Studieleidraad bij Numerieke methoden (1805)

1ste bachelor Wiskunde – 2de bachelor Fysica

Universiteit Hasselt

Inleiding

Gedurende 10 weken worden de basis begrippen en technieken in de numerieke wiskunde gepresenteerd. Het boek
A. Quarteroni, R. Sacco en F. Saleri, *Numerical Mathematics*, 2^{nd} Edition, Springer, 2007.
wordt als begeleidend materiaal gebruikt. Dit vak wordt vervolgd door twee andere numerieke vakken (jaar 2 en jaar 3) waarvoor hetzelfde boek wordt gebruikt. Het onderstaande boek kan als aanvullend materiaal gebruikt worden
Adhemar Bultheel, *Inleiding tot de numerieke wiskunde*, Acco, 2006, ISBN 9789033462535
Naast het uitwerken van theoretische opgaven zullen jullie gevraagd worden om praktische oefeningen te doen waarvoor MATLAB gebruikt kan worden. Daarom zijn jullie verzocht jullie notebooks mee te nemen voor de contactmomenten.
De evaluatie is gebaseerd op twee componenten: een schriftelijk examen na de afloop van de collegeweken en het huiswerk. Het eindcijfer wordt bepaald door het gemiddelde van de cijfers voor huiswerktaken (25%) en het schriftelijk examen (75%). Deze verdeling geldt alleen als het cijfer voor het schriftelijk examen groter of gelijk is aan 8. Als het cijfer voor het schriftelijk examen minder is dan 8 is dan telt het cijfer van het schriftelijk examen voor 100%. Deze regel geldt voor beide zittingen, het cijfer voor de huiswerktaken blijft geldig.

Les 1

Korte inhoud

Naast een korte motivatie worden er in deze les inleidende aspecten besproken: fouten, getallenvoorstelling, afrondfouten, conditionering van problemen, stabiliteit.

Zelfstudie-opdrachten

Foutenanalyse

1. Herhaal de definitie van *absolute* en *relatieve fout* (zie ook (2.22) op blz. 40). Het is belangrijk om het verschil tussen deze twee te begrijpen (supplementair: lees ook blz. 13–14 in het boek van A. Bultheel).

2. De afstand tussen twee gebouwen is $d = 1.36km$. Voor een nieuw bouwproject wordt deze afstand op de kaart gemeten die de schaal 1:10000 heeft. Het resultaat is $135mm$. Bepaal de absolute en relatieve fouten.

3. *Landau symbolen*: (kleine o vs. grote O)
   Gegeven twee functies $f, g : \mathbb{R} \to \mathbb{R}$ en $a \in \mathbb{R}$, $f(x)$ is kleine o van $g(x)$ in een

omgeving van a (dicht bij a) als geldt

$$\lim_{x\to a} \frac{f(x)}{g(x)} = 0.$$

Analoog, $f(x)$ is grote O van $g(x)$ in een omgeving van a als er $M \in (0, \infty)$ bestaat z.d.

$$\lim_{x\to a} \frac{|f(x)|}{|g(x)|} = M.$$

Notatie: $f(x) = o(g(x))$ resp. $f(x) = O(g(x))$.

De betekenis van deze symbolen kan ook voor de gevallen $a = \pm\infty$ verruimd worden.

4. Wat betekent f is $o(1)$ in de buurt van $a = 2$? Gelijkaardig, Wat betekent f is $O(x^p)$ $(p > 0)$ in de buurt van $a = 0$? En als $a = \infty$ met $p > 0$, resp. $p < 0$? Geef een voorbeeld voor elk van deze gevallen.

5. Lees vervolgens paragraaf 2.4 om een idee te krijgen over mogelijke foutbronnen (informatief).

6. Lees paragraaf 2.5. Bestudeer eerst paragraaf 2.5.1 over de klassieke voorstelling van getallen in een *talstelsel* met *grondtal/basis* β en leer het begrip *beduidende cijfers*. Bestudeer daarna paragraaf 2.5.2 over *drijvende/bewegende kommavoorstelling; floating point representation* en leer de begrippen *basis, mantisse, exponent*. Waarom is een *normalisatie* nodig bij drijvende kommavoorstellingen?

7. Druk de volgende getallen uit (gegeven in verschillende grondtallen) als een getal in grondtal 10:

$$(102)_3, -(23)_7, (101.22)_5.$$

8. Stel de afstand $d = 1360m$ (zie ook punt 2) voor als in de genormaliseerde drijvende komma vorm met grondtal $\beta = 10$ en geef duidelijk de elementen in deze voorstelling op. Stel vervolgens d voor in de drijvende komma vorm met basis $\beta = 8$.

9. We bespreken nu het verband tussen fouten en juiste cijfers in de voorstelling van getallen in het talstelsel met grondtal $\beta \in \mathbb{N}$. Merk op, $\beta > 1$!
Juiste cijfer: Zij $x \in \mathbb{R}$ met de exacte voorstelling ($0 \le n < \infty$ en $0 \le m \le \infty$))

$$x = \sum_{k=-m}^{n} c_k \beta^k.$$

Voor de eenvoud beschouwen we hier het geval $x > 0$. Zij verder $\bar{x} \in \mathbb{R}$,

$$\bar{x} = \sum_{k=-\bar{m}}^{\bar{n}} \bar{c}_k \beta^k$$

een benadering van x met $0 \le \bar{n} < \infty$ en $0 \le \bar{m} \le \infty$. De cijfer $\bar{c}_k$ is juist als geldt $|x - \bar{x}| \le \frac{1}{2}\beta^k$.

10. Toon aan dat als het cijfer $\bar{c}_k$ juist is dan zijn ook alle cijfers $\bar{c}_i$ met $i \geq k$ juist. Als $\bar{c}_k$ het laatste juiste cijfer is (v.l.n.r, of het kleinste index k) dan geldt $|x - \bar{x}| > \frac{1}{2}\beta^{k-1}$.

11. *Absolute fout vs. juiste cijfers*
    Stel dat voor de benadering $\bar{x}$ van x geldt dat alle cijfers voor de komma juist zijn en zij k het laatste juiste cijfer ($k < 0$). Uit de bovenstaande opmerkingen volgt dat het aantal juiste cijfers na de komma bepaald kan worden als het natuurlijk getal p waarvoor geldt:
    $$p \leq -\log_\beta(2|x - \bar{x}|) < p + 1.$$
    Hint: Gebruik de definitie van juiste cijfers en merk op dat $p = -k$.

Afrondingsfouten in de computer

12. Lees paragraaf 2.5.3.

13. Schrijf een MATLAB programma om de machinenauwkeurigheid te bepalen. Gebruik hiervoor twee datatypes, `single` en `double`. Leg het verschil uit.

14. Zij `eps` de machinenauwkeurigheid. Leg het uit waarom de volgende computerbewerkingen verschillende resultaten opleveren:
    `1.0 - 1.0 + eps/2.0` en `1.0 + eps/2.0 - 1.0`

15. Stel dat bij het uitvoeren van een computerprogramma x de waarde 0.2 aanneemt. Leg uit waarom het resultaat van de toets "`is` $1.0 - x = 0.8$ `?`" onverwacht kan zijn. Implementeer dit in MATLAB met `single` resp. `double` nauwkeurigheid.

16. Lees paragraaf 2.5.5).

17. Is het *getallenbereik/floating point numbers* $\mathbb{F}$ van een computer eindig?

Voortplantingen van fouten doorheen bewerkingen

18. Lees nu paragraaf 2.5.6. Merk het verschil op tussen de effecten op de absolute en de relatieve fouten.

19. Leid de benadering af voor de absolute en de relatieve fouten op het product, $\Delta(xy)$ en $\delta(xy)$.

Les 2

Korte inhoud

Eerst bespreken we in deze les de conditie van een numeriek probleem en de stabiliteit van numerieke algoritmen. Vervolgens bestuderen we lineaire stelsels, een onderwerp dat ook voorgekomen is in het vak lineaire algebra. Hier worden er voornamelijk numerieke aspecten beschouwd.

Conditie van een numeriek probleem

Wetend nu dat afrondingsfouten en de getallenvoorstelling in de computeraritmetiek het resultaat van de elementaire bewerkingen kunnen beïnvloeden willen we dit verder onderzoeken voor het geval van meer algemene numerieke problemen. Merk op dat de getallenvoorstelling ook als een verstoring in de data geïnterpreteerd kan worden, nl. dat computers niet een algemeen getal $x \in \mathbb{R}$ als input gebruiken maar eigenlijk een benadering $\tilde{x} \in \mathbb{F}$.
In een algemenere context heeft dit aspect eigenlijk consequenties voor zowel de input gegevens als ook de parameters van een gegeven probleem. Ten eerste zijn de input gegevens slechts benaderingen van de verwachte gegevens en ten tweede gebruikt men tijdens de numerieke bewerkingen parameters die slechts een benadering zijn van de "echte" parameters. Het is dus zinvol om het effect van zulke verstoringen op de oplossing van het probleem te bestuderen. M.a.w. wat zijn de fouten in het resultaat bij gegeven verstoringen in de input data/parameters? Deze fouten zijn inherent verbonden met het probleem zelf, we spreken dus over *probleem-fouten*. In het ideale geval wordt de verstoring van het resultaat evenredig met de input- en de parameter-verstoringen, m.a.w. kleine verstoringen in de data leiden tot kleine verstoringen in het resultaat. Dan is er sprake van een een *goed geconditioneerd probleem*, anders is het probleem *slecht geconditioneerd*. Om dit te kunnen bepalen moeten we een "foutenanalyse" doen.

20. Zij $A = \begin{pmatrix} 1.2969 & 0.8648 \\ 0.2161 & 0.1441 \end{pmatrix}$ en $b = (0.8642, 0.1440)^T$ en beschouw een verstoring van de rechterzijde $b + \Delta b = (0.86419999, 0.14400001)^T$ (merk op de verstoring is $O(10^{-8})$!). Bepaal de oplossing x van het lineair stelsel $Ax = b$ en de verstoring die het stelsel $A(x+\Delta x) = b+\Delta b$ oplost. Is het probleem goed of slecht geconditioneerd?

21. Om het begrip "conditie van een probleem" te verduidelijken beschouwen we het algemene probleem: gegeven de grootheden $x = (x_1, \ldots, x_m)^T \in \mathbb{R}^m$ en de continu differentieerbare afbeelding $f = (f_1, \ldots, f_n)^T : \mathbb{R}^m \to \mathbb{R}^n$, bepaal de grootheden $y = (y_1, \ldots, y_n)^T \in \mathbb{R}^n$ als
$$y = f(x). \tag{1}$$
In het voorbeeld van boven is $m = n = 2$, $A \in \mathbb{R}^{2\times 2}$ - inverteerbaar en speelt b de rol van "input-data" x terwijl de rol van het resultaat y is overgenomen door x: $x = f(b) = A^{-1}b$.

Zij nu $x+\Delta x = (x_1+\Delta x_1, \ldots, x_m+\Delta x_m)^T \in \mathbb{R}^m$ de verstoorde input-data die leiden tot het resultaat $y+\Delta y = (y_1+\Delta y_1, \ldots, y_n+\Delta y_n)^T \in \mathbb{R}^n$. De absolute fouten zijn $|\Delta x_k|$ resp. $|\Delta y_i|$ en de relatieve fouten $|\Delta x_k|/|x_k|$ resp. $|\Delta y_i|/|y_i|$. Verder nemen we aan dat de relatieve fouten in de input data klein zijn, m.a.w. $|\Delta x_k| << |x_k|$ en we onderzoeken hun effect op de relatieve fouten in het resultaat.

Denkend aan Taylor reeksen geldt voor $i = 1, \ldots, n$

$$\begin{array}{rcl} y_i + \Delta y_i & = & f_i(x+\Delta x) = f_i(x_1+\Delta x_1, \ldots, x_m+\Delta x_m) \\ & = & f_i(x) + \sum_{j=1}^m \frac{\partial f_i}{\partial x_j}(x)\Delta x_k + R_{f,i}(x, \Delta x). \end{array} \tag{2}$$

Met $|\Delta x| = \max_{j=1,\ldots,m} |\Delta x_j|$ voor de restterm geldt $R_{f,i}(x, \Delta x) = o(|\Delta x|)$. In de eerste orde benadering kunnen dus de absolute en relatieve fouten zoals volgt geschreven worden $(i = 1, \ldots, n)$

$$\begin{array}{rcl} \Delta y_i & \approx & \sum_{j=1}^m \frac{\partial f_i}{\partial x_j}(x)\Delta x_j \qquad \text{en} \\ \frac{\Delta y_i}{y_i} & \approx & \sum_{j=1}^m \frac{\partial f_i}{\partial x_j}(x)\frac{\Delta x_j}{y_i} = \sum_{j=1}^m \frac{\partial f_i}{\partial x_j}(x)\frac{x_j}{f_i(x)}\frac{\Delta x_j}{x_j} = \sum_{j=1}^m \kappa_{ij}\frac{\Delta x_j}{x_j}. \end{array} \tag{3}$$

De getallen $\kappa_{ij} = \frac{\partial f_i}{\partial x_j}(x)\frac{x_j}{f_i(x)}$ $(i = 1, \ldots, n,\ j = 1, \ldots, m)$ heten *(relatieve) conditiegetallen*. Deze vormen de $n \times m$ matrix $\mathcal{K} = (\kappa_{ij})$ (de *conditie matrix*) en geven aan met welke factor de relatieve fouten in de gegevens opgeblazen worden in het resultaat. Als er indices i en j bestaan waarvoor $|\kappa_{ij}| >> 1$ dan is het probleem slecht geconditioneerd (de relatieve fouten kunnen verveelvoudigd worden), anders is het probleem goed geconditioneerd. Als $|\kappa_{ij}| \leq 1$ voor alle indices i en j dan worden de relatieve fouten gedempt.

In een beknopte formulering kan men de bovenstaande analyse samenvatten als

`relatieve fout op het resultaat` $\approx$
`conditie-matrix` $\times$ `relatieve fout op de input data/parameters`

De analyse voor absolute fouten is gelijkaardig. In dit geval zijn de elementen van $\nabla f(x)$, de Jacobi matrix van f in x, de *(absolute) conditiegetallen*.

22. Dezelfde vragen kunnen in de tegenovergestelde richting gesteld worden. Onder de aanname dat de afbeelding f inverteerbaar is, m.a.w. gegeven het resultaat $y \in \mathbb{R}^n$ bestaat er een uniek m-tupel $x \in \mathbb{R}^m$ waarvoor geldt $y = f(x)$. Dan geldt ook $x = f^{-1}(y)$. Merk op, de vraag kan eventueel beperkt worden tot een deelverzameling in het domein van f en het beeld van deze verzameling door f. Dezelfde soort foutenanalyse kan gebruikt worden om een antwoord te geven aan de vraag: hoe groot mag de fout in de input-data zijn om de garantie te hebben dat de verstoring in het resultaat beperkt blijft?

23. Wat is de betrekking tussen de conditie matrices voor het directe probleem en het inverse probleem?

24. Maak oefening 1 op blz. 54.

25. Bestudeer de conditionering van de som en het product van twee getallen. M.a.w. bestudeer de conditionering van het probleem $y = f(x_1, x_2)$ met

$$a) f(x_1, x_2) = x_1 + x_2, \qquad \text{en} \qquad b) f(x_1, x_2) = x_1 x_2.$$

(Zie ook de vraag bij punt 19.)

