antar att f är tillräckligt reguljär och $h_k + a_2^{(k)}\frac{h_k^2}{2!} + a_3^{(k)}\frac{h_k^3}{3!} + \cdots$ är litet. Taylorutveckla f i $x^{(k)}$

$$f(x^{(k+1)}) = f(x^{(k)}) + \left(h_k + a_2^{(k)}\frac{h_k^2}{2!} + a_3^{(k)}\frac{h_k^3}{3!} + \cdots\right) f'(x^{(k)}) + \\ + \left(h_k + a_2^{(k)}\frac{h_k^2}{2!} + a_3^{(k)}\frac{h_k^3}{3!} + \cdots\right)^2 \frac{f''(x^{(k)})}{2} + \cdots$$

och eftersom $f(x^{(k)}) + h_k f'(x^{(k)}) = 0$ har vi

$$f(x^{(k+1)}) = \left(a_2^{(k)} f'(x^{(k)}) + f''(x^{(k)})\right)\frac{h_k^2}{2!} + \left(a_3^{(k)} f'(x^{(k)}) + 3a_2^{(k)} f''(x^{(k)}) + f'''(x^{(k)})\right)\frac{h_k^3}{3!} + O(h_k^4)$$

Ett bra val för $a_i^{(k)}$ är att kancellera så många termer som möjligt så vi kan t ex välja

$$a_2^{(k)} = -\frac{f''(x^{(k)})}{f'(x^{(k)})}$$
$$a_3^{(k)} = \frac{-f'(x^{(k)})f^{(3)}(x^{(k)}) + 3f''(x^{(k)})^2}{f'(x^{(k)})^2}$$
$$a_4^{(k)} = \frac{-f'(x^{(k)})^2 f^{(4)}(x^{(k)}) + 10f'(x^{(k)})f''(x^{(k)})f^{(3)}(x^{(k)}) - 15f''(x^{(k)})^2}{f'(x^{(k)})^3}$$
$$...$$

Slutligen blir iterationen

$$x^{(k+1)} = x^{(k)} + h_k\left(1 + a_2^{(k)}\frac{h_k}{2!} + a_3^{(k)}\frac{h_k^2}{3!} + \cdots\right)$$
$$= x^{(k)} - \frac{f(x^{(k)})}{f'(x^{(k)})}\left(1 + \frac{f''(x^{(k)})}{2f'(x^{(k)})}\frac{f(x^{(k)})}{f'(x^{(k)})} + \frac{3f''(x^{(k)})^2 - f'(x^{(k)})f'''(x^{(k)})}{3!f'(x^{(k)})^2}\left(\frac{f(x^{(k)})}{f'(x^{(k)})}\right)^2 + \cdots\right)$$

Om vi slutar i $a_3^{(k)}$ och sätter $a_4^{(k)} = a_5^{(k)} = ... = 0$ har vi fjärdeordningens metod:

$$x^{(k+1)} = x^{(k)} - \frac{f(x^{(k)})}{f'(x^{(k)})}\left(1 + \frac{f''(x^{(k)})}{2f'(x^{(k)})}\frac{f(x^{(k)})}{f'(x^{(k)})} + \frac{3f''(x^{(k)})^2 - f'(x^{(k)})f'''(x^{(k)})}{3!f'(x^{(k)})^2}\left(\frac{f(x^{(k)})}{f'(x^{(k)})}\right)^2\right)$$

Om vi dessutom försummar $a_3^{(k)}$ får vi tillbaka Householders kubiska metod.

Det är också möjligt att hitta uttryck för $(a_4^{(k)}, a_5^{(k)}, ...)$ och definiera femte-, sjätte-, sjundeordningens iterationsmetoder.

(C) Nu ska vi kombinera Newton-Raphsons metod, (i) och (ii) i tre steg: givet $x^{(0)}$, för $k = 0, 1, 2, 3, ...$

$$w^{(k)} = x^{(k)} - \frac{f(x^{(k)})}{f'(x^{(k)})}$$
$$y^{(k)} = w^{(k)} - \frac{2f(w^{(k)})f'(w^{(k)})}{2f'(x^{(k)})^2 - f(w^{(k)})f''(w^{(k)})}$$
$$x^{(k+1)} = y^{(k)} - \frac{f(y^{(k)})}{f'(y^{(k)})}\left(1 + \frac{f(y^{(k)})f''(y^{(k)})}{2f'(y^{(k)})^2}\right).$$

Notera att det är en ganska vanlig metod i numeriska metoder för att förbättra algoritm. En sådan kombination kallas *predictor-corrector metod*. I det här fallet används Newton-Raphsons metod som en predictor och metoderna (i) och (ii) som en corrector.

Vi undersöker konvergensordning av denna iteration. Sätt $e_k = x^{(k)} - \alpha$. Vi antar att f är tillräckligt reguljär. Taylorutveckla $f(x^{(k)})$ och $f'(x^{(k)})$ i α:

$$f(x^{(k)}) = f(\alpha) + e_k f'(\alpha) + \frac{e_k^2}{2!} f''(\alpha) + \frac{e_k^3}{3!} f'''(\alpha) + \frac{e_k^4}{4!} f^{(4)}(\alpha) + \cdots$$

$\Longrightarrow$

$$f(x^{(k)}) = f(\alpha)[e_k + c_2 e_k^2 + c_3 e_k^3 + c_4 e_k^4 + \cdots] \tag{1}$$

$$f'(x^{(k)}) = f'(\alpha)[1 + 2c_2 e_k + 3c_3 e_k^3 + 4c_4 e_k^3 + \cdots] \tag{2}$$

där $c_k = \frac{1}{k!}\frac{f^{(k)}(\alpha)}{f'(\alpha)}$, $k = 2, 3, ...$ Från (1) och (2) har vi

$$w^{(k)} = \alpha + c_2 e_k^2 + 2(c_3 - c_2^2) e_k^3 + \cdots \tag{3}$$

Insättningen av $W = w^{(k)} - \alpha$ i (3) ger

$$W = c_2 e_k^2 + 2(c_3 - c_2^2) e_k^3 + \cdots \tag{4}$$

Taylorutveckla $f(w^{(k)}), f'(w^{(k)}), f''(w^{(k)})$ i α:

$$f(w^{(k)}) = f'(\alpha)[W + c_2 W^2 + c_3 W^3 + c_4 W^4 + \cdots] \tag{5}$$

$$f'(w^{(k)}) = f'(\alpha)[1 + 2c_2 W + 3c_3 W^2 + 4c_4 W^3 + \cdots] \tag{6}$$

$$f''(w^{(k)}) = f'(\alpha)[2c_2 + 6c_3 W + 12c_4 W^2 + \cdots] \tag{7}$$

Kombinera (2)-(7) fås

$$y^{(k)} = \alpha + (c_2^2 - c_3) W^3 \tag{8}$$

Vidare utvecklar vi $f(y^{(k)}), f'(y^{(k)}), f''(y^{(k)})$ i α:

$$f(y^{(k)}) = f'(\alpha)[(c_2^2 - c_3)W^3 + c_2((c_2^2 - c_3)W^3)^2 + c_3((c_2^2 - c_3)W^3)^3 + \cdots] \tag{9}$$

$$f'(y^{(k)}) = f'(\alpha)[1 + 2c_2((c_2^2 - c_3)W^3) + 3c_3((c_2^2 - c_3)W^3)^2 + 4c_4((c_2^2 - c_3)W^3)^3 + \cdots] \tag{10}$$

$$f''(y^{(k)}) = f'(\alpha)[2c_2 + 6c_3((c_2^2 - c_3)W^3) + 12c_4((c_2^2 - c_3)W^3)^2 + \cdots] \tag{11}$$

(8)-(11) tillsammans med (3) ger

$$\begin{aligned} x^{(k+1)} &= \alpha + (c_2^2 - c_3)W^3 - [(c_2^2 - c_3)W^3 + (-2c_2^2 + c_3)((c_2^2 - c_3)W^3)^3 + \cdots] \\ &= \alpha + (2c_2^2 - c_3)(c_2^2 - c_3)^3 W^9 + \cdots \\ &= \alpha + c_2^9(2c_2^2 - c_3)(c_2^2 - c_3)^3 e_k^{18} + O(e_k^{19}) \end{aligned}$$

$\Longrightarrow$

$$e_{k+1} = c_2^9(2c_2^2 - c_3)(c_2^2 - c_3)^3 e_k^{18} + O(e_k^{19}).$$

Det visar att iterationen med tre steg har konvergensordning 18.

5. Att accelera konvergens: Aitkens Δ^2-metod

Ett annat sätt att göra konvergens snabbare är extrapolation. Den ger inte nödvändigt högre konvergens ordning. Vi går tillbaka till beviset för konvergensen. Det följer att

$$\frac{p_k - p}{p_{k+1} - p} \approx \phi'(p)$$

och om $\phi'(p) \neq 0$ då $p_n - p$ bildar ungefärligt en geometrisk serie. Dvs $\{p_k\}$ konvergerar mot p liknar

$$p_{k+1} - p = h(p_k - p)$$

med faktor h , $|h| < 1$. Då kan h och p bestämmas från p_k, p_{k+1}, p_{k+1} genom

$$p_{k+1} - p = h(p_k - p), \quad p_{k+2} - p = h(p_{k+1} - p).$$

Lös ut h och p får vi

$$h = \frac{p_{k+1} - p_{k+1}}{p_{k+1} - p_k}, \quad p = \frac{p_k p_{k+1} - p_{k+1}^2}{p_{k+2} - 2p_{k+1} + p_k^2}$$

Definiera $\Delta p_k = p_{k+1} - p_k$. Då $\Delta^2 p_k = \Delta p_{k+1} - \Delta p_k = p_{k+2} - 2p_{k+1} + p_k^2$, vilket ger

$$p = p_k - \frac{(\Delta p_k)^2}{\Delta^2 p_k}.$$

Aitkens Δ^2-metod fås med denna formel. Talföljden $\{p_k\}$ överförs till en ny talföljd $\{p'_k\}$ med Aitken extrapolation.

$$p'_k = p_k - \frac{(\Delta p_k)^2}{\Delta^2 p_k}.$$

Fördelen med den här omskrivningen är att $\{p'_k\}$ konvergerar mot p snabbare än $\{p_k\}$ gör om $\{p_k\}$ beter sig asymptotiskt som en geometrisk serie. Med detta menas att det finns ett h, med $|h| < 1$ sådant att $p_k \neq p$ då gäller att

$$p_{k+1} - p_k = (h + \delta_k)(p_k - p), \quad \lim_{k\to\infty} \delta_k = 0.$$

Nu ska vi studera konvergens av $\{p'_k\}$. Låt $e_k = p_k - p$. Enligt antagandet ovan gäller att $e_{k+1} = (h + \delta_k)e_k$. Då följder att

$$\begin{aligned}
&p_{k+2} - 2p_{k+1} + p_k^2 = e_{k+2} - 2e_{k+1} + e_k \\
=&(h + \delta_{k+1})e_{k+1} - 2(h + \delta_k)e_k + e_k = (h + \delta_{k+1})(h + \delta_k)e_k - 2(h + \delta_k)e_k + e_k \\
=&e_k((h + \delta_{k+1})(h + \delta_k) - 2(h + \delta_k) + 1) = e_k((h-1)^2 + \mu_k) \text{ där } \mu \to 0
\end{aligned}$$

och

$$p_{k+1} - p_k = e_{k+1} - e_k = e_k((h-1) + \delta_k)$$

Därför

$$p_{k+2} - 2p_{k+1} + p_k^2 \neq 0$$

för tillräckligt stort k eftersom $e_k \neq 0$, $h \neq 1$ och $\mu_k \to 0$. Detta visar att iterationen för p'_k är väldefinierad och

$$p'_k - p = e_k - e_k \frac{((h-1) + \delta_k)^2}{(h-1)^2 + \mu_k}$$

för tillräckligt stort k, vilket medför att

$$\lim_{k\to 0} \frac{p'_k - p}{p_k - p} = \lim_{k\to 0} \left(1 - \frac{((h-1) + \delta_k)^2}{(h-1)^2 + \mu_k}\right) = 1 - 1 = 0.$$

Med andra ord konvergerar $\{p'_k\}$ mot p snabbare än $\{p_k\}$ om $p_k \to p$ är linjärt. Annars konvergerar $\{p'_k\}$ generellt inte för att om $\{p_k\}$ konvergerar mer än linjärt så kommer nämnaren att gå mot noll snabbare än täljaren och då går gränsvärdet mot oändligheten.