26. Zij nu $p, q \in \mathbb{R}$ met $q \neq 0$ en $p^2 > 4q$. Bestudeer de conditionering van het probleem $y = f(p, q)$ waarbij y_1, y_2 de twee oplossingen zijn van de vergelijking $y^2 - py + q = 0$. Hoe is dit probleem geconditioneerd in de volgende gevallen

$$a) p = 2, q = 0.01, \qquad \text{en} \qquad b) p = 2, q = 0.999.$$

Stabiliteit van een algoritme

Beginnend bij een numeriek probleem zoals in (1) veronderstellen we nu dat dit d.m.v. een aantal stappen opgelost kan worden. M.a.w. we praten over *algoritmen* om het antwoord $y \in \mathbb{R}^n$ te verkrijgen of te benaderen (bij gegeven $x \in \mathbb{R}^m$). Het aantal stappen kan eindig (zoals bij een directe berekening) of oneindig zijn (zoals bij een iteratieve methode). In een abstracte redenering laten we $\varphi^{(j)}$ $(j = 1, 2, \dots)$ de stappen zijn van het algoritme. Zonder verlies van de algemeenheid zij $\tilde{y}$ de benadering van y (als het aantal stappen oneindig zijn, of als de berekening niet tot het einde wordt gebracht vanwege onvoldoende rekentijd/kracht), of $\tilde{y} = y$ bij een exacte berekening. De berekening van $\tilde{y}$ kan als volgt beschreven worden

$$x^{(0)} := x \to x^{(1)} := \varphi^{(1)}(x^{(0)}) \to x^{(2)} := \varphi^{(2)}(x^{(1)}) \to \cdots \to x^{(j+1)} := \varphi^{(j)}(x^{(j)}) \to \dots \tilde{y}.$$

In het simpelste geval zijn de stappen $\varphi^{(j)}$ elementaire bewerkingen. Met elke stap bij het uitvoeren van dit algoritme komen er nieuwe fouten bij vanwege afrondingen, computer voorstelling van getallen, benaderingen van functies zoals $\sin, \exp$, enz. Deze fouten stapelen zich op en kunnen tot een onverwacht resultaat leiden. Merk op dat dit effect is strikt te wijten aan het algoritme (*numerieke fouten*) en staat los van de fouten die kunnen optreden als gevolg van de conditie van het probleem, bijv. door verstoringen in de input data (probleem-fouten). We spreken dus over twee soorten fouten: de eerste soort is verbonden aan de conditie van het probleem en de tweede soort is bepaald door de manier waarop het resultaat wordt berekend, dus door de methode/het algoritme zelf. Een algoritme is *stabiel* als de numerieke fouten niet de probleem-fouten overschrijden. M.a.w. heeft het algoritme geen dominante bijdrage tot de totale fout in het resultaat. Als de fouten in het resultaat voornamelijk het algoritme als bron hebben dan is het algoritme *instabiel*.

Een van de belangrijkste taken van de numerieke analyse is het ontwikkelen van (numerieke) algoritmen en methoden moeten die zowel stabiel als ook convergent zijn (een aspect dat later in het college besproken zal worden).

27. Om een voorbeeld te geven van een simpele *stabiliteitsanalyse* beschouwen we het volgende probleem:

`Gegeven` $x_1, x_2 \in \mathbb{R}$, `bereken` $y = f(x_1, x_2) := x_1^2 - x_2^2$.

Neem aan dat de verstoringen in de input data leiden tot een relatieve fout die onder de machinenauwkeurigheid `eps` ligt. Dan geldt voor de relatieve probleem-fout

$$\left|\frac{\Delta y}{y}\right| \approx \sum_{k=1}^{2} \left|\frac{\partial f}{\partial x_k}(x_1, x_2)\right| \left|\frac{x_k}{f(x_1, x_2)}\right| \left|\frac{\Delta x_k}{x_k}\right| \leq 2 \left|\frac{(x_1/x_2)^2 + 1}{(x_1/x_2)^2 - 1}\right| \texttt{eps}, \qquad (4)$$

(zie ook (3)). De uitdrukking $x_1^2 - x_2^2$ kan op twee manieren berekend worden, $x_1^2 - x_2^2$ en $(x_1 - x_2)(x_1 + x_2)$. We beschouwen dus twee "algoritmen" (de notaties $\otimes, \oplus, \ominus$ zijn hieronder uitgelegd)

Algoritme A: $u = x_1 \otimes x_1, v = x_2 \otimes x_2, w = u \ominus v$

Algoritme B: $u = x_1 \oplus x_2, v = x_1 \ominus x_2, w = u \otimes v$

28. Merk op dat bij het uitvoeren van deze algoritmen eigenlijk computerbewerkingen gebruikt worden en daarom is het resultaat van de een standaard bewerking (zoals $+$, $-$ of $\times$) slechts een benadering. Dit verklaart de notaties $\oplus$, $\ominus$ en $\otimes$. Merk op dat voor een bewerking $\cdot \in \{+, -\times, /\}$ geldt

$$a \odot b = (a \cdot b)(1 + \varepsilon),$$

voor een zekere ε die voldoet aan $|\varepsilon| \leq \texttt{eps}$. Hier zijn de fouten als gevolg van een computervoorstelling van a en b niet meegenomen in de analyse. Anders gezegd zijn a en b als computergetallen beschouwd. Als dit niet het geval is dan moeten er i.p.v. a en b hun benaderingen in de verzameling van computergetallen $\mathcal{G}$ beschouwd worden, dus eigenlijk $a(1 + \varepsilon_a)$ en $b(1 + \varepsilon_b)$ met $\varepsilon_a, \varepsilon_b$ waarvoor geldt $|\varepsilon_\alpha| \leq \texttt{eps}$ $(\alpha \in \{a, b\})$.

29. Wat zijn de fouten in de computerbewerking van $a \cdot b$ in het geval dat a en b geen computergetallen zijn?

30. Om de numerieke fouten af te scheiden van de probleem-fouten nemen we aan dat x_1 en x_2 computergetallen zijn. Daarom zijn er geen fouten in de input data of parameters. Gebaseerd op de discussie van boven krijgt men voor Algortime A

$u = x_1^2(1 + \varepsilon_1), v = x_2^2(1 + \varepsilon_2),$
$w = (x_1^2(1 + \varepsilon_1) - x_2^2(1 + \varepsilon_2))(1 + \varepsilon_3) = (x_1^2 - x_2^2) + (x_1^2 - x_2^2)\varepsilon_3 + x_1^2\varepsilon_1 - x_2^2\varepsilon_2 + O(\texttt{eps}^2)$

Na het verwaarlozen van de $O(\texttt{eps}^2)$ termen en met $y = f(x_1, x_2) = x_1^2 - x_2^2$ geldt

$$\left|\frac{\Delta y_i}{y_i}\right| \leq \texttt{eps} \frac{x_1^2 + x_2^2 + |x_1^2 - x_2^2|}{|x_1^2 - x_2^2|} = \texttt{eps} \left[1 + \frac{(x_1/x_2)^2 + 1}{|(x_1/x_2)^2 - 1|}\right].$$

De numerieke fout wordt dus zeer groot als $|x_1| \approx |x_2|$. Desondanks wordt de orde van de probleem-fout niet overschreden en is dus Algoritme A in deze zin stabiel.

31. Herhaal deze analyse voor Algoritme B. Het resultaat wordt veel simpeler,

$$\left|\frac{\Delta y_i}{y_i}\right| \leq 3\texttt{eps}.$$

Merk op dat deze foutschatting niet van de input data x_1, x_2 afhangt en is beter dan die voor Algoritme B (ten minste onder de aanname dat $\varepsilon_1 \neq \varepsilon_2$). Het verschil in de twee algoritmen is dat bij A eerst de vermenigvuldigingen worden gedaan en als laatste de aftrekking, terwijl bij B de vermenigvuldiging de laatste bewerking is. Denkend aan de conclusie van boven over de conditionering van de elementaire bewerkingen leidt dit tot de volgende

Regel 1 Bij het uitvoeren van numerieke bewerkingen is het aanbevolen eerst de bewerkingen die slechter geconditioneerd zijn uit te voren en de beter geconditioneerde als laatste te gebruiken.

Vanzelfsprekend moet het resultaat van de twee algoritmen hetzelfde blijven als er geen sprake is van numerieke fouten of verstoringen, m.a.w. als alles exact wordt uitgevoerd.

32. Het bepalen van de oplossingen van kwadratische vergelijkingen.
Zij $p, q \in \mathbb{R}$ met $q \neq 0$, $p > 0$ en $p^2 > q$. Zij $y_{1,2} = \frac{p}{2} \pm \frac{1}{2}\sqrt{p^2 - 4q}$ de oplossingen van de vergelijking $y^2 - py + q = 0$. Om deze te bepalen beginnen we met de volgende berekeningen

$$u = p^2, v = u - 4q, w = \sqrt{v} (\geq 0).$$

Omdat p en w hetzelfde teken hebben berekenen we eerst $y_1 = \frac{1}{2}(p + w)$ (waarom?). Merk op dat het resultaat van de computerbewerking slechts een benadering van y_1 is, $\tilde{y}_1$. Vervolgens kunnen we zoals volgt doorgaan:

$$A : y_2 = \frac{1}{2}(p - w) \qquad\qquad B : y_2 = \frac{q}{y_1}.$$

Welke van de twee methoden is stabieler? Voor welke input data p, q is de onderscheiding belangrijk?

33. Het berekenen van de waarde $P(x)$ voor een veelterm P.
Zij $a_1, a_2 \in \mathbb{R}$ gegeven en beschouw de veelterm $P(x) = a_2x^2 + a_1x$. De waarde in het punt $x \in \mathbb{R}$ kan op twee manieren bepaald worden, volgens de gegeven formule of als $P(x) = x(a_1 + a_2x)$. In welke gevallen zijn de numerieke fouten kleiner? Voor welke waarden van x worden de numerieke fouten zeer groot?

Zij nu $a_0, \ldots, a_n \in \mathbb{R}$ gegeven en beschouw de veelterm $P(x) = a_nx^n + \cdots + a_1x + a_0$. Geef een uitleg van de volgende

Regel 2 Gegeven $x \in \mathbb{R}$, het berekenen van $P(x)$ is het aanbevolen om het Hornerschema te gebruiken.

Les 3

Lineaire stelsels, methode van Gauß

Korte inhoud

Voor het oplossen van lineaire stelsel $AX = B$ bestuderen we directe methoden die gebaseerd zijn op de methode van Gauß oftewel GEM (Gauß Elimination Method). Directe methoden zijn methoden die de oplossing exact opleveren binnen een eindig aantal stappen als de bewerkingen niet beïnvloedt zijn door fouten. In het tweede numerieke vak zullen er ook iteratieve methoden gepresenteerd worden.

34. Zij $n \in \mathbb{N}$ oneven. Beschouw de matrices $A, B \in \mathbb{R}^{n\times n}$

$$A = \begin{pmatrix} 2 & -1 & & & & & & & \\ -1 & 2 & -1 & & & & & & \\ & -1 & 2 & -1 & & & & & \\ & & -1 & 2 & -1 & & & & \\ & & & \ddots & \ddots & \ddots & & & \\ & & & & \ddots & \ddots & \ddots & & \\ & & & & & -1 & 2 & -1 & \\ & & & & & & -1 & 2 & -1 \\ & & & & & & & -1 & 2 \end{pmatrix}$$

en

$$B = \frac{1}{n+1}\begin{pmatrix} n & n-1 & n-2 & \dots & \dots & 3 & 2 & 1 \\ n-1 & 2(n-1) & 2(n-2) & \dots & \dots & 6 & 4 & 2 \\ n-2 & 2(n-2) & 3(n-2) & \dots & \dots & 9 & 6 & 3 \\ \vdots & \vdots & \vdots & & & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & & & \vdots & \vdots & \vdots \\ 3 & 6 & 9 & \dots & \dots & 3(n-2) & 2(n-2) & n-2 \\ 2 & 4 & 6 & \dots & \dots & 2(n-2) & 2(n-1) & n-1 \\ 1 & 2 & 3 & \dots & \dots & n-2 & n-1 & n \end{pmatrix}.$$

A komt voor in de numerieke benadering van oplossingen van wiskundige modellen die bijv. *diffusie* processen beschrijven (warmte flux, concentratie van een stof die opgelost is in het water, ezv.). De elementen van B kunnen geschreven worden als

$$b_{ij} = \begin{cases} \frac{(n+1-i)j}{n+1}, & \text{als } j < i, \\ \frac{(n+1-j)i}{n+1}, & \text{als } j \geq i. \end{cases}$$

(a) Toon aan dat B de inverse is van A.

(b) Zij verder de kolomvector $c \in \mathbb{R}^n$, $c = \frac{1}{n^2}(1, -1, 1, \dots, (-1)^{n-1})^T$. Bepaal de oplossing $x \in \mathbb{R}^n$ van het stelsel $Ax = c$.

35. **Substitutie:** Deze methode is toepasbaar voor stelsels waarbij de matrix A een speciale vorm heeft, nl. driehoekig. Dat betekent dat of alle elementen onder de hoofddiagonaal zijn 0 voor een bovendriehoekige matrix, resp. alle elementen boven de hoofddiagonaal zijn 0 voor een benedendriehoekige matrix. Lees paragraaf 3.2.1, blz. 66–69 (supplementair blz. 44–45 in het boek van A. Bultheel) over *boven- en benedendriehoekige stelsels/traingular systems* en de *achterwaartse/voorwaartse substitutie*. Het is voldoende om alleen de rij-versies te bestuderen.

36. *Complexiteit:* Zij $A \in \mathbb{R}^{n\times n}$ driehoekig en $B \in \mathbb{R}^n$ gegeven. Bereken het aantal vermenigvuldigingen en het aantal optellingen die ervoor nodig zijn om de oplossing X van het stelsel $AX = B$ te bepalen. Dit is een voorbeeld van *complexiteit* analyse.

37. Schrijf een MATLAB functie die voor een gegeven benedendriehoekige matrix $L \in \mathbb{R}^{n\times n}$ en een vector $B \in \mathbb{R}^n$ de oplossing van de vergelijking $LX = B$ oplevert. Doe dit ook voor bovendriehoekige matrices, dus bereken de oplossing van het stelsel $UX = B$. Je mag hier ook de programma's in het boek gebruiken als inspiratiebron. Leg de rol van de commando `if min(abs(diag(L))) == 0 ...` uit. Zou je deze eventueel anders implementeren?

38. **Eliminatie:** Lees paragraaf 3.3 blz. 70–73 (supplementair blz. 46–47 in het boek van A. Bultheel). Let op, voor de *spilelementen/pivots* moet gelden $a_{ii}^{(i-1)} \neq 0$ (hier is $a_{11}^{(0)} = a_{11}$)! Bereken alweer het aantal bewerkingen voor de eliminatie.

Gauß-Eliminatie Methode

In deze sectie leren we over de methode van Gauß vor het oplossen van linearie stelsels en de LU (of LR) decompositie. De presentatie is te vinden in paragraaf 3.3.1 tot Theorem 3.4; deze stelling hoeft niet bestudeerd te worden (zie ook paragraaf 2.2–2.5 in het boek van A. Bultheel). Het doel is het stelsel tot een simpele vorm (triangulair) te brengen. Voor een stelsel in deze vorm is een oplossing makkelijk te vinden via substitutie (terugwaarts of voorwaarts).

39. *Elementaire transformaties en matrices*
De *Gauß-Eliminatie Methode (GEM)* is gebaseerd op een *elementaire transformatie*: het aftrekken van m keer rij i van rij k. In een latere fase zullen er nog twee transformaties komen: het verwisselen van de rijen i en k en het vermenigvuldigen van rij i met $m \in \mathbb{R}/\{0\}$. Deze zullen later gebruikt worden om de methode stabieler te maken. Alle drie transformaties veranderen de oplossingsverzameling niet. M.a.w., het stelsel dat we krijgen na het toepassen van een elementaire transformatie heeft precies dezelfde oplossing(en) als het oorspronkelijke stelsel.