Å andra sidan kan vi utnyttja p_k för att konstruera följande snabb iterationsmetod genom att sätta

$$p'_k := \phi(p_k), \quad p''_k := \phi(p'_k), \quad k = 0, 1, 2, ...$$

$$p_{k+1} = p_k - \frac{p'_k - p_k}{p''_k - 2p'_k + p_k}.$$

Den nya metoden är

$$p_{k+1} = \psi(p_k), \ \psi(x) := \frac{x(\phi(\phi(x)) - \phi(x)^2}{\phi(\phi(x)) - 2\phi(x) + x}.$$

Vi skall bevisa att *p är en fixpunkt till ϕ är ekvivalent till att p är en fixpunkt till ψ.*

Självklart att p är en fixpunkt för ϕ om den är en fixpunkt för ψ eftersom, per definition $\psi(p)(\phi(\phi(p))-2\phi(p)+p) = p(\phi(\phi(p))-\phi(p)^2 \Leftrightarrow (\psi(p)-p)(\phi(\phi(p))-2\phi(p)+p) = (\phi(p)-p)^2$. Omvänt om $p = \phi(p)$ ser vi att $\psi(p)$ är obestånd form av typ $\frac{0}{0}$. För att komma runt detta problem beräknar vi gränsvärdet $\psi(x)$ då $x \to p$ under villkoren $\phi(x)$ är differentierbar i $x = p$ och $\phi'(p) \neq 0$. Med hjälp av L'Hopitals regel

$$\psi(p) = \lim_{x\to p}\psi(x) = \lim_{x\to p}\frac{\phi(\phi(x)) + x\phi'(\phi(x))\phi'(x) - 2\phi(x)\phi'(x)}{\phi'(\phi(x))\phi'(x) - 2\phi'(x) + 1} = \frac{p + p\phi'(p)^2 - 2p\phi'(p)}{1 + \phi'(p)^2 - 2\phi'(p)} = p.$$

Vår nästa uppgift är att undersöka konvergensegenskap. Vi nöjer oss med fallet där ϕ är två gånger kontinuerligt deriverbar i en omgivning av p och $\phi'(p) \neq 1$. Vi skall bevisa att ψ-iteration är kvadratisk genom att bevisa att $\psi'(p) = 0$. En direkt beräkning ger

$$\psi(x) = \frac{x(\phi(\phi(x)) - \phi(x)^2}{\phi(\phi(x)) - 2\phi(x) + x} = x - \frac{(\phi(x) - x)^2}{\phi(\phi(x)) - 2\phi(x) + x}.$$

Inför $F(x) := \phi(x) - x$, dvs $F(x) + x = \phi(x)$, och $F(p) = 0$, då

$\phi(\phi(x))-\phi(x) = F(\phi(x))+\phi(x)-\phi(x) = F(F(x)+x) \Longrightarrow \phi(\phi(x))-2\phi(x)+x = F(x+F(x))-F(x)$

Således

$$\psi(x) = x - \frac{F(x)^2}{F(x + F(x)) - F(x)}.$$

Taylorutveckla $F(x + F(x))$ i x:

$$F(x + F(x)) = F(x) + F'(x)F(x) + \frac{1}{2}F''(\xi)F(x)^2, \text{ för något } \xi \text{ mellan } x \text{ och } x + F(x).$$

$$\Longrightarrow \quad \frac{F(x + F(x)) - F(x)}{F(x)} = F'(x) + \frac{1}{2}F''(\xi)F(x)$$

$$\Longrightarrow \quad \psi(x) = x - \frac{F(x)}{F'(x) + \frac{1}{2}F''(\xi)F(x)}$$

$$\Longrightarrow \quad \frac{\psi(x) - \psi(p)}{x - p} = \frac{x - \frac{F(x)}{F'(x)+\frac{1}{2}F''(\xi)F(x)} - p}{x - p}$$

$$= 1 - \frac{F(x) - F(p)}{x - p} \cdot \frac{1}{F'(x) + \frac{1}{2}F''(\xi)F(x)} \to 1 - F'(p) \cdot \frac{1}{F'(p)} = 0$$

då $x \to p$. I beräkningarna ovan har vi använt $F(p) = 0$, $F'(p) = \phi'(p) - 1 \neq 0$. Detta medför att ψ-iterationen är kvadratisk.

Vi avslutar detta avsnitt med följande kommentar: *ψ-iterationen är kvadratisk konvergent även om $|\phi'(p)| > 1$*, vilket betyder att ϕ-iterationen divergerar.

6. Att hitta rötter till polynomekvationer med Newton-Raphsons metod

Betrakta polynomet $\pi(x) = a_0x^n + a_1x^{n-1} + \cdots + a_n$. Då är Newton-Raphsons iteration för att söka en rot p:

$$p_{k+1} = p_k - \frac{\pi(p_k)}{\pi'(p_k)}$$

Med andra ord behöver vi beräkna värdet av polynomet π och dess derivata π' i p_k. De kan fås på följande sätt: För $x = \xi$:

$$\pi(\xi) = (\cdots((a_0\xi + a_1)\xi + a_2)\xi + \cdots)\xi + a_n.$$

Det kan beskrivas rekursivt på formen

$$\begin{aligned} b_0 &:= a_0 \\ b_i &:= b_{i-1}\xi + a_i, \quad i = 1, 2, ..., n \end{aligned}$$

Från detta

$$\pi(\xi) = b_n.$$

Detta är *Horners regel*. Om polynomet $\pi(x)$ delas med $x - \xi$ har vi efter polynomdivision

$$\pi(x) = (x - \xi)\pi_1(x) + b_n$$

där

$$\pi_1(x) := b_0 x^{n-1} + b_1 x^{n-2} + \cdots + b_{n-1}.$$

(Vi kan också visa ovanstående påstående genom att jämföra koefficienterna i denna ekvation.) Följaktligen

$$\pi'(x) = \pi_1(x) + (x-\xi)\pi_1'(x) \quad \Longrightarrow \quad \pi'(\xi) = \pi_1(\xi) = (\cdots((b_0\xi + b_1)\xi + a_2)\xi + \cdots)\xi + b_{n-1}.$$

På samma sätt som för beräkning av $\pi(\xi) = b_n$ kan vi bestämma $\pi'(\xi) = c_{n-1}$ där

$$\begin{aligned} c_0 &:= b_0 \\ c_i &:= c_{i-1}\xi + b_i, \quad i = 1, 2, ..., n-1 \end{aligned}$$

och närmevärde till roten p fås av

$$p_{k+1} = p_k - \frac{b_n(p_k)}{c_{n-1}(p_k)}.$$

7. Sturm-polynomföljder

Vi har studerat konvergens av iterationsmetoder och tillämpning i lösandet av polynomekvationer. För att kunna få en konvergent talföljd behöver vi ett startvärde som krävs nära den sökta roten. Men i allmänhet vet vi inte var roten ligger exakt. Detta avsnitt kan ses som en del försök att lokalisera reella rötter med hjälp av polynomkoefficenter.

Låt $q(x)$ vara ett polynom av grad n,

$$q(x) = a_0 x^n + a_1 x^{n-1} + \cdots + a_n, \quad a_0 \neq 0.$$

Det är möjligt att bestämma antalet reella rötter av $q(x)$ i ett specificerat område genom att undersöka antalet teckenförändringar för vissa punkter $x = a$ i en följd av polynom $q_i(x)$, $i = 0, 1, ..., m$, av avtagande grader, t ex [2].

För att bättre formulera problemet inför vi följande definition.

Definition. (*Sturms polynomföljd*) Polynomföljden

$$q(x) = q_0(x), q_1(x), ..., q_m(x)$$

av reella polynom är en *Sturm polynomföljd för polynomet* $q(x)$ om:

(a) Alla reella rötter av $q_0(x)$ är enkla.
(b) $\operatorname{sign} q_1(\xi) = -\operatorname{sign} q_0'(\xi)$ om ξ är en reell rot av $q_0(x)$, där sign står för tecken.
(c) För $i = 1, 2, ..., m-1$,

$$q_{i+1}(\xi)q_{i-1}(\xi) < 0$$

om ξ är en reell rot av $q_i(x)$.
(d) Det sista polynomet $q_m(x)$ har inga reella rötter.

Exempel. Låt

$$q_0(x) := 3x^5 + 8x^4 - 3x^3 - 18x^2 - 6x + 4$$
$$q_1(x) := -15x^4 - 32x^3 + 9x^2 + 36x + 6$$
$$q_2(x) := -116/25 + 24x/25 + 246x^2/25 + 346x^3/75$$
$$q_3(x) := -179400/29929 - 625950x/29929 - 363150x^2/29929$$
$$q_4(x) := 816582836/146531025 + 74463352x/16281225$$
$$q_5(x) := -25617603025/17368576854$$

Lite räkningar med hjälp av `Mathematica` visar att $q_0(x)$ har fem enkla reella rötter $\xi_1 = -2$, $\xi_2 = -1$,$\xi_3 = 1/3$, $\xi_4 = -\sqrt{2}$, $\xi_5 = \sqrt{2}$, och

$q_0(x) = (-8/75 - x/5)q_1(x) - p_2(x)$
$q_1(x) = (-75/59858 - 1125x/346)q_2(x) - q3(x)$
$q_2(x) = (-342028612/2197965375 - 5177717x/13618125)q_3(x) - q_4(x)$
$q_3(x) = (-928954191549375/693098848884488 - 2956263429375x/1114306831004)q_4(x) - q_5(x)$

Därav uppfyller villkor (b) i definitionen. Självklart gäller att $\operatorname{sign} q_1(\xi_i) = -\operatorname{sign} q_0'(\xi_i)$, för $i = 1, ..., 5$ eftersom $q_1(x) = -q_0'(x)$ och det finns inga rötter hos det sista polynomet q_5. Så polynomföljden är en Sturmföljd.

Först visar vi att vi kan konstruera en Stums följd för ett polynom med alla enkla reella rötter $\pi(x)$ på följande sätt.

Exempel. (Euklides algoritm) Definiera

$$\pi_0(x) = \pi(x),\ \pi_1(x) = -\pi_0'(x) = -\pi'(x).$$

Bilda nu rester $\pi_{i+1}(x)$ rekursivt genom att dela $\pi_{i-1}(x)$ med $\pi_i(x)$:

$$\pi_{i-1}(x) = q_i(x)\pi_i(x) - c_i\pi_{i+1}(x),\ i = 1, 2, ...$$

där graden av $\pi_i(x)$ är större än graden av $\pi_{i+1}(x)$ och konstanterna $c_i > 0$ men för övrigt godtyckliga. Vi inser omedelbart att denna procedur är Euklides algoritm. På grund av polynomens grader avtar tar divisionen slut efter $m \leq n$ steg:

$$\pi_{m-1}(x) = q_m(x)\pi_m(x),\ \pi_m(x) \not\equiv 0.$$

Det sista polynomet $\pi_m(x)$ är den största gemensamma delaren till de initiala två polynomen $\pi(x)$ och $\pi_1(x) = -\pi'(x)$. Om alla reella rötter till $\pi(x)$ är enkla så har $\pi(x)$ och $\pi'(x)$ inte gemensamma rötter. Alltså har $\pi_m(x)$ inga reella rötter och då är villkor (d) uppfyllt. Om $\pi_(\xi) = 0$ ger algoritmen $\pi_{i-1}(\xi) = -c_i\pi_{i+1}(\xi)$. Antag motsatsen att ξ är en rot till $\pi_{i+1}(x) = 0$, dvs, $\pi_{i+1}(\xi) = 0$. Då medför Euklides algoritm $\pi_{i+1}(\xi) = \cdots = \pi_m(\xi) = 0$, en motsägelse till $\pi_m(\xi) \neq 0$. Alltså är villkor (c) uppfyllt. De övriga två villkoren är triviala.

Sturms följder dyker upp naturligt i många sammanhang, t ex i moment problem [1] eller i tillämpningar (se t ex exempel i avsnitt 9.1). Vi betraktar följande Jacobimatris (en symmetrisk tridiagonalmatris)

$$\begin{pmatrix} \alpha_1 & \beta_2 & & & \\ \beta_2 & \alpha_2 & \cdot & 0 & \\ & \cdot & \cdot & \cdot & \\ & 0 & \cdot & \cdot & \beta_n \\ & & & \beta_n & \alpha_n \end{pmatrix}$$

Beteckna determinanten till delmatriserna $J_i - xI$, $\pi_i(x)$, där $J_i = \begin{pmatrix} \alpha_1 & \beta_2 & & & \\ \beta_2 & \alpha_2 & \cdot & 0 & \\ & \cdot & \cdot & \cdot & \\ & 0 & \cdot & \cdot & \beta_i \\ & & & \beta_i & \alpha_i \end{pmatrix}$,

$i = 1, 2, ..., n$. Det är självklart att

$$\pi_1(x) = \alpha_1 - x.$$

$$\pi_2(x) = \begin{vmatrix} \alpha_1 - x & \beta_2 \\ \beta_2 & \alpha_2 - x \end{vmatrix} = (\alpha_2 - x)(\alpha_1 - x) - \beta_2^2$$