    De elementaire transformaties kunnen als matrix-bewerkingen geschreven worden:

zij $T_{i,k}$, $T_{i,k}(m)$ en $T_i(m)$ de $n \times n$ matrices (met $i,k \in \{1,\ldots,n\}$ en $m \in \mathbb{R}$)

$$T_{i,k} = \begin{pmatrix} 1 & 0 & \ldots & 0 & \ldots & 0 & \ldots & 0 \\ 0 & 1 & \ldots & 0 & \ldots & 0 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & 0 & \ldots & \mathbf{0} & \ldots & \mathbf{1} & \ldots & 0 \\ \vdots & \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & 0 & \ldots & \mathbf{1} & \ldots & \mathbf{0} & \ldots & 0 \\ \vdots & \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 0 & \ldots & 0 & \ldots & 1 \end{pmatrix}, T_{i,k}(m) = \begin{pmatrix} 1 & \ldots & 0 & \ldots & 0 & \ldots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \ldots & 1 & \ldots & 0 & \ldots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \ldots & \mathbf{-m} & \ldots & 1 & \ldots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \ldots & 0 & \ldots & 0 & \ldots & 1 \end{pmatrix}$$

en

$$T_i(m) = \begin{pmatrix} 1 & \ldots & 0 & \ldots & 0 \\ \vdots & \ddots & \vdots & & \vdots \\ 0 & \ldots & \mathbf{m} & \ldots & 0 \\ \vdots & & \vdots & \ddots & \vdots \\ 0 & \ldots & 0 & \ldots & 1 \end{pmatrix}$$

Merk op dat deze licht afwijken van de eenheidsmatrix: in $T_{i,k}$ geldt voor de elementen in de rijen i en k en dezelfde kolommen $t_{ik} = t_{ki} = 1$, $t_{ii} = t_{kk} = 0$, in $T_{ik}(m)$ geldt $t_{ki} = -m$ en in $T_i(m)$ geldt $t_{ii} = m$.

40. Geef een bewijs voor de volgende
Propositie. Er geldt:

(a) Voor alle $i,k \in \{1,\ldots,n\}$ en $m \in \mathbb{R}$ zijn $T_{i,k}$ en $T_{i,k}(m)$ inverteerbaar en er geldt $T_{i,k}^{-1} = T_{i,k}$, resp. $T_{i,k}(m)^{-1} = T_{i,k}(-m)$. Als $m \neq 0$ dan is ook $T_i(m)$ inverteerbaar en er geldt $T_i(m)^{-1} = T_i(\frac{1}{m})$.

(b) Voor alle $i,k \in \{1,\ldots,n\}$ en $m \in \mathbb{R}$ geldt $det(T_{i,k}) = -1$, $det(T_{i,k}(m)) = 1$ en $det(T_i(m)) = m$.

(c) Zij $k,p > i$ en $m,n \in \mathbb{R}$. Dan geldt $G = T_{i,k}(m)T_{i,p}(n) = T_{i,k}(m) + T_{i,p}(n) - I$ met I de eenheidsmatrix. Merk op dat in de matrix G de hoofddiagonaal alleen uit elementen 1 bestaat en alle andere elementen 0 zijn m.u.v. van g_{ki} en g_{pi} die $-m$, resp. $-n$ zijn.

(d) Zij $i \neq j$, $k > i$, $p > j$ en $m,n \in \mathbb{R}$. Dan geldt $T_{i,k}(m)T_{j,p}(n) = T_{i,k}(m) + T_{j,p}(n) - I$.

(e) Zij $G^{(i)} = I + M$ met $m_{rk} = 0$ als $k \neq i$ of als $r \leq i$ (dus met niet-nul elementen alleen de hoofddiagonaal en onder de hoofddiagonaal in de i-de kolom). Als $k < p$ dan geldt $G^{(k)}G^{(p)} = G^{(k)} + G^{(p)} - I$. Als $k > p$ dan geldt $G^{(k)}G^{(p)} = I + M$ met $m_{rk} = 0$ als $k \neq p$ of $k \neq q$ en als $r \leq p$ voor m_{rp}, respectievelijk als $r \leq q$ voor m_{rq}. M.a.w. de niet-nul elementen van M staan in de kolommen p en q, onder de hoofddiagonaal.

41. Geef een bewijs voor de volgende:
Propositie.
a) Zij $L \in \mathbb{R}^{n\times n}$ benedendriehoekig (dus $l_{ij} = 0$ als $j > i$) en inverteerbaar. Dan is L^{-1} tevens benedendriehoekig.
b) Zij $L_1, L_2 \in \mathbb{R}^{n\times n}$ benedendriehoekig. Dan is het product L_1L_2 een benedendriehoekige matrix.
Een gelijkaardig resultaat geldt ook voor bovendriehoekige matrices.

LU/LR-decompositie

We schrijven de matrix A als product van twee triangulaire matrices te schrijven, L (beneden/lower/left)-traingulair en U (boven/upper/right) triangulair. Dan kan een oplossing van $AX = B$ makkelijk bepaald worden als twee substituties, voorwaarts en achterwaarts.

42. Merk op dat de GEM als een matrixbewerking geschreven kan worden. Veronderstel dat voor de matrix A geldt dat bij elke stap i ($i = 1, \ldots, n-1$), $a_{ii} \neq 0$. M.a.w. het element op de hoofddiagonal kan altijd als spilelement genomen worden. Bijvoorbeeld de *diagonaal gedomineerde matrices*, die later besproken worden, hebben wel deze eigenschap maar dit geldt niet voor alle (inverteerbare) matrices. De GEM zal later aangepast worden voor stelsels met inverteerbare matrices. Deze methode heet GEM *met pivotering*.

43. *Spil/pivot en eliminatie.* Als $a_{ii} \neq 0$ dan wordt a_{ii} de spil. De elementen onder de spil worden 0 gemaakt (geëlimineerd) m.b.v. het aftrekken van m keer rij i van rij k ($(k > i)$ voor een geschikt gekozen getal m. Dat betekent de vermenigvuldigingen met de matrices $T_{i,k}(m_{i,k})$ waarbij $m_{i,k} = \frac{a_{ki}}{a_{ii}}$. Volgens de proposite van punt 40 is het product $G^{(i)} = T_{i,i+1}(m_{i,i+1}) \ldots T_{i,n}(m_{i,n})$ een matrix met alleen niet-nullen op de hoofddiagonaal (deze elementen zijn 1) en onder de hoofddiagonaal in kolom i.

44. *LU-decompositie.* Met $A^{(1)} = A$ zij $A^{(i)} = G^{(i-1)}A^{(i-1)} = G^{(i-1)} \ldots G^{(1)}A$ ($i = 2, \ldots, n$). Merk op dat $A^{(n)}$ is het resultaat van de GEM en dus een bovendriehoekige matrix. We bestuderen de structuur van dit product en tonen aan dat eigenlijk dit tot de vorm van een product LU met L benedendriehoekige matrix en U bovendriehoekige matrix gebracht kan worden.

    Voor de eenvoud zijn hier de stappen gegeven alleen voor de matrix A maar de uitbreiding tot de uitgebreide matrix $[A|b]$ van het stelsel $Ax = b$ is rechttoe-rechtaan.

45. We beschouwen nog steeds het bijzondere geval waarin de GEM zoals boven (dus zonder pivotering) uitvoerbaar is. Er geldt:
Stelling (zie ook Theorem 3.4 op blz. 76). Zij $A \in \mathbb{R}^{n\times n}$ een inverteerbare matrix waarvoor de GEM zonder pivotering uitvoerbaar is. Dan bestaat er een unieke decompositie $A = LU$ met $L \in \mathbb{R}^{n\times n}$ benedendriehoekig, $l_{ii} = 1$ voor alle $i = 1, \ldots, n$

en $U \in \mathbb{R}^{n\times n}$ bovendriehoekig.
Bewijs. Voor de eenduidigheid zij $A = L_1U_1$ en $A = L_2U_2$ twee LU decomposities. Dan geldt $L_1U_1 = L_2U_2$ en alle matrices zijn inverteerbaar. Vervolgens geldt $L_2^{-1}L_1 = U_2U_1^{-1}$. Maar volgens de propositie bij punt 41 is $L_2^{-1}L_1$ benedendriehoekig en $U_1^{-1}U_2$ bovendriehoekig. Omdat op de hoofddiagonaal van L_1 en L_2 alleen 1 voorkomt geldt hetzelfde ook voor het product $L_2^{-1}L_1$. Daarom geldt

$$L_2^{-1}L_1 = U_2U_1^{-1} = I \quad \text{(de eenheidmatrix)}$$

en hieruit volgt de eenduidigheid.

Voor de LU decompositie zonder pivotering geldt $A^{(n)} = G^{(n-1)} \dots G^{(1)}A$ met $A^{(n)}$ bovendriehoekig. Daarom geldt

$$A = (G^{(1)})^{-1} \dots (G^{(n-1)})^{-1}A^{(n)}.$$

Voor elke $i = 1, \dots, n-1$ geldt dat $G^{(i)} = T_{i,i+1}(m_{i,i+1}) \dots T_{i,n}(m_{i,n})$ en volgens de proposities bij punt 40 geldt $(G^{(i)})^{-1} = T_{i,n}(-m_{i,n}) \dots T_{i,i+1}(-m_{i,i+1})$. Daarom is $(G^{(i)})^{-1}$ benedendreihoekig en uit punt 41 volgt dat het product $L = (G^{(1)})^{-1} \dots (G^{(n-1)})^{-1}$ ook benedendriehoekig is en $g_{ii} = 1$ voor $i = 1, \dots, n$. Met $U = A^{(n-1)}$ hebben we dus A geschreven als het product $A = LU$ met L benedendrihoekig en U bovendriehoekig.

46. *Ter info:* Ga naar paragraaf 3.3.3 en 3.3.4 voor praktische aspecten over LU-factorisatie.

47. **Pivotering:** Lees paragraaf 3.5 over pivotering (supplementair paragraaf 3.1–3.2 in het boek van A. Bultheel). Leg uit waarom pivotering nodig is. De *complete/totale pivotering* laten we achterwege.

48. *(Optimale) rijpivotering/partial pivoting.* De keuze van spilelementen kan op verschillende manieren gebeuren. De spil in kolom i kan bijvoorbeeld het eerste niet-nul element a_{ki}^{i-1} $(i \le k \le n)$ genomen worden. Dit is eigenlijk ook de manier van werken tot nu toe, onder de aanname dat pivotering niet nodig is. Anderijds, door de spilelementen handig te kiezen wordt de eliminatie met pivotering stabieler. Om preciezer te zijn, zij $\left[A^{(i-1)}|B^{(i-1)}\right]$ de uitgebreide matrix na stap $i-1$ van de eliminatie met pivotering. Met de notatie op blz. 88 geldt $a_{kj}^{(k-1)} = 0$ voor alle $k < i$ en $j > k$. Dat is het resultaat van de eliminatie t/m stap $i-1$. Om de spil in de kolom i te kiezen zoeken we eerst $k \in i, \dots, n$ z.d.

$$|a_{ki}^{(i-1)}| = \max_{r=i,\dots,n} |a_{ri}^{(i-1)}| \tag{5}$$

en vervolgens verwisselen we de rijen i en k. Dit heet *optimale rijpivotering/partial pivoting* of *Gauß eliminatie met optimale rijverwisseling.* Vervolgens wordt het stelsel getransformeerd z.d. de elementen onder de spil in kolom i 0 worden. Bestudeer ook de voorbeelden 3.5 en 3.6.

49. Net zoals bij punt 43 geldt dat de GEM met pivotering tevens als een matrixbewerking geschreven kan worden. Bij stap i ($i = 1, \ldots, n-1$) wordt eerst een spil gezocht in de i-de kolom. Stel dat deze spil het element a_{ki} is. Dan wordt het een rijverwisseling gedaan, m.a.w. een vermenigvuldiging met de (permutatie-)matrix $P^{(i)} = T_{i,k}$ (in het geval $a_{ii} \neq 0$ is de rijverwisseling niet nodig en dus kan $P^{(i)} = I$, de eenheidmatrix, genomen worden). De rest volgt zoals eerder. M.a.w. in het matrix-product zoals bij punt 44 komen er nog de permutatiematrices $P^{(i)}$.

    Met $A^{(1)} = A$ en $G^{(i)} = T_{i,i+1}(m_{i,i+1}) \ldots T_{i,n}(m_{i,n})$ het product dat de eliminatie onder de spil uitvoert geldt dat $A^{(i+1)} = G^{(i)}P^{(i)}A^{(i)} = G^{(i)}P^{(i)} \ldots G^{(1)}P^{(1)}A$ ($i = 1, \ldots, n-1$). Alweer is $A^{(n)}$ het resultaat van een GEM (nu met rijpivotering) en dus een bovendriehoekige matrix.

50. *Merk op*: de GEM met pivotering brengt het stelsel $Ax = b$ tot de vorm $LUx = Pb$. Leg uit waarom.

51. *Ter info*: De stelling bij punt 45 wordt gegeven voor het geval dat in de GEM geen pivotering nodig is. Een gelijkaardig resulaat geldt ook voor de GEM met pivotering, namelijk dat $LU = PA$. Hier is $P = P^{(n-1)} \ldots P^{(1)}$ het product van ale matrices $T_{i,k}$ die de rijverwisselingen beschrijven. Dan is de LU-decompositie niet eenduidig.

52. Schrijf een MATLAB functie die voor een gegeven matrix $A \in \mathbb{R}^{n \times n}$ en een vector $B \in \mathbb{R}^n$ de oplossing van de vergelijking $AX = B$ oplevert. Gebruik hiervoor twee eliminatie methoden, met simpele rijpivotering (na stap $i-1$ wordt rij i verwisseld met de eerste rij van boven naar beneden waarvoor geldt $a_{ri}^{(i-1)} \neq 0$ met $r \geq i$) en met optimale rijpivotering. Gebruik dit om het stelsel $AX = B$ op te lossen met A de matrix bij punt 34 en $b = -(n+1)^2(f(x_1), \ldots, f(x_n))^T$ met $x_k = \frac{k}{n+1}$ en $f(x) = x(1-x)$ (hier is $n \in \mathbb{N}$ een parameter).

53. De *Hilbert matrix* $H^{(n)} \in \mathbb{R}^{n \times n}$ is een bekend voorbeeld van een slecht geconditioneerde matrix. De elementen van $H^{(n)}$ zijn $h_{ij} = \frac{1}{i+j-1}$, $i, j = 1, \ldots, n$. Zij b de vector met alle elementen 1. Gebruik je Matlab programma om het stelsel $H^{(n)}x = b$ op te lossen. Probeer dit in zowel `single` als ook `double` nauwkeurigheid. Ga na of er fouten zijn op het resultaat.

54. Een mogelijkheid om de eliminatie nog stabieler te krijgen is de *totale pivotering*. Dit wordt vooral gebruikt als de elementen van A sterk van elkaar afwijken, nl. ze hebben sterk verschillende grootorden. In dit geval kiest men de spil z.d.

$$|a_{kj}^{(i-1)}| = \max_{r,s=i,\ldots,n} |a_{rs}^{(i-1)}|. \tag{6}$$

    Vervolgens worden er niet alleen de rijen i en k verwisseld in de uitgebreide matrix van het stelsel maar ook de kolommen i en j. Blijft het nieuwe stelsel in dit geval equivalent met het oorspronkelijke? Leg uit en als dit niet het geval is, wat moet men gedaan worden om aan het eind de oplossing terug te vinden?

Les 4

Andere aspecten die gerelateerd zijn aan de GEM

In deze paragraaf worden er verschillende aspecten besproken die gerelateerd zijn aan de GEM: hoe kan de methode verbeterd worden en wat kan men doen in het geval van speciale matrices?

55. Bestudeer paragraaf 3.6 (supplementair blz. 58–59 in het boek van A. Bultheel) over het berekenen van de inverse van een matrix.