Om vi sätter $\pi_0(x) := 1$. Då erhålls

$$\pi_2(x) = (\alpha_2 - x)\pi_1(x) - \beta_2^2 \pi_0(x)$$

Nästa polynom blir

$$\begin{aligned} \pi_3(x) &= \begin{vmatrix} \alpha_1 - x & \beta_2 & 0 \\ \beta_2 & \alpha_2 - x & \beta_3 \\ 0 & \beta_3 & \alpha_3 - x \end{vmatrix} = (\alpha_3 - x) \begin{vmatrix} \alpha_1 - x & \beta_2 \\ \beta_2 & \alpha_2 - x \end{vmatrix} - \beta_3 \begin{vmatrix} \alpha_1 - x & 0 \\ \beta_2 & \beta_3 \end{vmatrix} \\ &= (\alpha_3 - x)\pi_2(x) - \beta_2^2 \pi_1(x) \end{aligned}$$

Gissningsvis borde vi få

$$\pi_i(x) = (\alpha_i - x)\pi_{i-1}(x) - \beta_i^2 \pi_{i-2}(x), \quad i = 2, ..., n$$

Detta kan bevisas med hjälp av induktion. Vi har redan bevisat att likheten gäller för $i = 2$ och 3. Antag nu den gäller för $i = k$. Då beräknar vi determinanten genom att utveckla den längst sista raden

$$\begin{vmatrix} \alpha_1 - x & \beta_2 & & & & \\ \beta_2 & \alpha_2 - x & \cdot & 0 & & \\ & \cdot & \cdot & \cdot & & \\ & 0 & \cdot & \cdot & \beta_{k+1} & \\ & & & \beta_{k+1} & \alpha_{k+1} - x & \end{vmatrix}$$

$$= (\alpha_{k+1} - x) \begin{vmatrix} \alpha_1 - x & \beta_2 & & & & \\ \beta_2 & \alpha_2 - x & \cdot & 0 & & \\ & \cdot & \cdot & \cdot & & \\ & 0 & \cdot & \cdot & \beta_k & \\ & & & \beta_k & \alpha_k - x & \end{vmatrix} + (-1)^{k+1+k} \beta_{k+1} \begin{vmatrix} \alpha_1 - x & \beta_2 & & & & \\ \beta_2 & \alpha_2 - x & \cdot & 0 & & \\ & \cdot & \cdot & \cdot & & \\ & 0 & \cdot & \alpha_{k-1} - x & 0 & \\ & & & \beta_k & \beta_{k+1} & \end{vmatrix}$$

$$= (\alpha_{k+1} - x)\pi_k(x) - \beta_{k+1}^2 \begin{vmatrix} \alpha_1 - x & \beta_2 & & & \\ \beta_2 & \alpha_2 - x & \cdot & 0 & \\ & \cdot & \cdot & \cdot & \\ & 0 & \cdot & \beta_{k-1} & \\ & & \beta_{k-1} & \alpha_{k-1} - x & \end{vmatrix}$$

$$= (\alpha_{k+1} - x)\pi_k(x) - \beta_{k+1}^2 \pi_{k-1}(x).$$

Därav har vi bevisat:

Proposition. *Polynomföljden $\pi_i(x)$ är definierade av följande tre-termer rekursion:*

$$\begin{aligned} \pi_0(x) &= 1 \\ \pi_1(x) &= \alpha_1 - x \\ \pi_i(x) &= (\alpha_i - x)\pi_{i-1}(x) - \beta_i^2 \pi_{i-2}(x), \quad i = 2, ..., n. \end{aligned}$$

Vidare har vi rekursionen för derivatorna av polynomen:

$$\begin{aligned}\pi_0'(x) &= 0\\ \pi_1'(x) &= -1\\ \pi_i'(x) &= -\pi_{i-1}(x) + (\alpha_i - x)\pi_{i-1}'(x) - \beta_i^2\pi_{i-2}'(x), \quad i = 2, ..., n.\end{aligned}$$

Vårt nästa mål är att bevisa följande egenskap hos $\pi_i(x)$:

Sats. *Polynomföljden $\pi_i(x)$ är en Sturms följd för alla reella tal α_j, β_j och $\beta_j \neq 0$, $j = 2, ..., n$.*

Bevis. Det är uppenbart att det sista polynomet $\pi_0(x) = 1$ inte har några reella rötter, dvs, villkor (d) i definitionen är uppfyllt. Det är även ganska lätt att inse om ξ är en reell rot till $\pi_i(x)$ så gäller att, för $i = 1, 2, ..., n$

$$\pi_{i+1}(\xi)\pi_{i-1}(\xi) < 0$$

om $\beta_i \neq 0$ eftersom rekursionen ovan ger

$$\pi_{i+1}(\xi) = (\alpha_{i+1} - \xi)\pi_i(\xi) - \beta_{i+1}^2\pi_{i-1}(\xi) = -\beta_{i+1}^2\pi_{i-1}(\xi).$$

Det visar att villkor (c) i definitionen för Sturms följd är uppfyllt.

Nu skall vi bevisa med induktion att alla rötter $x_k^{(i)}$, $k = 1, ..., i$ till $\pi_i(x)$, $i = 1, ..., n$ är reella och enkla:

$$x_1^{(i)} > x_2^{(i)} > \cdots > x_i^{(i)}$$

och rötterna till $\pi_{i-1}(x)$ respektive $\pi_i(x)$ separerar varandra strikt:

$$x_1^{(i)} > x_1^{(i-1)} > x_2^{(i)} > x_2^{(i-1)} > \cdots > x_{i-1}^{(i-1)} > x_i^{(i)}.$$

Påståendet är trivialt för $i = 1$. Antag nu att det är sant för $i \geq 1$, dvs att rötterna $x_k^{(i)}$ och $x_k^{(i-1)}$ till $\pi_i(x)$ respektive $\pi_{i-1}(x)$ uppfyller

$$x_1^{(i)} > x_1^{(i-1)} > x_2^{(i)} > x_2^{(i-1)} > \cdots > x_{i-1}^{(i-1)} > x_i^{(i)}.$$