56. **Voorconditionering** (zie ook paragraaf 3.12.1)
We hebben gezien dat kleine spilelementen een negatieve invloed hebben op de nauwkeurigheid van de berekening. De optimale rijpivotering is een mogelijke oplossing hiervoor maar deze kan gecombineerd worden met een simpele transformatie *scaling, equilibration*. Zij D de matrix met alle elementen 0 behalve op de hoofddiagonaal, waarvoor geldt

$$d_{ii} = \big(\sum_{k=1}^{n} |a_{ik}|\big)^{-1}, \quad i = 1, \ldots, n.$$

Alternatief kan $d_{ii} = \big(\max_{k=1,\ldots,n} |a_{ik}|\big)^{-1}$ genomen worden. Merk op dat in beide gevallen D de inverse matrix is van een "benadering" van A. Deze benadering heeft alleen op de hoofddiagonaal die niet-nul elementen en kan dus makkelijk geïnverteerd worden. Beschouw dan het getransformeerde stelsel

$$DAx = Db.$$

Dit is equivalent met $Ax = b$ maar heeft het voordeel dat in de nieuwe matrix DA de elementen ongeveer dezelfde grootte hebben. Dit is een simpele vorm van een zgn. *voorconditionering* waarbij het stelsel $Ax = b$ vervangen wordt door het stelsel $PAx = Pb$. Hier is P een inverteerbare matrix en daarom zijn de twee stelsels equivalent. De matrix P moet aan twee condities voldoen: P (of een benadering hiervan) moet makkelijk te berekenen zijn en het conditiegetal $\mathcal{K}(PA)$ is kleiner dan dat van A. Zoals reeds gezien heeft dit getal een grote invloed op de nauwkeurigheid. Extreem-situaties zijn $P = I$ (de eenheidsmatrix) en $P = A^{-1}$. Leg uit waarom deze twee niet praktisch toepasbaar zijn. De keuze $P = D$ met D zoals boven is een redelijk compromis maar voor elk specifiek probleem kunnen er betere keuzes van P gemaakt worden.

57. Beschouw het stelsel van 2 vergelijkingen en 2 onbekenden

$$\begin{pmatrix} 1 & 99 \\ 0.0029 & -0.0001 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 500 \\ 0.014 \end{pmatrix}.$$

De exacte oplossing wordt gegeven door $(x_1, x_2) = (5, 5)$. Gebruik de eenvoudige methode van Gauß om het stelsel op te lossen. Voer hierbij de berekeningen uit in

bewegende komma met 4 beduidende cijfers. Bereken vervolgens het conditiegetal van de matrix

$$A := \begin{pmatrix} 1 & 99 \\ 0.0029 & -0.0001 \end{pmatrix}$$

Je kunt hiervoor zowel de inverse matrix A^{-1} expliciet berekenen als ook het MATLAB bevel `cond` gebruiken met `inf` als tweede argument. Gebruik nu een simpele transformatie D (scaling) die het conditiegetal van de matrix A verkleint en bereken het conditiegetal van de matrix DA. Pas opnieuw de methode van Gauss toe op het getransformeerde stelsel

$$PA \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = P \begin{pmatrix} 500 \\ 0.014 \end{pmatrix}.$$

58. *Methode van Gauß met voorconditionering*: Beschouw de matrix $B \in \mathbb{R}^{n \times n}$ uit punt 34. Gebruik het Matlab programma uit punt 52 om het stelsel

$$Bx = b, \qquad b = (1, 1, \ldots, 1)^T \in \mathbb{R}^n,$$

op te lossen voor $n = 1, 2, \ldots, 100$. Maak vervolgens een grafiek die n uitzet tegenover de fout

$$\|Bx - b\|_2,$$

waarbij $\|v\|_2 = \left(v_1^2 + \cdots + v_n^2\right)^{\frac{1}{2}}$ voor de *vectornorm* van $v = (v_1, \ldots, v_n)^T \in \mathbb{R}^n$ staat. Verbeter de fout door gebruik te maken van de voorconditioneringsmethode uit punt 56. Voeg deze fout toe aan de grafiek.

59. **Sparse matrices**
Een *sparse matrix* (*ijl, dunbezet*, of *schaars*) heeft de meeste elementen 0. Sparse matrices komen vaak voor als resultaat van bijv. numerieke discretisatie van differentiaalvergelijkingen. Vooral als de matrix groot is ($n >> 1$) en als het aantal niet-nul elementen $O(n)$ is wordt het voordelig om zulke matrices niet als een matrix op te slaan maar als een vector. In dit geval moet naast de niet-nulwaarde ook de bijbehorende indices i en j opgeslagen worden. Daarmee wordt het geheugenbezetting sterk verminderd (de opslagruimte daalt van $O(n^2)$ naar $O(n)$, wel met extra ruimte voor indices) en de berekeningen sneller omdat bijv. bij matrix-vector vermenigvuldigingen de 0 elementen overgeslagen kunnen worden. Om in `Matlab` met sparse matrices te werken moeten deze als `sparse` gedeclareerd worden. De *sparsity pattern* (ijlheidspatroon) is de verzameling van locatie van niet-nul elementen in een sparse matrix. In `Matlab` is deze te visualiseren met het commando `spy`.
Ter info: In paragraaf 3.9, 3.9.1 worden sparse matrices besproken. Er wordt ook een algortime gepresenteerd dat gebruikt kan worden om de sparsity pattern te verminderen.

60. **Banded matrices** (zie ook paragraaf 3.7, 3.7.1)
Een speciale categorie van sparse matrices zijn de *bandmatrices* waarbij nietnul elementen op slechts een klein aantal diagonalen (lijnen in de matrix waarop het verschil van de rij-index en de kolom-index constant is) voorkomen. Enkele voorbeelden zijn de diagonaalmatrix ($A \in \mathbb{R}^{n\times n}$ met $A_{ij} = 0$ als $i \neq j$) of de tridiagonaalmatrix (vb. matrix A bij punt 34). Een matrix heeft de *bandtype* (k_1, k_2) met $k_{1,2} \in \mathbb{N}$ als $A_{ij} = 0$ voor alle $j < i - k_1$ en $j > i + k_2$. De *bandbreedte* van een matrix met bandtype (k_1, k_2) is het getal $k_1 + k_2 + 1$. Voor (k_1, k_2) bandmatrices geldt
Stelling: Zij $A \in \mathbb{R}^{n\times n}$ een inverteerbare bandmatrix met bandtype (k_1, k_2) waarvoor de Gauß eliminatie zonder pivotering mogelijk is. Dan zijn de matrices L en U bij de LU-decompositie tevens bandmatrices met bandtype $(k_1, 0)$ resp. $(0, k_2)$.

61. Los het stelsel $AX = b$ op met A de matrix in punt 34 en b de vector in punt 52. Neem $n = 10^4$ en gebruik de backslash operator, i.e. $X = A\backslash b$. Definieer A zowel als een ijle matrix (gedefineerd m.b.v. het bevel `spdiags`) als ook als gewone matrix (voeg toe het bevel `A=full(A);` na `A=spbdiags...`). Bestudeer het verschil in geheugengebruik (`whos`) en de tijdsduur om het probleem op te lossen (`tic,toc`).

```
tic
n=10^4; e=ones(n,1);
A=spdiags([-e 2*e -e],-1:1,n,n);
f=@(x)x.*(1-x);
x=1/(n+1)*[1:n]';
b=-(n+1)^2*f(x);
X=A\b;
toc
```

62. **Diagonaal gedomineerde matrices**
Een matrix $A \in \mathbb{R}^{n\times n}$, $A = (a_{ij})_{i,j=\overline{1,n}}$ is *diagonaal gedomineerd* als

$$|a_{ii}| \geq \sum_{j=1, j\neq i}^{n} |a_{ij}| \quad \text{voor alle } i = 1, \ldots, n.$$

Zulke matrices komen vaak voor bij de numerieke oplossing van differentiaalvergelijkingen. Voor zulke matrices geldt
Stelling: Zij $A \in R^{n\times n}$ diagonaal gedomineerd en inverteerbaar. Dan bestaat er de LU-decompositie van A, $A = LU$ met L-benedendriehoekig en U-bovendriehoekig, die middels GEM zonder pivotering bepaald kan worden.
In de volgende opdrachten schetsen we het bewijs-idee, dat gebaseerd is op het volgende: de standaard GEM is het resultaat van een reeks matrix bewerkingen, $LU = G^{(n-1)}P^{(n-1)} \ldots G^{(1)}P^{(1)}A$. Met de conventie $A^{(1)} = A$ tonen we aan dat als het resultaat na k stappen $A^{(k+1)} = G^{(k)}P^{(k)} \ldots G^{(1)}P^{(1)}A$ diagonaal gedomineerd is ($k > 0$) dan is geen pivotering nodig bij stap $k + 2$, m.a.w. $P^{(k+2)} = I$.

63. Toon aan dat omdat $A^{(1)} = A$ diagonaal gedomineerd en inverteerbaar is dan geldt $a_{11}^{(1)} \neq 0$. Daarom kan $a_{11}^{(1)}$ als eerste spil gebruikt worden.

64. Zij $m_{1,k} = \frac{a_{k1}^{(1)}}{a_{11}^{(1)}}$ en $A^{(2)} = G^{(1)}A$ met $G^{(1)} = T_{1,2}(m_{1,2})T_{1,3}(m_{1,3}) \cdots T_{1,n}(m_{1,n})$. Er geldt voor $k = 2, \ldots, n$ en $j = 1, \ldots, n$: $a_{kj}^{(2)} = a_{kj}^{(1)} - m_{1,k}a_{1j}^{(1)}$. Toon aan dat er geldt

$$|a_{kk}^{(2)}| \geq \sum_{j=1, j\neq k}^{n} |a_{kj}^{(2)}|.$$

65. Leg uit waarom $A^{(2)}$ inverteerbaar is.

66. De inductie redenering kan vervolledigd worden door dezelfde redenering toe te passen om te bewijzen dat als $A^{(k)}$ diagonaal gedomineerd en inverteerbaar is dan geldt $a_{k+1k+1}^{(k+1)} \neq 0$ en vervolgens is $A^{(k+1)}$ diagonaal gedomineerd en inverteerbaar.

Les 5

Interpolatie

Korte inhoud

Zij f een functie die of een ingewikkelde vorm heeft die praktisch niet te gebruiken is m.u.v. de waarde voor particuliere argumenten x_i, $i = 0, \ldots, n$, of simpelweg niet bekend is m.u.w. de waarden in deze argumenten. Daarom worden er veronderstelt dat er slechts de punten $\{(x_i, y_i), i = 0, \ldots, n\}$ beschikbaar zijn met $y_i = f(x_i)$.
Interpolatie is een manier om de waarde van f in gegeven punten te benaderen, gebaseerd op de beschikbare informatie. Het idee is om de functie f te benaderen met een andere functie die een veel makkelijkere vorm heeft en waarvoor de praktische implementatie in een code simpel verloopt. M.a.w., er wordt gezocht naar een simpele functie g die voldoet aan $g(x_i) = y_i$ voor alle i. De functie g kan bijv. een van de volgende types hebben:

$$g(x) = a_n x^n + \cdots + a_0, \ (n^{de} \text{ graads veelterm})$$

$$g(x) = \tfrac{1}{2}a_0 + \textstyle\sum_{k=0}^{n} \left[a_k \cos(kx) + b_k \sin(kx)\right] \ \text{(trigonometrische veelterm)}$$

In sommige gevallen zijn niet alleen de waarden $y_i = f(x_i)$ bekend maar ook de afgeleiden tot een zekere orde, nl. $y_i^{(k)} = f^{(k)}(x_i)$, $i = 0, \ldots, n$, $k = 0, \ldots, \mu_i$ met $\mu_i \geq 0$. De vraag blijft onveranderd: bepaal een simpele functie g die voldoet aan $g^{(k)}(x_i) = y_i^{(k)}$, $i = 0, \ldots, n$, $k = 0, \ldots, \mu_i$.

67. Lees blz. 333 (zie ook paragraaf 1-2 in het boek van A. Bultheel) voor een inleiding.

Veeltermінterpolatie

68. Veeltermen hebben een simpele vorm en zijn oneindig vaak differentieerbaar. Daarom zijn ze ook uitstekende kandidaten voor de benadering van functies. Lees blz. 334-335 (zie ook hoofdstuk 3 en 4 in het boek van A. Bultheel) waarin het interpolatieprobleem en de interpolerende veelterm worden geïntroduceerd. Merk op: omdat voor de benadering alleen de waarden van f en niet haar afgeleiden worden gebruikt wordt de *Lagrange interpolatie* genoemd. Uitbreidingen waarin ook afgeleiden gebruikt worden zullen later besproken worden.

69. Bestudeer paragraaf 8.1 over de *interpolerende veelterm volgens Lagrange*. Onthoud de expliciete vorm van de *Lagrange veeltermen*.

70. Maak Exercise 3 op blz. 376.

71. Maak Exercise 2 op blz. 375. Geef een bewijs voor Stelling 8.1 dat gebaseerd is op een resultaat uit de lineaire algebra (de Vandermonde determinant).

72. Zij $\{x_i, i = 0, \ldots, n\}$ een stel punten in $\mathbb{R}$ die onderling verschillend zijn en $\ell_i \in \mathbb{P}_n$ $(i = 0, \ldots, n)$ de bijbehorende Lagrange veeltermen. Geef een bewijs voor de volgende beweringen:

    (a) $\{\ell_i, i = 0, \ldots, n\}$ is een stel lineair onafhankelijke veeltermen en vormen dus een basis voor $\mathbb{P}_n$.

    (b) Er geldt:

        i. $\sum_{i=0}^{n} \ell_i(x) = 1$ voor alle $x \in \mathbb{R}$;
        ii. $\sum_{i=0}^{n} x_i^k \ell_i(0) = 0$ voor alle $k = 1, \ldots, n$.
        iii. $\sum_{i=0}^{n} x_i^{n+1} \ell_i(0) = (-1)^n x_0 \cdot \cdots \cdot x_n$.

    Hint: Gebruik de eenduidigheid van de interpolerende veelterm.

73. Zij $(x_k, y_k) \in \mathbb{R}^2$, $(k = 0, \ldots, n)$ gegeven punten met onderling verschillend abscissen x_k. Toon aan dat de Lagrange interpolerende veelterm $p : \mathbb{R} \to \mathbb{R}$, $p \in \mathbb{P}_n$ gedefineerd kan worden als de determinant

$$det \begin{pmatrix} 1 & x & x^2 & \ldots & p(x) \\ 1 & x_0 & x_0^2 & \ldots & f_0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 & \ldots & f_n \end{pmatrix} = 0.$$

74. Bestudeer paragraaf 8.1.1 (zie ook hoofdstuk 6 in het boek van A. Bultheel) over de interpolatiefout. In het bijzonder bestudeer de stelling 8.2 en haar bewijs.

75. Geef drie benaderingen voor $\sqrt{19}$ gebaseerd op de interpolerende veeltermen van orde $0, 1$, resp. 2 en schat de bijbehorende fouten af. Je mag de punten handig kiezen zodat de berekeningen simpel zijn.

76. Voor gegeven $n \in \mathbb{N}$ en de abscissen $x_i = \frac{i}{n}$, $i = 0, \ldots n$ beschouw de interpolerende veelterm $\Pi_n \in \mathbb{P}_n$ voor de functie $\sin(x)$. Zij $x \in (0, 1)$ en $\{x_i, i = 1, \ldots, n\}$ willekeurig. Voor welke n voldoet de benadering $\Pi_n(x)$ van $\sin(x)$ aan $|\Pi_n(x) - \sin(x)| \leq 10^{-5}$.

77. Schrijf een MATLAB programma dat als input data de orde n, het argument x en de punten $(x_i, y_i), i = 0, \ldots, n$ krijgt en de waarde $\Pi_n(x)$ van de interpolerende veelterm teruggeeft. Test dit programma op de twee bovenstaande voorbeelden.

78. Bepaal het aantal bewerkingen voor het algoritme bij punt 77 (de complexiteit).

Les 6

79. Bestudeer paragraaf 8.2 (zie ook paragraaf 9.1–9.6 in het boek van A. Bultheel) over *interpolerende veelterm volgens Newton* en over *gedeelde differenties*. Merk op, de Lagrange resp. Newton vormen zijn twee verschillende manieren om dezelfde interpolerende veeltermen te schrijven! Het gaat dus niet om twee verschillende veeltermen.

80. *Interpolerende veelterm volgens Newton*
Hieronder is $\mathbb{P}_n = \{p : \mathbb{R} \to \mathbb{R},$ er bestaan $a_0, \ldots, a_n \in \mathbb{R}$ z.d. $p(x) = a_n x^n + \cdots + a_1 x + a_0\}$ de verzameling van n-de graads of lager veeltermen. We leiden de Newton vorm af voor de interpolerende veelterm.

Definitie: *Gedeelde differenties van orde i*
Zij $f : \mathbb{R} \to \mathbb{R}$ continu en $a \leq x_0 < x_1 < \cdots < x_n \leq b$ gegeven abscissen. De gedeelde differenties worden recursief gedefinieerd als:

Orde 0: $f[x_i] := f(x_i)$, met $i \in \{0, \ldots, n\}$

Orde 1: $f[x_i, x_{i+1}] := \frac{f[x_{i+1}] - f[x_i]}{x_{i+1} - x_i}$, met $i \in \{0, \ldots, n-1\}$

$\ldots$

Orde k: $f[x_i, \ldots, x_{i+k}] := \frac{f[x_{i+1}, \ldots, x_{i+k}] - f[x_i, \ldots, x_{i+k-1}]}{x_{i+k} - x_i}$, met $k \leq n$ en $i \in \{0, \ldots, n-k\}$.

We tonen aan dat

$$\begin{aligned} \Pi_n(x) &= f[x_0] + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1) \\ &\quad + \cdots + f[x_0, \ldots, x_n](x - x_0) \ldots (x - x_{n-1}). \end{aligned}$$

Dit is geïnspireerd door de volgende opmerking: $\Pi_n(x_i) - \Pi_{n-1}(x_i) = 0$ voor elke $i \in \{0, \ldots, n-1\}$ (waarom?). Omdat $\Pi_n - \Pi_{n-1}$ en n-de graad veelterm is dan bestaat er een $a \in \mathbb{R}$ z.d.

$$\Pi_n(x) - \Pi_{n-1}(x) = a(x - x_0) \ldots (x - x_{n-1}).$$

We tonen hieronder aan dat $a = f[x_0, \ldots, x_n]$. We beginnen met een notatie: $\Pi_{i,k} \in \mathbb{P}_k$ is de interpolerende veelterm voor f in de punten $(x_j, f(x_j))$, $j = i, \ldots, i+k$ met $i \in \{0, \ldots, n\}$ en $k \in \{0, \ldots, n-i\}$. Merk op dat $\Pi_k = \Pi_{0,k}$. We bewijzen nu
Stelling. Er geldt

$$\Pi_n(x) = \Pi_{n-1}(x) + f[x_0, \ldots, x_n](x - x_0) \ldots (x - x_{n-1}).$$

Bewijs. We geven een bewijs door inductie: voor alle $k \in \{1, \ldots, n\}$ geldt

$$\Pi_{i,k}(x) = f[x_i, \ldots, x_{i+k}](x - x_i) \ldots (x - x_{i+k-1}) + q, \tag{7}$$

met $q \in \mathbb{P}_{k-1}$. Merk op dat de eerste term in (7) is een k-de graads veelterm terwijl q een graad minder heeft. Er geldt dus $\Pi_{i,k}(x) = f[x_i, \ldots, x_{i+k}]x^k + \tilde{q}$ met $\tilde{q} \in \mathbb{P}_{k-1}$ en daarom is de coëfficiënt van x^k exact $f[x_i, \ldots, x_{i+k}]$.

De uitspraak is waar voor $k = 1$ omdat

$$\Pi_{i,1}(x) = f(x_0) + \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x - x_0) = f[x_0] + f[x_0, x_1](x - x_0)$$

en $f[x_0] \in \mathbb{R}$ is tevens een 0-de graads veelterm.

Zij $k \geq 2$. We nemen aan dat de uitspraak waar is voor $k - 1$, dus voor alle $i \in \{0, \ldots, n - k + 1\}$ geldt

$$\Pi_{i,k-1}(x) = f[x_i, \ldots, x_{i+k-1}](x - x_i) \ldots (x - x_{i+k-2}) + q_i, \tag{8}$$

met $q_i \in \mathbb{P}_{k-2}$. We tonen eerst aan dat de k-de graads veelterm $p \in \mathbb{P}_n$

$$p(x) = \frac{(x - x_i)\Pi_{i+1,i+k}(x) - (x - x_{i+k})\Pi_{i,i+k-1}(x)}{x_{i+k} - x_i}$$

interpolerend is voor f in $(x_{i+j}, f(x_{i+j}))$, $j = \{i, i + 1, \ldots, i + k\}$. Omdat de interpolerende veelterm uniek is geldt ook $\Pi_{i,k} = p$.

Ten eerste voor elke $j \in \{1, \ldots, k - 1\}$ geldt $\Pi_{i+1,i+k}(x_{i+j}) = \Pi_{i,i+k-1}(x_{i+j}) = f(x_{i+j})$. Daarom geldt ook dat $p(x_{i+j}) = f(x_{i+j})$. Vervolgens,

$$p(x_i) = \Pi_{i,i+k-1}(x_i) = f(x_i) \text{ en } p(x_{i+k}) = \Pi_{i+1,i+k}(x_{i+k}) = f(x_{i+k}).$$

Daarom is p interpolerend.

Omdat $\Pi_{i,i+k-1}, \Pi_{i+1,i+k} \in \mathbb{P}_{k-1}$, uit (7) en (8) volgt dat

$$\begin{aligned}
\Pi_{i,k}(x) &= p(x) = \tfrac{(x-x_i)\Pi_{i+1,i+k}(x)-(x-x_{i+k})\Pi_{i,i+k-1}(x)}{x_{i+k}-x_i} \\
&= \tfrac{x-x_i}{x_{i+k}-x_i}\Pi_{i+1,i+k}(x) - \tfrac{x-x_{i+k}}{x_{i+k}-x_i}\Pi_{i,i+k-1}(x) \\
&= \tfrac{x-x_i}{x_{i+k}-x_i}\big(f[x_{i+1}, \ldots, x_{i+k}](x - x_{i+1}) \ldots (x - x_{i+k-1}) + q_{i+1}\big) \\
&\quad -\tfrac{x-x_{i+k}}{x_{i+k}-x_i}\big(f[x_i, \ldots, x_{i+k-1}](x - x_i) \ldots (x - x_{i+k-2}) + q_i\big),
\end{aligned}$$

voor zekere $q_i, q_{i+1} \in \mathbb{P}_{k-2}$. Hieruit volgt dat de coëfficiënt a van x^k in $\Pi_{i,k}(x)$ is

$$a = \frac{1}{x_{i+k} - x_i}\big(f[x_{i+1}, \ldots, x_{i+k}] - f[x_i, \ldots, x_{i+k-1}]\big).$$

Volgens de definitie van de gedeelde differentie is $a = f[x_i, \ldots, x_{i+k}]$. Dit is precies de coëfficiënt van x^k in (7) en hiermee is het bewijs volledig.

81. Zoals gezien is de gedeelde differentie $f[x_0, x_1, ..., x_n]$ onafhankelijk van de volgorde van de argumenten x_k, $k = 0, \ldots, n$. Dit volgt onmiddelijk uit de gelijkheid

$$f[x_0, x_1, ..., x_n] = \sum_{k=0}^{n} \frac{f(x_k)}{\omega'_{n+1}(x_k)}.$$

Geef een bewijs voor deze gelijkheid (je kunt hiervoor de inductie gebruiken).

82. (Supplementair!) Geef een bewijs voor de onderstaande *middelwaarde stelling voor gedeelde differenties*. Wat is het gevolg voor $p[x_0, \ldots, x_{n+1}]$ als $p \in \mathbb{P}_n$?
Zij $f : [a, b] \to \mathbb{R}$ en functie die $(n+1)$ keer continu differentieerbaar is en de punten $a \leq x_0 < x_1 < ... < x_{n+1} \leq b$. Dan bestaat er een $\zeta \in (a, b)$ z.d.

$$f[x_0, x_1, ..., x_{n+1}] = \frac{1}{(n+1)!} f^{(n+1)}(\zeta).$$

Hint: Zie ook paragraaf 8.2.2.

83. Zij $0 < x_0 < x_1 < \cdots < x_n$. Bepaal de gedeelde differentie $f[x_0, \ldots, x_n]$ voor de functie $f(x) = \frac{1}{x}$.
Hint: Probeer de inductie.

84. Schrijf een MATLAB programma dat als input data de orde n, het argument x en de punten $(x_i, y_i), i = 0, \ldots, n$ krijgt en de waarde $y_n(x)$ van de interpolerende veelterm teruggeeft. Gebruik nu de interpolerende veelterm volgens Newton. Test dit programma op de twee voorbeelden bij punt 77. Tel in dit geval ook op het aantal bewerkingen en vergelijk het resultaat met dat voor de interpolerende veelterm volgens Lagrange.

85. Zij $a, b, \lambda \in \mathbb{R}$ gegeven z.d. $a < b$ en $f : [a, b] \to \mathbb{R}$, $f(x) = e^{\lambda x}$. Zij $\Pi_n \in \mathbb{P}_n$ de veelterm die f interpoleert in de punten $x_i, i = 0, \bar{} n$ waarbij de punten onderling verschillend zijn en in het interval $[a, b]$ willekeurig gekozen zijn. Toon aan dat er geldt

$$\lim_{n \to \infty} \max\{|f(x) - \Pi_n(x)|, x \in [a, b]\} = 0,$$

m.a.w. dat Π_n uniform in $[a, b]$ naar f convergeert als $n \to \infty$. Leg uit waarom dit resultaat wel geldt voor de gegeven functie en niet voor de functie $g(x) = \frac{1}{1+x^2}$ zoals besproken in de les.

Veelterminterpolatie: Hermite

86. Bestudeer paragraaf 8.5 over de *Hermite interpolatie*. In de algemene vorm kan het *Hermite interpolatie probleem (HIP)* zoals volgt geformuleerd worden:
Voor gegeven x_i, $i = 0, \ldots, n$ en $y_i^{(k)}$, $i = 0, \ldots, n$, $k = 0, \ldots, \mu_i$ $(\mu_i \geq 0)$ en met $m = \mu_0 + \cdots + \mu_n + n$ bepaal een veelterm $H_m \in \mathbb{P}_m$ die voldoet aan $H_m^{(k)}(x_i) = y_i^{(k)}$, $i = 0, \ldots, n$, $k = 0, \ldots, \mu_i$.
De waarden $y_i^{(k)}$ kunnen bijv. de afgeleiden van een functie f zijn, $y_i^{(k)} = f^{(k)}(x_i)$, $i = 0, \ldots, n$, $k = 0, \ldots, \mu_i$. De veelterm H_m heet *Hermite interpolerende veelterm* voor de punten x_i met multipliciteiten $\mu_i + 1$. Zoals voor de Lagrange veelterm kan men bewijzen dat het HIP een eenduidige oplossing heeft.

87. Voor de interpolatiefout geldt een resultaat dat gelijkaardig is aan het resultaat voor de Lagrange interpolatie. Geef een bewijs voor de volgende
Stelling. Zij $x_i \in [a,b]$, $i = 0,\ldots,n$ onderling verschillend, $\mu_i \in \mathbb{N}$ ($\mu_i \geq 0$), $m = \mu_0 + \cdots + \mu_n + n$ en $f : [a,b] \to \mathbb{R}$ $m+1$ continu differentieerbaar. Zij $H_m \in \mathbb{P}_m$ de Hermite interpolerende veelterm die voldoet aan $H_m^{(k)}(x_i) = f^{(k)}(x_i)$, $i = 0,\ldots,n$ en $k = 0,\ldots,\mu_i$ en $\Omega_m(x) = (x-x_0)^{\mu_0+1} \cdot \cdots \cdot (x-x_n)^{\mu_n+1}$. Dan bestaat er voor elke $x \in [a,b]$ een $\zeta_x \in (a,b)$ waarvoor geldt

$$f(x) - H_m(x) = \frac{\Omega_m(x)}{(m+1)!} f^{(m+1)}(\zeta_x).$$

88. Bepaal de Hermite veelterm die de functie sin : $[0, 4\pi] \to \mathbb{R}$ en haar eerste afgeleiden interpoleert in de punten $x_0 = 0$, $x_1 = 4\pi/3$, $x_2 = 8\pi/3$ en $x_3 = 4\pi$. Schrijf een MATLAB programma dat de interpolerende veelterm en de functie grafisch weergeeft. Vervolgens gebruik het programma bij 77 om de Lagrange interpolerende veelterm weer te geven voor dezelfde functie en in de punten x_i zoals boven. Wat valt op? Probeer nu de Lagrange interpolerende veelterm met 7 punten, $\bar{x}_k = 4\pi k/6$, $k = 0,\ldots 6$.

Les 7

Spline interpolatie

89. **Stuksgewijze interpolatie** (zie ook paragraaf 8.4).
Zoals reeds gezien hangt de interpolatiefout van de orde van de veelterm en van de distributie van de punten in het interval. Voor zowel de Lagrange als ook de Hermite interpolerende veelterm geldt voor elke $x \in [a,b]$

$$|f(x) - p_n(x)| \leq \frac{|b-a|^{n+1}}{(n+1)!}\|f^{(n+1)}\|_\infty \quad \text{met } \|f^{(n+1)}\|_\infty = \sup_{z\in[a,b]} |f^{(n+1)}(z)|.$$

Een nadeel is dat de verhoging van de orde (of van het aantal interpolatiepunten) niet per se tot een daling van de interpolatiefout leidt. De hogere orde afgeleiden kunnen sneller groeien dan n. Erboven op is de kans op een oscillerend gedrag groter bij hogere orde veeltermen dan bij lagere orde. Een manier om de interpolatiefout in het punt x te verminderen is het interval waarin de interpolatiepunten x_i worden gekozen zo klein mogelijk rond x te kiezen. Daardoor zijn de factoren $x - x_i$ klein, wat tot een vermindering van de interpolatiefout leidt.

Anderzijds is in sommige gevallen het interval $I = [a,b]$ vast en men wil een benadering vinden voor de waarde van f in alle punten in dat interval. een manier om de boven genoemde oplossing toe te passen is het interval I te delen in kleinere deelintervallen $I_j = [x_{j-1}, x_j]$, $j = 1, \ldots, n$ en interpolerende veeltermen $p_k^j \in \mathbb{P}_k$ in elk van deze intervallen te vinden (gebruik makend van extra interpolatiepunten in elk interval I_j, of als er ook afgeleiden geïnterpoleerd worden). Voor de eenvoud hebben hier alle veeltermen dezelfde orde maar dat hoeft niet. Als er een $h > 0$ bestaat z.d. $0 < x_j - x_{j-1} \leq h$ voor alle j dan geldt voor elke $x \in I_j$ en dus overal in $[a,b]$

$$|f(x) - p_k^j(x)| \leq \frac{h^{k+1}}{(k+1)!}\|f^{(k+1)}\|_\infty.$$

Het voordeel is dat voor h klein genoeg is ook de interpolatie fout gegarandeerd onder een gewenste drempel en daarvoor hoeft men de orde van de veelterm niet te verhogen. Het nadeel is dat zulke benaderingen stuksgewijs glad zijn. In het inwendige van elk interval I_j is de benadering een veelterm. Aangenomen dat de punten x_j zelf ook interpolatie punten zijn in beide intervallen I_j en I_{j+1} dan zijn in de punten x_j slechts de linker- en rechterlimieten gelijk en de benaderende functie is wel continu maar hoeft niet eens differentieerbaar te zijn. Om een C^1 benadering te verkrijgen kan een Hermite interpolatie beschouwd worden. Als in de punten x_j zowel $f(x_j)$ als ook $f'(x_j)$ geïnterpoleerd worden dan zijn de linker- een rechterlimieten gelijk voor zowel f als ook f' en het resultaat is ook C^1. Hetzelfde idee kan gebruikt worden om hogere orde differentieerbaarheid te verkrijgen.