Av ovanstående Propositionen $\pi_k(x)$ är på formen $\pi_k(x) = (-1)^k x^k + \cdots$. Graden av $\pi_k(x)$ är lika med k. Då vet vi att $\pi_{i-1}(x)$ inte ändrar tecken för $x > x_1^{(i-1)}$. Vidare har vi antagit att alla rötter $x_k^{(i-1)}$ är enkla. Så föregående olikheterna medför

$$\operatorname{sign}\pi_{i-1}(x_k^{(i)}) = (-1)^{i+k}, \ k = 1, 2, ..., k.$$

Alltså erhålles enligt rekursionen i Propositionen

$$\pi_{i+1}(x_k^{(i)}) = -\beta_i^2\pi_{i-1}(x_k^{(i)}), \ k = 1, 2, ..., k.$$

Eftersom $\beta_i^2 > 0$ får vi

$$\operatorname{sign}\pi_{i+1}(x_k^{(i)}) = (-1)^{i+k+1}, \ k = 1, 2, ..., i,$$

$$\operatorname{sign}\pi_{i+1}(+\infty) = (-1)^{i+1}, \ \operatorname{sign}\pi_{i+1}(-\infty) = 1,$$

och $\pi_{i+1}(x)$ ändrar tecken i alla intervallen

$$[x_1^{(i)}, \infty), \ (-\infty, x_i^{(i)}], \ [x_{k+1}^{(i)}, x_k^{(i)}], \ k = 1, ..., n-1.$$

Således är rötterna $x_k^{(i+1)}$ till polynomet $\pi_{i+1}(x)$ reella och enkla och de separerar rötterna $x_k^{(i)}$ till $\pi_i(x)$:

$$x_1^{(i+1)} > x_1^{(i)} > x_2^{(i+1)} > x_2^{(i)} > \cdots > x_i^{(i)} > x_{i+1}^{(i+1)}.$$

Därav har vi bevisat att villkor (a) är uppfyllt.

Till sist skall vi kontrollera om villkor (b) är uppfyllt. Vi vet nu att $\pi_n(x)$ har enkla reella rötter $\xi_1 > \xi_2 > \cdots > \xi_n$. Från tecken beräkningar ovan och rekursionen för derivatorna inser vi att

$$\operatorname{sign} \pi_{n-1}(\xi_k) = (-1)^{n+k}, \quad \operatorname{sign} \pi_n'(\xi_k) = (-1)^{n+k+1} = -\operatorname{sign} \pi_{n-1}(\xi_k)$$

för $k = 1, ..., n$. Så är beviset klart. □

Slutligen skall vi visa följande sats som kan hjälpa oss att hitta en god initial gissning för iterationsmetoder.

Sats. *Antalet reella rötter till polynom $\pi(x) = \pi_0(x)$ i intervallet $a \leq x < b$ är lika med $w(b) - w(a)$, där $w(x)$ är antalet teckenändringar i Sturms följd $\pi_0(x), \pi_1(x), ..., \pi_m(x)$ i x.*

Bevis. Beviset bygger på undersökning av hur störningar av värdet a påverkar antalet teckenändringar $w(a)$ i följden

$$\pi_0(a), \pi_1(a), ..., \pi_m(a)$$

Så länge a inte är en rot av någon av polynomen $\pi_i(x)$,$i = 0, 1, ..., m$, är det självklart ingen förändring. Om a är en rot av $\pi_i(x)$, studerar vi två fall: $i > 0$ och $i = 0$.

I det första fallet, $i < m$ enligt villkor (d) i definitionen för Sturms följd, och $\pi_{i+1}(a) \neq 0$, $\pi_{i-1}(a) \neq 0$ enligt villkor (c)). Om $\pi_i(x)$ ändrar tecken vid $x = a$, då för en tillräckligt liten störning $h > 0$, kan vi ställa upp ett av följande tecken tabeller för tecknen till polynomen $\pi_j(a)$, $j = i - 1, i, i + 1$:

	$a-h$	a	$a+h$
$i-1$	$-$	$-$	$-$
i	$-$	0	$+$
$i+1$	$+$	$+$	$+$

	$a-h$	a	$a+h$
$i-1$	$+$	$+$	$+$
i	$-$	0	$+$
$i+1$	$-$	$-$	$-$

	$a-h$	a	$a+h$
$i-1$	$-$	$-$	$-$
i	$+$	0	$-$
$i+1$	$+$	$+$	$+$

	$a-h$	a	$a+h$
$i-1$	$+$	$+$	$+$
i	$+$	0	$-$
$i+1$	$-$	$-$	$-$

I varje situation, $w(a-h) = w(a) = w(a+h)$: antalet teckenförändringar förblir densamma. Detta är också sant om $\pi_i(x)$ inte ändrar tecken vid $x = a$.

I det andra fallet, drar vi slutsatsen från villkor (b) att följande teckenmönster gäller:

i	$a-h$	a	$a+h$
0	$-$	0	$+$
1	$-$	$-$	$-$

i	$a-h$	a	$a+h$
0	$+$	0	$-$
1	$+$	$+$	$+$

Vi ser nu att i varje situation $w(a - h) = w(a) = w(a + h) - 1$: exakt en teckenändring när vi passerar genom en rot av $\pi_0(x)?\pi(x)$.

För $a < b$ och tillräckligt liten $h > 0$,

$$w(b) - w(a) = w(b - h) - w(a - h)$$

pekar på antalet rötter av $\pi(x)$ i intervallet $a - h < x < b - h$. Då kan $h > 0$ väljas godtycklig liten, ovanstående skillnad anger antalet rötter även i intervallet $a \leq x < b$. □

Vi avslutar detta avsnitt med en algoritm, intervallhalvering, för att beräkna i :te roten till $\pi_n(x)$ genererade av den tridiagonala matrisen $(\xi_1 > \xi_2 > \cdots > \xi_n)$.

För $x = -\infty$, har Sturms följden i Propositionen teckenmönstret

$$+, +, ..., +$$

Således $w(-\infty) = 0$. Av den föregående satsen $w(\mu)$ anger antalet rötter ξ av $\pi_n(x)$ med $\xi < \mu$: $w(\mu) \geq n + 1$ om och endast om $\xi_i < \mu$.

8. Uppskattningar av rötter till polynomekvationer

Låt $\pi(x) = a_0x^n + a_1x^{n-1} + \cdots a_{n-1}x + a_n$ där $a_0 \neq 0$. Det finns många sätt att uppskatta i termer av koefficienterna $a_0, ..., a_n$. Vi studerar två av dem.