De benadering die we nu hebben besproken is een functie $p : [a,b] \to \mathbb{R}$ in de

functieruimte

$$S_h^{k,r}[a,b] = \{p \in C^r([a,b])/p|_{I_j} \in \mathbb{P}_k \text{ voor alle } j = 1,\ldots,n\},$$

waarbij $p|_{I_j}$ de restrictie van p op het interval I_j betekent. Voor de functieruimte in paragraaf 8.4 geldt $X_h^k = S_h^{k,0}[a,b]$ omdat de functies daar niet differentieerbaar hoeven te zijn.

90. Beschouw alweer de situatie bij punt 88 en bepaal de 2^e orde Lagrange interpolerende veeltermen in de intervallen $[0, 4\pi/3]$, $[4\pi/3, 8\pi/3]$, $[8\pi/3, 4\pi]$ met de tussenpunten $2\pi/3$, 2π, $10\pi/3$. Vergelijk het resultaat in dit geval met de resultaten bij 77 en 88.

91. **Splines** (zie ook paragraaf 8.7 tot paragraaf 8.7.1)
Het nadeel van de stuksgewijze benadering van boven is dat een gladdere benadering alleen mogelijk is als men voldoende informatie heeft over f. Om preciezer te zijn, als de benadering C^r moet zijn dan moeten ook alle afgeleiden $f^{(s)}$, $s = 0,\ldots,r$ bekend zijn in alle punten x_j. Deze afgeleiden zijn niet altijd beschikbaar. Een andere mogelijkheid is om de interpolatie-conditie voor afgeleiden te vervangen door de continuïteit C^r. Zij de punten x_j zo gekozen dat $x_0 = a$ en $x_n = b$. Dan geldt voor de C^r interpolatie

$$p^{(s)}(x_j - 0) = p^{(s)}(x_j + 0), \qquad \text{voor alle } s = 0,\ldots,r \text{ en } j = 1,\ldots,n-1$$

($p(x \mp 0)$ staat voor de linker-/rechterlimiet van p in het punt x). Zulke interpolerende functies, die stuksgewijze veeltermen zijn en maximale gladheid hebben heten *splines*. Een definitie in de volledige algemeenheid gaat verder dan de doelen van een inleidend vak.

92. **Stuksgewijze affiene splines**
Denkend aan de functieruimte in punt 89, de simpelste variant is $S_h^{1,0}[a,b]$, de stuksgewijze affiene (1^e orde veelterm) functies. Bij gegeven (x_j, y_j), $j = 0,\ldots,n$ en met $y_j = f(x_j)$ voor een functie $f \in C^2([a,b])$, de interpolerende spline is de functie $p : [a,b] \to \mathbb{R}$ waarvoor geldt $p(x_j) = y_j$ voor alle j en er bestaan $a_j, b_j \in \mathbb{R}$ z.d. $p(x) = a_j + b_j x$ voor alle $x \in [x_{j-1}, x_j]$, $j = 1,\ldots,n$.

93. *Voorbeeld*: Zij (x_0, y_0) en (x_1, y_1) twee punten in $\mathbb{R}^2$ gegeven. Bepaal p, de bijbehorende affiene interpolerende spline $p \in S^{1,0}[a,b]$ met $a = x_0$ en $b = x_1$. Schrijf duidelijk op aan welke condities voldoet deze spline. Alternatief, beschouw eerst de punten $(x_0, 1)$ en $(x_1, 0)$ en vervolgens $(x_0, 0)$ en $(x_1, 1)$. Bepaal in beide gevallen φ_0 en φ_1, de interpolerende splines. Druk p uit als lineaire combinatie van φ_0 en φ_1.

94. Toon de existentie en eenduidigheid aan van de stuksgewijze affiene interpolerende spline.

95. Net zoals de interpolerende veelterm volgens Lagrange kan de $S_h^{1,0}$ spline geschreven worden in termen van basisfuncties φ_j:

$$p(x) = \sum_{j=0}^{n} f(x_j)\varphi_j(x),$$

zie ook het voorbeeld bij punt 93. Bepaal de basisfuncties φ_j (merk op, deze functies zijn stuksgewijs affien, $\varphi_j \in S_h^{1,0}$).

96. Zij $h = \max_{j=1,\ldots,n} x_j - x_{j-1}$. Geef een bovengrens voor de spline interpolatiefout $f(x) - p(x)$.

97. **Kubische splines** (zie ook paragraaf 8.7.1)
De functies bij punt 92 zijn continu maar niet differentieerbaar. Veel gebruikt zijn functies in $S_h^{3,2}[a,b]$, nl. interpolerende splines die 2 keer continu differentieerbaar zijn en hun restrictie op elk interval I_j een 3^e orde veelterm is. Zij (x_j, y_j), $j = 0,\ldots,n$ en met $y_j = f(x_j)$ gegeven met functie $f \in C^4([a,b])$. De *interpolerende kubische spline* is een functie $p : [a,b] \to \mathbb{R}$ waarvoor geldt $p(x_j) = y_j$ voor alle j en voldoet aan

$$p'(x_j - 0) = p'(x_j + 0) \text{ en } p"(x_j - 0) = p"(x_j + 0) \quad \text{voor } j = 1,\ldots,n-1.$$

Omdat p een 3^e orde veelterm is op elk deelinterval I_j geldt voor elke $j = 1,\ldots,n$ dat

$$p(x) = a_j + b_j(x - x_j) + c_j(x - x_j)^2 + d_j(x - x_j)^3, \text{ voor alle } x \in I_j = [x_{j-1}, x_j].$$

Om de functie p te verkrijken moeten de coëficciënten a_j, b_j, c_j en $d_j \in \mathbb{R}$ bepaald worden. De interpolatie conditie geeft

$$\begin{array}{rcll} p(x_j + 0) & = & f(x_j), & \text{voor } j = 0,\ldots,n-1, \\ p(x_j - 0) & = & f(x_j), & \text{voor } j = 1,\ldots,n, \\ p'(x_j - 0) & = & p'(x_j + 0), & \text{voor } j = 1,\ldots,n-1, \\ p"(x_j - 0) & = & p"(x_j + 0), & \text{voor } j = 1,\ldots,n-1. \end{array} \tag{9}$$

Merk op dat (9) een lineair stelsel is met $4n-2$ vergelijkingen en $4n$ onbekenden, de coëfficciënten. Om de spline eenduidig te kunnen bepalen worden er de twee zgn. *natuurlijke condities* toegevoegd, nl.

$$p"(x_0 + 0) = 0 \quad \text{en} \quad p"(x_n - 0) = 0. \tag{10}$$

Zoals later uitgelegd leidt dit tot een functie die de 2^e afgeleide minimaliseert. Een spline interpolerende functie die voldoet aan 10 heet *natuurlijke spline*.

Merk op dat (9)–(10) en lineair stelsel is met $4n$ vergelijkingen en onbekenden.

98. *Voorbeeld*: Zij (x_0, y_0) en (x_1, y_1) twee punten in $\mathbb{R}^2$ gegeven. Bepaal p, de natuurlijke interpolerende spline in $S_h^{3,2}[a,b]$ (met $x_0 = a$ en $x_1 = b$, zie ook punt 93). Schrijf eerst duidelijk op aan welke condities deze spline voldoet.

99. Het bewijs voor de existentie en eenduidigheid van de kubische interpolerende spline die ook aan de conditie in (10) voldoet kan gereduceerd worden tot de analyse van het stelsel (9)–(10). Dit kan geschreven worden als $\mathcal{A}\mathcal{Y} = \mathcal{F}$ met
$\mathcal{A} \in \mathbb{R}^{4n\times 4n}$ (de elementen laten we nu buiten beschouwing),
$\mathcal{F} \in \mathbb{R}^{4n}$, $\mathcal{F} = (f(x_0), \ldots, f(x_{n-1}), f(x_1), \ldots, f(x_n), 0, \ldots, 0)^T$
en de onbekenden $\mathcal{Y} \in \mathbb{R}^{4n}$, $\mathcal{Y} = (a_1, b_1, c_1, d_1, \ldots, a_n, b_n, c_n, d_n)^T$.

De existentie en eenduidigheid van de functie p is dus equivalent met de existentie en eenduidigheid van een oplossing voor het stelsel $\mathcal{A}\mathcal{Y} = \mathcal{F}$. Omdat $\mathcal{A}$ een vierkante matrix is is het bewijs voor eenduidigheid genoeg. Dit volgt uit

Stelling (*eenduidigheid*). Er bestaat ten hoogste één functie $p \in S_h^{(3,2)}[a,b]$ die voldoet aan de condities (9)–(10).
Bewijs: Zij $p_1, p_2 \in S_h^{3,2}[a,b]$ twee natuurlijke splines die de data (x_j, y_j), $j = 0, \ldots, n$ interpoleren. Dan voldoet $p = p_1 - p_2 \in S_h^{3,2}[a,b]$ aan (10) en er geldt $p(x_j) = 0$ voor $j = 0, \ldots, n$.

Vervolgens tonen we aan dat $p" \equiv 0$ (overal in $[a,b]$). Daarom is p ook stuksgewijs affien in $[a,b]$, dus affien in elk interval I_j. Uit $p(x_{j-1}) = p(x_j) = 0$ volgt dan $p \equiv 0$ op I_j voor alle j en dus $p_1 \equiv p_2$. Om het bewijs voor $p"$ te verkrijgen bestuderen we het kwadraat van de $L^2[a,b]$ norm:

$$\begin{aligned}
\|p"\|^2 &= \int_a^b \left[p"(x)\right]^2 dx = \sum_{j=1}^n \int_{x_{j-1}}^{x_j} \left[p"(x)\right]^2 dx \\
&= \sum_{j=1}^n p"(x)p'(x)|_{x_{j-1}}^{x_j} - \sum_{j=1}^n \int_{x_{j-1}}^{x_j} p'''(x)p'(x)dx \\
&= \sum_{j=1}^n p"(x)p'(x)|_{x_{j-1}}^{x_j} - \sum_{j=1}^n p'''(x)p(x)|_{x_{j-1}}^{x_j} + \sum_{j=1}^n \int_{x_{j-1}}^{x_j} p''''(x)p(x)dx
\end{aligned}$$

Omdat de restrictie van p op elk interval I_j een 3^e graads veelterm is geldt $p'''' \equiv 0$ in I_j en is de laatse som 0. De middelste som is tevens 0 omdat $p(x_j) = 0$ voor alle j. Verder, omdat $p \in C^2([a,b])$ geldt ook:

$$p'(x_j - 0) = p'(x_j + 0) \quad \text{en} \quad p"(x_j - 0) = p"(x_j + 0) \quad \text{voor alle } j = 1, \ldots, n-1.$$

Daarom geldt $\|p"\|^2 = p"(b)p'(b) - p"(a)p'(a)$ en de conclusie volgt uit (10).

100. Geef een bewijs voor het volgend resultaat
Propositie. Zij $p \in S_h^{(3,2)}[a,b]$ de interpolerende spline die voldoet aan de condities (9)–(10). Voor elke functie $w \in C^2[a,b]$ z.d. $w(x_j) = 0$ $(j = 0, \ldots, n)$ geldt

$$\int_a^b p"(x)w"(x)dx = 0.$$

101. **Minimalisering eigenschap**.
Stelling. Zij $p \in S_h^{3,2}[a,b]$ de natuurlijke interpolerende spline voor de data (x_j, y_j),

$j = 0, \ldots, n$ (met $y_j = f(x_j)$) en $g \in C^2[a,b]$ een willekeurige functie die voldoet aan $g(x_j) = y_j$, $j = 0, \ldots, n$. Toon aan dat er geldt

$$\|g"\| \geq \|p"\|.$$

Merk op: Dit resultaat is een minimalisatie eigenschap, nl. van alle interpolerende functies die ook $C^2[a,b]$ zijn hebben de natuurlijke interpolerende splines in $S_h^{3,2}[a,b]$ de minimale L^2 norm van de tweede afgeleide.

102. *Interpolatiefout.* Spline interpolerende functies en hun afgeleiden convergeren uniform op het interval $[a,b]$ zoals volgt uit de
Stelling. Zij $f \in C^4([a,b])$ en $p \in S_h^{3,2}[a,b]$ de natuurlijke interpolerende spline van f in de punten x_j, $j = 0, \ldots, n$. Zij $h = \max_{j=1,\ldots,n} x_j - x_{j-1}$. Dan bestaat er een $C > 0$ die niet van f afhangt z.d.

$$\|f^{(r)} - p^{(r)}\|_\infty \leq Ch^{4-r}\big[\sum_{k=0}^{4} \|f^{(k)}\|_\infty\big], \quad \text{voor} \quad r = 0, 1, 2, \text{ of } 3$$

(zonder bewijs).
Merk op dat het bovenstaande ook een uniforme convergentie resultaat is voor de spline interpolerende functie en haar afgeleiden. M.a.w. als $X_n := \{x_j^{(n)}, j = 0, \ldots, n\}$ een rij interpolatienoden is z.d. voor $h^{(n)} := \max_{j=1,\ldots,n} x_j^{(n)} - x_{j-1}^{(n)}$ geldt dat $\lim_{n\to\infty} h^{(n)} = 0$ dan convergeert ook $p_n^{(r)}$ (de bijbehorende interpolerende spline) uniform naar $f^{(r)}$. Voor de 3^e afgeleide moet nog gelden dat er een $\alpha \in (0,1]$ bestaat z.d. $\min_{j=1,\ldots,n} x_j^{(n)} - x_{j-1}^{(n)} \leq \alpha h^{(n)}$ uniform in n.

103. *Algoritme.* Zij $\{x_j, j = 0, \ldots, n\}$ gegeven. Om de bijbehorende natuurlijke interpolerende kubische spline p te bepalen beginnen we met de opmerking dat p een 3^e orde veelterm is op elk deelinterval I_j. Er geldt voor elke $j = 1, \ldots, n$

$$p(x) = a_j + b_j(x - x_j) + c_j(x - x_j)^2 + d_j(x - x_j)^3, \text{ voor alle } x \in I_j = [x_{j-1}, x_j].$$

M.a.w. moeten de coëfficiënten a_j, b_j, c_j en d_j bepaald worden. Deze volgen uit de condities (9) en (10), wat tot een lineair stelsel met $4n$ vergelijkingen en $4n$ onbekenden leidt. Zoals gezien bij punt 99 heeft dit stelsel een eenduidige oplossing. Het nadeel van deze aanpak is dat als n groter wordt dan zijn de rekentijd, het conditiegetal en vervolgens de rekenfout hoger. Dit kan voorkomen worden door het probleem op te splitsen in deelproblemen van orde n.

Merk eerst op dat uit $p(x_j - 0) = y_j$ volgt

$$a_j = y_j, \quad \text{voor alle } j = 1, \ldots, n. \tag{11}$$

Met de notatie $h_j = x_j - x_{j-1}$ en omdat $p(x_{j-1}+0) = y_{j-1}$ geldt voor alle $j = 1, \ldots, n$

$$-h_j b_j + h_j^2 c_j - h_j^3 d_j = y_{j-1} - y_j, \quad \text{voor alle } j = 1, \ldots, n. \tag{12}$$

De condities (10) leiden tot

$$2c_1 - 6h_1 d_1 = 0, \quad \text{en } 2c_n = 0. \tag{13}$$

De continuïteit van de 1^e afgeleide in x_j geeft

$$b_j = b_{j+1} - 2h_{j+1}c_{j+1} + 3h_{j+1}^2 d_{j+1}, \quad \text{voor alle } j = 1, \ldots, n-1 \tag{14}$$

en tenslotte uit de continuïteit van de 2^e afgeleide in x_j volgt

$$2c_j = 2c_{j+1} - 6h_{j+1}d_{j+1}, \quad \text{voor alle } j = 1, \ldots, n-1 \tag{15}$$

Voor de eenvoud definiëren we ook $c_0 := 0$. Uit (13) en (15) volgt

$$d_j = \frac{(c_j - c_{j-1})}{3h_j}, \quad \text{voor alle } j = 1, \ldots, n \tag{16}$$

Vervolgens uit (12) en (16) volgt

$$b_j = \frac{y_j - y_{j-1}}{h_j} + \frac{h_j}{3}(2c_j + c_{j-1}), \quad \text{voor alle } j = 1, \ldots, n \tag{17}$$

Tenslotte, (14), (16) en (17) leiden tot

$$h_j c_{j-1} + 2(h_j + h_{j+1})c_j + h_{j+1}c_{j+1} = 3\left[\frac{y_{j+1} - y_j}{h_{j+1}} - \frac{y_j - y_{j-1}}{h_j}\right], \tag{18}$$

voor alle $j = 1, \ldots, n-1$ (ga na!).

Met $c_0 = 0$ is (18) een stelsel $\mathcal{A}C = Y$ met $\mathcal{A} \in \mathbb{R}^{(n-1)\times(n-1)}$ een bandmatrix die tevens diagonaal gedomineerd is (zie ook punt 62). Daarom kan de GEM zonder pivotering toegepast worden en de matrices L en U in de LU decompositie zijn ook bandmatrices (zie punt 59).

104. *Voorbeeld:* We zullen de natuurlijke interpolerende kubische spline voor de functie $f(x) = \sqrt{x+1}$ met steunputen $x_0 = 0, x_1 = 3, x_2 = 8$ en $x_3 = 15$ bepalen aan de hand van het algoritme hierboven (punt 104). Het stelsel $\mathcal{A}C = Y$ in (18) wordt gegeven door

$$\begin{pmatrix} 2(h_1+h_2) & h_2 \\ h_2 & 2(h_2+h_3) \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} = 3 \begin{pmatrix} \frac{y_2-y_1}{h_2} - \frac{y_1-y_0}{h_1} \\ \frac{y_3-y_2}{h_3} - \frac{y_2-y_1}{h_2} \end{pmatrix}.$$

Met $h_1 = x_1 - x_0 = 3,\ h_2 = 5,\ h_3 = 7$, en $y_0 = \sqrt{x_0+1} = 1,\ y_1 = 2,\ y_2 = 3,\ y_3 = 4$ heeft het stelsel de oplossing $c_1 = -\frac{306}{12565}, c_2 = -\frac{26}{12565}$. Vergelijkingen (17) en (16) geven vervolgens b_j en d_j voor $j = 1, 2, 3$. De coëfficiënten a_j worden geven door (11). Hiermee verkrijgen we het polynoom

$$p(x) = \begin{cases} 2 + \frac{10729}{37695}(x-3) - \frac{306}{12565}(x-3)^2 - \frac{34}{12565}(x-3)^3, & \text{voor alle } x \in I_1, \\ 3 + \frac{5749}{37695}(x-8) - \frac{26}{12565}(x-8)^2 + \frac{8}{5385}(x-8)^3, & \text{voor alle } x \in I_2, \\ 4 + \frac{5203}{37695}(x-15) + \frac{26}{263865}(x-15)^3, & \text{voor alle } x \in I_3. \end{cases}$$

105. Schrijf een MATLAB programma om de kubische natuurlijke spline te bepalen die de functie $f(x) = 1/(1+x^2)$ interpoleert voor $x_j = -5 + 10 * j/n$, $j = 0, \ldots, n$. Probeer $n = 5$ en $n = 10$. Merk op, dit is het voorbeeld van Runge waarvoor de standaard veelterminterpolatie geen goede resultaten oplevert!

Multivariabele interpolatie

Het interpolatieprobleem kan veralgemeniseerd worden voor data in meerdere dimensies, nl. $(\bar{x}_k, f(\bar{x}_k)) \in \mathbb{R}^{d+1}$ $(d > 1)$ (zie ook paragraaf 8.6). Hier zijn $\bar{x}_k$ punten in $\mathbb{R}^d$ en $f : \mathbb{R}^d \to \mathbb{R}$ een gegeven functie. Voor de eenvoud blijven we in het geval $d = 2$, maar de ideën kunnen veralgemeniseerd worden tot hogere dimensies. We schrijven daarom $\bar{x}_k = (x_k, y_k) \in \mathbb{R}^2$.

106. *Cartesische rooster* (zie ook paragraaf 8.6.1)
Zij $\Omega = [a, b] \times [c, d] \subset \mathbb{R}^2$. In dit geval kan de twee dimensionale interpolerende functie gezien worden als het product van twee eendimensionale interpolerende functies. Bijv. voor gegeven $x_0 = a < x_1 < \cdots < x_m = b$ en $y_0 = c < y_1 < \cdots < y_n = d$ beschouwen we de Lagrange veeltermen $\ell_i^x(x) = \Pi_{k=0,k\neq i}^m \frac{x-x_k}{x_i-x_k}$ (zie ook punt 72) en vervolgens de functie

$$P^x(f) : \Omega \to \mathbb{R}, \quad P^x(f)(x, y) = \sum_{i=0}^{m} f(x_i, y)\ell_i^x(x). \tag{19}$$

Merk op dat deze functie de waarden van f gebruikt langs de (verticale) lijnstukken $\{x_i\} \times [c, d]$. M.a.w. de waarden van f moeten beschikbaar zijn langs zulke lijnstukken. Er geldt (ga na!)

$$P^x(f)(x_i, y) = f(x_i, y), \quad \text{voor alle } y \in [c, d] \text{ en } i = 0, \ldots, m. \tag{20}$$

Analoog kunnen we de interpolerende veeltermen in de y-richting beschouwen, nl.

$$P^y(f) : \Omega \to \mathbb{R}, \quad P^y(f)(x, y) = \sum_{j=0}^{n} f(x, y_j)\ell_j^y(y) \tag{21}$$

met $\ell_j^y : [c, d] \to \mathbb{R}$, $\ell_j^y(y) = \Pi_{k=0,k\neq j}^m \frac{y-y_k}{y_j-y_k}$ $(j = 0, \ldots, n)$. Nu moeten de waarden van f langs de horizontale lijnstukken $[a, b] \times \{y_j\}$ beschikbaar zijn en er geldt

$$P^y(f)(x, y_j) = f(x, y_j), \quad \text{voor alle } x \in [a, b] \text{ en } j = 0, \ldots, n. \tag{22}$$

Merk op: Men kan gelijkaardige functies definiëren beginnend met eendimensionale stuksgewijze veeltermen (splines) of met Hermite interpolerende veeltermen.

De interpolerende functies $P^x(f)$ en $P^y(f)$ gebruiken de waarden van f op lijnstukken op een asymmetrische manier (of vertikale lijnstukken, of horizontale lijnstukken). Deze functies kunnen gebruikt worden om andere interpolerende functies te

definiëren die de beschikbare informatie symmetrisch gebruiken. Bijv. als men f kent in slechts de punten (x_i, y_j) $(i = 0, \ldots, m$ en $j = 0, \ldots, n)$ dan moet eerst een interpolerende functie gebruikt worden voor $f(x_i, y)$ met $y \in [c, d]$ en $i = 0, \ldots, m$, resp. voor $f(x, y_j)$ met $x \in [a, b]$ en $j = 0, \ldots, n$. Een mogelijkheid is de samenstelling van $P^x(f)$ en $P^y(f)$ te beschouwen, nl. $P(f) : \Omega \to \mathbb{R}$, $P = P^x(P^y(f))$. Dan geldt

$$P(f)(x, y) = \sum_{i=0}^{m} P^y(f)(x_i, y)\ell_i^x(x) = \sum_{i=0}^{m}\sum_{j=0}^{n} f(x_i, y_j)\ell_j^y(y)\ell_i^x(x). \tag{23}$$

Voor deze functie geldt

$$P(f)(x_i, y_j) = f(x_i, y_j), \quad \text{voor alle } i = 0, \ldots, m \text{ en } j = 0, \ldots, n. \tag{24}$$

De functie in (23) wordt bepaald door slechts de waarden $f(x_i, y_j)$. In sommige situaties is er meer informatie beschikbaar over f en men wil deze ook gebruiken (denk bijv. aan functies die een zekere waarde, bijv. 0, langs bepaalde lijnstukken aanneemt). Een andere multivariabele interpolerende functie is $P^{xy}(f) : \Omega \to \mathbb{R}$, $P^{xy}(f) = P^x(f) + P^y(f) - P(f)$ met $P(f)$ gegeven in (23).

Als afsluiting van dit punt los de volgende vragen op.

(a) Toon aan dat voor de interpolerende functie $P(f) = P^y(P^x(f))$ (zie (24)) geldt $P^{xy}(f)(x_i, y_j) = f(x_i, y_j)$ voor alle $i = 0, \ldots, m$ en $j = 0, \ldots, n$.

(b) Bepaal de uitdrukking van $P^{xy}(f)$ en toon aan dat er geldt

$$P^{xy}(f)(x_i, y) = f(x_i, y) \text{ en } P^{xy}(f)(x, y_j) = f(x, y_j) \tag{25}$$

voor alle $x \in [a, b]$, $y \in [c, d]$, $i = 0, \ldots, m$, en $j = 0, \ldots, n$.

(c) Zij $\Omega = [0, \ell] \times [0, h]$. Bepaal de functie die f in de punten $(0, 0)$, $(\ell, 0)$, (ℓ, h) en $(0, h)$ interpoleert.

(d) Een sporthal moet gebouwd worden op de horizontale oppervlakte $[-a, a] \times [-b, b]$. De muren hebben de hoogte H overal. Help de architect een dak te ontwerpen wetend dat het dak overal op de muren leunt en dat de maximale hoogte van het gebouw, $H + h$, in het midden $(0, 0)$ wordt bereikt. Beschouw $a, b, H, h > 0$ als gegeven parameters.
Hint: Denk aan (25).

107. *Driehoeken* (zie ook paragraaf 8.6.2)
Zij $h > 0$ gegeven en beschouw de driehoekige verzameling $T_h = \{(x, y) \in \mathbb{R}^2 / 0 \leq x, y \leq h, x + y \leq h\}$. Zoals in rechthoekige gebieden beschouwen we nu interpolerende functies langs lijnstukken. Voor de eenvoud beschouwen we alleen affiene interpolatie. We beginnen eerst met de interpolatie in de x-richting, dus voor een vaste $y \in [0, h]$ en definiëren een functie die affien is in de x variabele en f interpoleert in de punten $(0, y)$ en $(h - y, y)$. We beschouwen

$$P^x(f) : [0, h - y] \to \mathbb{R}, \quad P^x(f)(x, y) = \frac{h - x - y}{h - y} f(0, y) + \frac{x}{h - y} f(h - y, y).$$

Vervolgens zetten we $x \in [0,h]$ vast en beschouwen

$$P^y(f) : [0, h-x] \to \mathbb{R}, \quad P^y(f)(x,y) = \frac{h-x-y}{h-x} f(x,0) + \frac{y}{h-x} f(x, h-x).$$

Tenslotte laten we de som $x + y = c \in [0,h]$ constant blijven (en dus $x \in [0,c], y \in [0, c-x]$) en definiëren

$$P^{xy}(f)(x,y) = \frac{x}{x+y} f(x+y, 0) + \frac{y}{x+y} f(0, x+y).$$

Deze drie interpolerende functies kunnen gebruikt worden om interpolerende functies in T_h te definiëren. De simpelste variant is

$$P(f) : T_h \to \mathbb{R}, \quad P(f) = P^x(P^y(P^{xy}(f))). \tag{26}$$

Een andere interpolerende functie is bijv. $S(f) = \frac{1}{2}\big[P^x(f) + P^y(f) + P^{xy}(f) - P(f)\big]$.

(a) Toon aan dat voor de interpolerende functie in (26)

$$P(f)(x,y) = \frac{h-x-y}{h} f(0,0) + \frac{x}{h} f(h,0) + \frac{y}{h} f(0,h). \tag{27}$$

Merk op dat deze functie een 1^e graads veelterm is in twee variabelen.

108. Een andere manier om de veelterm bij punt 107 te bepalen is alweer via *basisfuncties*. M.a.w. we zoeken naar functies $\varphi_k : T_h \to \mathbb{R}$ $(k = 0, 1, 2)$, $\varphi_k(x,y) = a_k + b_k x + c_k y$ die voldoen aan de condities:

$$\begin{aligned} &\varphi_0(0,0) = 1, \varphi_0(h,0) = 0, \varphi_0(0,h) = 0, \\ &\varphi_1(0,0) = 0, \varphi_1(h,0) = 1, \varphi_1(0,h) = 0, \\ &\varphi_2(0,0) = 0, \varphi_2(h,0) = 0, \varphi_2(0,h) = 1. \end{aligned}$$

Bepaal de functies φ_k en toon aan dat voor de veelterm $P(f)$ in (27) geldt

$$P(f)(x,y) = f(0,0)\varphi_0(x,y) + f(h,0)\varphi_1(x,y) + f(0,h)\varphi_2(x,y).$$

109. *Stuksgewijze interpolatie*
Zoals in het eendimensionale geval is het delen van Ω in deelverzamelingen een praktische oplossing om simpele interpolerende functies te bepalen die ook goede convergentie eigenschappen hebben. Een mogelijkheid is Ω te *traingulariseren*, d.w.z. dat Ω opgedeeld wordt in niet overlappende driehoeken. Tenslotte wordt in elke driehoek een interpolerende functie gedefinieerd en er worden condities opgelegd bij de overgang tussen twee aangrenzende driehoeken. Dit is precies het idee van de *eindige elementen methode*.

Veronderstel dat de hoekpunten van een driehoek T zijn $P_k = (x_k, y_k)$, $k = 1, 2$ of 3. Zoals voor Lagrange interpolerende veeltermen (zie punt 72) kunnen we *basisfuncties*

$\varphi_k : T \to \mathbb{R}$ definiëren die 1^e graads veeltermen zijn in x en y en voldoen aan $\varphi_k(P_j) = \delta_{kj}$ (het Kronecker symbool) voor $j \in \{1, 2, 3\}$. Dan is $P_1(f) : T \to \mathbb{R}$,

$$P_1(f)(x, y) = \sum_{k=1}^{3} f(x_k, y_k)\varphi_k(x, y) \tag{28}$$

een interpolerende veelterm voor f in de hoekpunten van T.

Om een expliciete vorm van de functies φ_k te schrijven kunnen deze in termen van oppervlakten geïnterpreteerd worden. Om preciezer te zijn, zij $opp(P, Q, R)$ de oppervlakte van de driehoek met de hoekpunten P, Q en R. Er geldt

$$opp(P, Q, R) = \frac{1}{2}|det(P, Q, R)| \quad \text{met} \quad det(P, Q, R) = \begin{vmatrix} 1 & x_P & y_P \\ 1 & x_Q & y_Q \\ 1 & x_R & y_R \end{vmatrix}.$$

Daarmee kunnen we nu de expliciete vorm van de basisfuncties φ_k schrijven.

(a) Zij $P = (x, y)$ een willekeurig punt in T. Toon aan dat als i, j de indices van de andere twee hoekpunten van T zijn dan P_k dan geldt:

$$\varphi_k(x, y) = \frac{opp(P, P_i, P_j)}{opp(P_k, P_i, P_j)} = \left| \frac{det(P, P_i, P_j)}{det(P_k, P_i, P_j)} \right|. \tag{29}$$

110. *Verspreide data, Shepard interpolatie* (*ter info*)
Als de punten $P_k = (x_k, y_k)$ $(k = 1, \ldots, n)$ verspreid zijn in $\mathbb{R}^2$ zonder dat deze in een rechthoekige, driehoekige of gelijkaardige structuur georganiseerd kan worden kan men een idee toepassen dat ook bij de interpolerende veeltermen volgens Lagrange werd gebruikt (punt 72). Voor gegeven punten $P, Q \in \mathbb{R}^2$ zij $d(P, Q)$ de afstand tussen P en Q. Deze afstand kan de standaard, euclidische afstand zijn maar ook een andere, bijv. $d_r(P, Q) = \left(|x_P - x_Q|^r + |y_P - y_Q|^r\right)^{1/r}$ met $r \in [1, \infty]$. Beschouw nu de *basisfuncties*

$$u_k : \mathbb{R}^2 \to \mathbb{R}, \quad u_k(P) = \Pi_{i=1, i\neq k}^{n} \frac{d(P, P_i)}{d(P_k, P_i)} \text{ voor elke } P = (x, y) \in \mathbb{R}^2. \tag{30}$$

(a) Zij $f : \mathbb{R}^2 \to \mathbb{R}$ een continue functie en $\mathcal{P}(f) : \mathbb{R}^2 \to \mathbb{R}$,

$$\mathcal{P}(f)(P) = \sum_{k=1}^{n} f(P_k)u_k(P) \text{ voor elke } P = (x, y) \in \mathbb{R}^2.$$

Toon aan dat $\mathcal{P}(f)$ de functie f interpoleert in de punten P_k.