Sats. *För alla rötter ξ_i till ett godtyckligt polynom $\pi(x) = a_0x^n + a_1x^{n-1} + \cdots a_{n-1}x + a_n$ med $a_0 \neq 0$ gäller att*

$$|\xi_i| \leq \max\left\{\left|\frac{a_n}{a_0}\right|, 1 + \left|\frac{a_{n-1}}{a_0}\right|, ..., 1 + \left|\frac{a_1}{a_0}\right|\right\}$$

$$|\xi_i| \leq \max\left\{1, \sum_{j=1}^{n}\left|\frac{a_j}{a_0}\right|\right\}$$

Bevis. Vi överför problemet till att uppskatta egenvärden till matris på formen (kallas Frobenius companion matrix)

$$C = \begin{pmatrix} 0 & 0 & \cdots & 0 & -c_0 \\ 1 & 0 & \cdots & 0 & -c_1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & -c_{n-1} \end{pmatrix}$$

Med hjälp av induktion kan vi visa att det karakteristiska polynomet till C är

$$\chi(x) = x^n + c_{n-1}x^{n-1} + \cdots + c_1x + c_0$$

Och detsamma för C^T.

Gershgorins cirkelsats [3] säger att alla egenvärden till denna matris ligger i union av följande diskskivor:

$$|z| \leq |c_0|$$
$$|z| \leq 1 + |c_j|,\ j = 1, ..., n-2$$
$$|z + c_{n-1}| \leq 1$$

Den sista diskskivan ligger innanför

$$|z| \leq 1 + |c_{n-1}|$$

Alltså ligger alla egenvärden i området

$$|z| \leq \max\{|c_0|, 1 + |c_1|, ..., 1 + |c_{n-1}|\}$$

Eftersom rötter till polynomet $\pi(x)$ är samma som rötterna till $x^n + \frac{a_1}{a_0}x + \cdots + \frac{a_n}{a_0} = 0$. Vi får den första uppskattningen med observationen att $c_j = a_j/a_0$, $j = 1, ..., n$.

Observera att $\chi(x)$ är även det karakteristiska polynomet till C^T. Tillämpning av Gershgorins cirkelsats igen, får vi union av följande områden för egenvärdena

$$|z| \leq 1, \quad |z + c_{n-1}| \leq |c_0| + |c_1| + \cdots + |c_{n-2}|$$

Den sista diskskivan är delmängd av

$$|z| \leq \sum_{j=0}^{n-1} |c_j|.$$

Tillsammans har vi att egenvärdena ligger i

$$|z| \leq \max\{1, \sum_{j=0}^{n-1} |c_j|\}$$

vilket visar den andra uppskattningen. □

Exempel. Låt $\pi(x) = x^3 - 2x^3 + x - 1$. Rötterna ligger i

$$|z| \leq \max\{1, 2, 3\} = 3$$

eller

$$|z| \leq \max\{1, 1 + 1 + 2\} = 4$$

Då den första uppskattningen ger en tightare gräns.

9. Iterationsmetoder för att lösa linjära ekvationssystem

När vi skall lösa stora linjära ekvationssystem $Ax = b$ så är direkta metoder så som Gauss elimination, LU–faktorisering inte lämpliga. Låt oss försöka lösa ett ekvationssystem med matrisen A av 20000×20000 med 1 på diagonalen och likformigt slumpade tal mellan 0 och 10^{-4} för de övriga element. Vi försöker följande kod i Matlab

```
A=0.0005*rand(20000);
b=rand(20000,1);
for i=1:20000; A(i,i)=1; end;
x=A\b
```

Då kan Matlab inte genomföra beräkningen. Det är för stort! Det största problemet är kanske skalning vid stort konditionstal. Att bilda och faktorisera stora matriser är också mycket kostsamt. Då använder vi iterationsmetoder. Om vi kan få till en bra formulering är de rätt effektiva.

9.1. **Ett modellproblem.** Betrakta differentialekvationen

$$-u''(x) = f(x) \quad \text{för } 0 \leq x \leq 1$$

med randvillkor $u(0) = u(1) = 0$. Vi approximerar $u''(x)$ med differens

$$u''(x) = \frac{u(x+h) - 2u(x) + u(x-h)}{h^2} + O(h^2)$$

där $h = 1/(N+1)$. Då delas $[0, 1]$ i N små intervall. Låt $u_i = u(ih)$, $g_i = h^2 f(ih)$. Vi har ett ekvationssystem

$$-u_{i-1} + 2u_i - u_{i+1} = g_i, \quad i = 1, ..., N$$

$$u_0 = u_{N+1} = 0$$

På matrisformen

$$Tu = -g$$

där

$$T = \begin{pmatrix} 2 & -1 & 0 & \cdots & 0 & 0 \\ -1 & 2 & -1 & \cdots & 0 & 0 \\ 0 & -1 & 2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 2 & -1 \\ 0 & 0 & 0 & \cdots & -1 & 2 \end{pmatrix}$$

Om vi vill lösa ekvationen iterativt söker vi inte den exakta lösningen. Vi söker en lösning $u^{(k+1)}$ som är mer korrekt än $u^{(k)}$: Men vi måste ha en maskineri för att generera den t ex

$$-u_{i-1}^{(k)} + 2u_i^{(k+1)} - u_{i+1}^{(k)} = g_i, \quad i = 1, ..., N$$

Denna iteration är *Jacobis iteration*. Vi kan naturligtvis ta en annan möjligt gammal värde

$$-u_{i-1}^{(k+1)} + 2u_i^{(k+1)} - u_{i+1}^{(k)} = g_i, \quad i = 1, ..., N$$

Denna iteration är *Gauss-Seidels iteration*.

Nu har vi iterationer. Men hur analyserar vi konvergens och feluppskattning? Det är rätt trassligt att angripa problem som ser så här ut.

Ett vanligt verktyg är att skriva dem på matrisform!

9.2. **Splittringsmatris.** Som tidigare nämnt för att hitta lösning för $x = f(x)$ så att skapa en iteration $x^{(k+1)} = \phi(x^{(k)})$ försöker vi göra liknande för $Ax = b$.

- Dela matrisen A i två delar: $A = M - N$ där M ska vara så lik A som möjligt men det ska vara lättare lösa ekvationssystem med M som koefficientmatris (som kanske inte är så med A). Mer precis får vi ett ekvationssystem $Mx = Nx + b = Mx + (b - Ax)$
- En "naturlig"Iteration blir:

$$Mx^{(k+1)} = Nx^{(k)}x + b \text{ eller ekvivalent } x^{(k+1)} = x^{(k)} - M^{-1}(Ax^{(k)} - b)$$

Fixpunkten x^* till iterationen är lösningen till $Ax^* = b$. Det är nu ganska lätt att uppskatta felet. Låt $e^{(k)} = x^{(k)} - x^*$.

$$e^{(k+1)} = e^{(k)} - M^{-1}Ae^{(k)} = (I - M^{-1}A)e^{(k)}$$

$$\implies \|e^{(k+1)}\| \leq \|(I - M^{-1}A)e^{(k)}\| \leq \|I - M^{-1}A\|\|e^{(k)}\|$$

Om $\|I - M^{-1}A\| < 1$ konvergerar iterationen. Men den omvända är inte sant. Ofta kan vi uppskatta denna norm med spektral radie av matrisen $I - M^{-1}A$.

Två speciella uppdelningar.

- Jacobis iteration: M är diagonalelement av A.
- Gauss-Seidels iteration: M är triangulär matris med undretriangulära delen av A.

Exempel. Vi ska lösa ekvationssystemet $Ax = b$ med iterationsmetod, där

$$A = \begin{pmatrix} 2 & 10 & 0 & -1 \\ 0 & -1 & 1 & 5 \\ 5 & 1 & 0 & 0 \\ -1 & 0 & 10 & 0 \end{pmatrix} \quad b = \begin{pmatrix} 30 \\ 25 \\ 10 \\ 70 \end{pmatrix}$$

Idén är att hitta ett lämpligt sätt att dela matrisen till två delar. Vi inser att varje rad har ett dominerande tal och de dessutom ligger inte i samma kolonn. Då kan vi byta raderna så att vi löser ekvationen på formen

$$\begin{pmatrix} 5 & 1 & 0 & 0 \\ 2 & 10 & 0 & -1 \\ -1 & 0 & 10 & 0 \\ 0 & -1 & 1 & 5 \end{pmatrix} x = \begin{pmatrix} 10 \\ 30 \\ 70 \\ 25 \end{pmatrix} \Leftrightarrow \left[\begin{pmatrix} 5 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 5 \end{pmatrix} + \begin{pmatrix} 0 & 1 & 0 & 0 \\ 2 & 0 & 0 & -1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 \end{pmatrix} \right] x = \begin{pmatrix} 10 \\ 30 \\ 70 \\ 25 \end{pmatrix}$$

$$\Longleftrightarrow x = \begin{pmatrix} 2 \\ 3 \\ 7 \\ 5 \end{pmatrix} + \begin{pmatrix} 0 & -0.2 & 0 & 0 \\ -0.2 & 0 & 0 & 0.1 \\ 0.1 & 0 & 0 & 0 \\ 0 & 0.2 & -0.2 & 0 \end{pmatrix} x$$

Gauss-Seidels metod med startvektor $x^{(0)} = (2, 3, 7, 5)^T$ ger:

$$x^{(1)} = \begin{pmatrix} 2 - 0.6 \\ 2 - 0.2 \cdot 1.4 + 0.1 \cdot 5 \\ 7 + 0.1 \cdot 1.4 \\ 5 + 0.2 \cdot 3.22 - 0.2 \cdot 7.14 \end{pmatrix} = \begin{pmatrix} 1.4 \\ 3.22 \\ 7.14 \\ 4.216 \end{pmatrix} \text{ och } x^{(1)} = \begin{pmatrix} 1.356 \\ 3.150 \\ 7.136 \\ 4.203 \end{pmatrix}$$

Exempel. (En iterationsmetod för beräkning av matrisinvers) Vi använder oss av analog för att beräkna heltalsdivison $x = 1/q$ där q är ett positivt heltal. Det är samma som att lösa ekvationen $f(x) := 1/x - q = 0$. Genom att tillämpa Newton-Raphsons metod får vi

$$p_{k+1} = p_k - p_k(qp_k - 1).$$

Nu antar vi att en kvadratisk matris A är inverterbar. Vi ska lösa matrisekvationen $AX = I$ med Newton-Raphsons metod. Då har vi iterationen

$$X_{k+1} = X_k + X_k(I - AX_k),$$

med en startmatris X_0 så att matrisföljden konvergerar mot A^{-1} kvadratiskt. För att gränsmatrisen ska vara inversen till A måste $AX_k = X_kA$ för alla $k > 0$ om $AX_0 = X_0A$.

Sätt $E_k = I - AX_k$. Då

$$\begin{aligned} E_{k+1} &= I - AX_{k+1} = I - A(X_k + X_k(I - AX_k)) = I - AX_k - AX_k(I - AX_k) \\ &= (I - AX_k)(I - AX_k) = (I - AX_k)^2 = E_k^2 = E_0^{2^k} \end{aligned}$$

Använd en lämplig matrisnorm (se t ex [3]) kan vi få en uppskattning

$$\|E_k\| \leq \|E_0\|^{2^k} \to 0 \text{ då } k \to \infty$$

om $\|I - AX_0\| = \|E_0\| < 1$, dvs $AX_k \to I$. Vi får dessutom

$$\frac{\|E_{k+1}\|}{\|E_k\|^2} < 1$$

Så konvergensen är kvadratisk.

Kvarstår att bevisa $AX_k = X_kA$ om $AX_0 = X_0A$. Detta kan göras med matematisk induktion. Vi har redan bassteget klart. Antag nu att $AX_k = X_kA$. Vi har

$$\begin{aligned} AX_{k+1} &= A(X_k + X_k(I - AX_k)) = AX_k + AX_k(I - AX_k) \\ &= X_kA + X_kA(I - X_kA) = X_kA + X_k(I - AX_k)A = (X_k + X_k(I - AX_k))A = X_{k+1}A \end{aligned}$$

Så har vi bevisat att $AX_k \to I$ medför att $X_k \to A^{-1}$.

Referenser

[1] N. I. Akhiezer,The classical moment problem and some related questions in analysis. New York: Hafner Publishing Co., 1965

[2] F.R. Gantmacher, Matrix theory Vol I & II, AMS Chelsea Publishing.

[3] A. Holst & V. Ufnarovski, Matrix theory, Studentlitteratur.

[4] A. S. Householder, The Numerical Treatment of a Single Nonlinear Equation, McGraw-Hill, New York, 1970.

[5] T. LoFaro, The Dynamics of Symmetric Bimodal Maps, International Journal of Bifurcation and Chaos, 1997.

[6] `http://library.wolfram.com/infocenter/Demos/387/`

[7] J. Stoer and R. Bulirsch, Introduction to numerical analysis, Springer-Verlag, 1980

[8] E. W.Weisstein, "Halley's method". MathWorld.

[9] Y. Zhou, Föreläsningsanteckningar i Numerisk analys, 2017.