Les 8

Numerieke differentiatie

In dit hoofdstuk leren we afgeleiden van een functie te benaderen bij gegeven informatie. Een voorbeeld is het volgende: zij de punten (x_i, y_i) gegeven ($i = 0, \ldots, n$, $n \in \mathbb{N}$) en beschouw de continu differentiëerbare functie $f : \mathbb{R} \to \mathbb{R}$ z.d. $f(x_i) = y_i$ voor alle i. Voor gegeven $x \in \mathbb{R}$ benader de afgeleide $f'(x)$.
Meer details zijn te lezen in paragraaf 10.10, 10.10.1 of in hoofdstuk 5 van het boek van A. Bultheel, paragraaf 1–4, 6, 7. Voor de eennvoud gebruiken we de volgende notaties:

$$f_i := f(x_i), \quad f_i^1 := f'(x_i) \quad \text{en i.h.a.} \quad f_i^k := f^{(k)}(x_i) \quad (k \in \mathbb{N}, i = 0, \ldots, n).$$

111. *Numerieke differentiatie door afleiden van de interpolerende veelterm*
Zij $f : \mathbb{R} \to \mathbb{R}$ $n+1$ keer continu differentiëerbaar en de punten (x_i, y_i) met $y_i = f(x_i)$ ($i = 0, \ldots, n$, $n \in \mathbb{N}$). Zij $\Pi_n \in \mathbb{P}_n$ de bijbehorende (Lagrange) interpolerende veelterm en $x \in \mathbb{R}$ en $E_n(x) = f(x) - \Pi_n(x)$ de interpolatiefout. Hieruit volgt

$$f'(x) = \Pi_n'(x) + E_n'(x).$$

M.a.w. de fout bij het benaderen van de afgeleide $f'(x)$ door $\Pi_n'(x)$ is $E_n'(x)$. Gebruik makend van de uitdrukking voor de interpolatiefout (zie ook punt 74 en merk op dat het punt ζ van x afgangt, $\zeta = \zeta(x)$) krijgen we

$$f'(x) - \Pi_n'(x) = \frac{1}{(n+1)!}\omega_{n+1}(x)f^{(n+2)}(\zeta(x))\zeta'(x) + \frac{1}{(n+1)!}\omega_{n+1}'(x)f^{(n+1)}(\zeta(x)).$$

Leg nu uit waarom deze formule in het algemeen niet echt praktisch bruikbaar is en waarom dan wel voor het geval $x = x_i$. Geef een schatting van $f'(x_i) - \Pi_n'(x_i)$.

112. *Conditionering van de differentiatie*
De differentiatie is *slecht geconditioneerd* met als gevolg dat de numerieke differentiatie methodes zorgvuldig gekozen moeten worden. Een simpel voorbeeld in deze zin is het volgende. Zij f continu differentiëeerbaar en met $0 < \varepsilon << 1$ (klein getal) beschouw de verstoring van f

$$f_\varepsilon : \mathbb{R} \to \mathbb{R}, \quad f_\varepsilon(x) = f(x) + \varepsilon \sin\big(\frac{x}{\varepsilon^2}\big).$$

Merk op dat hoewel deze verstoring klein is, nl. $O(\varepsilon)$, de verstoring in de afgeleide heel groot kan zijn,

$$f_\varepsilon'(x) = f'(x) + \frac{1}{\varepsilon}\cos\big(\frac{x}{\varepsilon^2}\big).$$

In dit geval geldt dat hoe kleiner de verstoring van f is hoe groter die in f'.

113. Bestudeer de stabiliteit van de numerieke differentiatie in de volgende situatie. Zij $f : [a, b] \to \mathbb{R}$ een C^2 functie en $x_0 \in [a, b]$ en $h \in \mathbb{R}\backslash\{0\}$ z.d. $x_0 - h \in [a, b]$. Een

benadering voor $f_0^1 = f'(x_0)$ is $y_0^1 = \frac{f(x_0)-f(x_0-h)}{h}$. Veronderstel dat i.v.m. de representatie in een computer slechts benaderingen van $f(x_0)$ en $f(x_0-h)$ beschikbaar zijn met relatieve fouten ε_0 en ε_1 ($|\varepsilon_k| \leq \texttt{eps}$, $k = 1, 2$). M.a.w. wordt f_0^1 benaderd door $z_0^1 = \frac{f(x_0)(1+\varepsilon_0)-f(x_0-h)(1+\varepsilon_1)}{h}(1+3\varepsilon_2)$ met $|\varepsilon_2| \leq \texttt{eps}$. Merk op, de factor $(1+3\varepsilon_2)$ is afkomstig van twee operaties: het aftrekken van twee computerwaarden van f en de deling door h, plus de afronding van h als computergetal. Na het verwaarlozen van de hogere orde termen blijft dus $3\varepsilon_2$ als perturbatie over. Wat is de relatieve fout in de benadering van f_0^1? M.a.w. geef een schatting voor $|y_0^1 - z_0^1|/|y_0^1|$. In welke situatie is deze fout groot?

114. *Differentiatie volgens Lagrange*
Zoals gezien kan de Lagrange interpolerende veelterm gebruikt worden om afgeleiden te benaderen en dat liefst in de interpolatiepunten. In dit geval geldt

$$f_k^1 = f'(x_k) \approx \sum_{i=0}^{n} f_i \ell_i'(x_k).$$

We veronderstellen nu dat de interpolatiepunten uniform verdeeld zijn op het interval $[a, b]$, nl. $x_i = a + ih$ met $i = 0, \ldots, n$ en $h = \frac{b-a}{n}$. Merk op dat voor elke $x \in [a, b]$ er een unieke $u \in [0, n]$ bestaat z.d. $x = a + uh$. Dan geldt voor de Lagrange veeltermen

$$\ell_i(x) = \tilde{\ell}_i(u) = (-1)^i \frac{u(u-1)\ldots(u-i+1)(u-i-1)\ldots(u-n)}{i!(n-i)!}$$

(ga na!) en volgens de kettingregel dat $\ell_i'(x) = \frac{1}{h}\tilde{\ell}_i'(u)$. Toon aan dat voor de fout op de numerieke differentiatie geldt

$$|f'(x_i) - \Pi_n'(x_i)| \leq \frac{i!(n-i)!}{(n+1)!} h^n \|f^{(n+1)}\|_\infty,$$

waarin $\|g\|_\infty$ de maximum-norm van g is (het maximum van g in $[a, b]$).

115. Zij $n = 2$ en $x_i = a + ih$ met $i = 0, 1, 2$ en $h = (b-a)/2$. Zoals boven gezien geldt voor $u = (x-a)/h$

$$f(x) = \frac{(u-1)(u-2)}{2} f_0 - u(u-2) f_1 + \frac{u(u-1)}{2} f_2.$$

Hieruit volgt de onderstaande formule voor de numerieke differentiatie (vergeet de kettingregel niet!)

$$f'(x) = \frac{1}{h}\left(\frac{2u-3}{2} f_0 - (2u-2) f_1 + \frac{2u-1}{2} f_2\right).$$

Zoals gezien zijn de formules voor de numerieke differentiatie volgens Lagrange vooral bruikbaar als de benadering in een van de ingterpolatiepunten wordt beschouwd. Leid de formule af voor de benadering van f_i^1 voor $i = 0, 1, 2$ en geef in een schatting van de fout in elk punt.

116. Gegeven $(x_i, y_i^{(0)}, y_i^{(1)})$, $i = 0, \ldots, n$ z.d. voor de C^2 functie $f : \mathbb{R} \to \mathbb{R}$ geldt $f^{(k)}(x_i) = y_i^{(k)}$ voor $k = 0$ en $k = 1$ en $i = 0, \ldots, n$. Hoe zou je de afgeleide $f'(x)$ benaderen? Geef een motivatie voor de keuze.
Opmerking: er wordt niet een bepaalde methode verwacht!

117. Een manier om numerieke differentiatie formules af te leiden en tevens om een schatting te geven voor de fout is gebaseerd op taylorreeksen. Als voorbeeld nemen we de *centrale differentieformule*

$$f'(x_1) = f_1^1 \approx \frac{f_2 - f_0}{2h}, \tag{31}$$

met $x_i = a + ih$ $(i = 0, 1, 2)$, $h = (b - a)/2$ en $f_i = f(x_i)$. We veronderstellen dat $f \in C^3$ en gebruiken de Taylor reeksen rond x_1

$$f_0 = f_1 - hf_1^1 + \frac{h^2}{2} f_1^2 - \frac{h^3}{6} f'''(\zeta_0), \text{ resp. } f_2 = f_1 + hf_1^1 + \frac{h^2}{2} f_1^2 + \frac{h^3}{6} f'''(\zeta_2),$$

met $\zeta_0, \zeta_2 \in (a, b)$. Als we deze benaderingen in de rechterzijde van (31) meenemen krijgen we

$$\frac{f_2 - f_0}{2h} = f_1^1 + \frac{h^2}{12}\big(f'''(\zeta_0) + f'''(\zeta_2)\big).$$

Omdat $f \in C^3$ bestaat er een $\zeta_1 \in (a, b)$ z.d. $f'''(\zeta_1) = \frac{1}{2}(f'''(\zeta_0) + f'''(\zeta_2))$ en vervolgens

$$f_1^1 = \frac{f_2 - f_0}{2h} - \frac{h^2}{6} f'''(\zeta_1).$$

Als we alleen in de grootorde van de fout geïnteresseerd zijn dan kan de laatste geschreven worden als

$$f_1^1 = \frac{f_2 - f_0}{2h} + O(h^2).$$

118. Herhaal de stappen van boven om f_0^1 en f_2^1 te benaderen.

119. *Hogere orde afgeleiden* Deze strategiëen kunnen ook gebruikt worden voor de benadering van hogere orde of van partiële afgeleiden. De formule

$$f_1^2 \approx \frac{1}{h^2}(f_0 - 2f_1 + f_2)$$

wordt veel gebruikt in de numerieke benadering van oplossingen van differentiaalvergelijkingen. Ook in dit geval kan de fout bepaald worden m.b.v. Taylor reeksen. Zij $f \in C^4$. Zoals boven geldt

$$f_0 = f_1 - hf_1^1 + \frac{h^2}{2} f_1^2 - \frac{h^3}{6} f_1^3 + \frac{h^4}{24} f^{(iv)}(\zeta_0),$$
$$f_2 = f_1 + hf_1^1 + \frac{h^2}{2} f_1^2 + \frac{h^3}{6} f_1^3 + \frac{h^4}{24} f^{(iv)}(\zeta_2),$$

met $\zeta_0, \zeta_2 \in (a,b)$. Omdat $f \in C^4$ bestaat er een $\zeta_1 \in (a,b)$ z.d.

$$\frac{1}{h^2}(f_0 - 2f_1 + f_2) = f_1^2 + \frac{h^2}{12} f^{(iv)}(\zeta_1),$$

of simpeler $f_1^2 = \frac{1}{h^2}(f_0 - 2f_1 + f_2) + O(h^2)$.

120. Toon aan dat er geldt

$$\begin{aligned}
f_0^1 &= \frac{1}{h}(f_1 - f_0) + O(h) \text{ (voorwaartse differentie)},\\
f_1^1 &= \frac{1}{h}(f_1 - f_0) + O(h) \text{ (rugwaartse differentie)},\\
f_0^1 &= \frac{1}{12h}(f_{-2} - 8f_{-1} + 8f_1 - f_2) + O(h^4),\\
f_0^2 &= \frac{1}{h^2}(f_0 - 2f_1 + f_2) + O(h),\\
f_0^3 &= \frac{1}{2h^3}(-f_{-2} + 2f_{-1} - 2f_1 + f_2) + O(h^2).
\end{aligned}$$

Merk op het verschil in de orde van de fout bij de benadering van f_1^1 met centrale resp. rugwaartse differenties. Tevens merk op dat dezelfde formule wordt gebruikt voor de benadering van zowel f_0^2 en f_1^2. Wat is het effect op de orde van de fout?

121. Zoals volgt uit de formules voor de differentiatiefout geldt de algemene vuistregel: hoe dichter bij elkaar de interpolatiepunten en het punt x waarin de afgeleide wordt benaderd, hoe kleiner de fout. Daarom kunnen we zoals in les 6 splines gebruiken voor de numerieke differentiatie. Beschouw een simpel geval met de punten (x_i, y_i), $i = 0,1,2,3$ met $x_i = ih$ ($h > 0$ is gegeven) en benader de afgeleiden $f'(x_0)$, $f'(x_1)$ en $f"(x_1)$ voor een functie $f \in C^2$ waarvoor geldt $f(x_i) = y_i$ ($i = 0,1,2,3$). Gebruik twee methoden, gebaseerd op natuurlijke splines en op (Lagrange of Newton) interpolerende veelterm. Vergelijk de resultaten.

122. Gebruik de resultaten van boven voor $h = 1/3$ en voor de functies $f(x) = \sin(\pi x)$ resp. $f(x) = e^x$.

123. Een manier om de nauwkeurigheid van de numerieke differentiatie te verbeteren is de *extrapolatiemethode*. Dit is een algemeen idee dat ook in andere situaties toepasbaar is. Als voorbeeld nemen we de *voorwaartse differentie* benadering van afgeleiden, nl. $f'(x_0) \approx f_{x_0}^1(h)$ met $f_{x_0}^1 : \mathbb{R} \to \mathbb{R}$ gegeven als

$$f_{x_0}^1(h) := \frac{f(x_0 + h) - f(x_0)}{h}. \tag{32}$$

Als $f \in C^3$ dan bestaat er voor elke $h \in \mathbb{R}$ een $\zeta = \zeta(h)$ z.d. geldt (denk aan de 2^e orde Taylor benadering van f rond x_0!)

$$f_{x_0}^1(h) = f'(x_0) + \frac{h}{2} f"(x_0) + \frac{h^2}{6} f"'(\zeta(h)). \tag{33}$$

Volgens de definitie van de afgeleide geldt $f'(x_0) = f^1_{x_0}(0)$ maar de vraag is of en hoe men $f^1_{x_0}(0)$ kan berekenen.

Een manier is de zgn. extrapolatie. Dit is niets anders dan $p(0)$ waarbij p een benaderende functie is van f^1. Als voorbeeld veronderstellen we dat de waarden $f(x_0)$, $f(x_0+h)$, $f(x_0+2h)$ wel gegeven zijn. Dan kunnen ook de benaderingen van $f'(x_0)$ bepaald worden, $f^1_{x_0}(h)$ en $f^1_{x_0}(2h)$. Net zoals in (33) geldt

$$f^1_{x_0}(2h) = f'(x_0) + hf"(x_0) + \frac{2h^2}{3} f'''(\zeta(2h)),$$

voor een zekere $\zeta = \zeta(2h) \in \mathbb{R}$.

De eerste orde interpolerende veelterm $p \in \mathbb{P}_1$ van $f^1_{x_0}$ is (ga na!)

$$p(t) = \frac{2h-t}{h} f^1_{x_0}(h) + \frac{t-h}{h} f^1_{x_0}(2h).$$

Hiermee kan $f^1_{x_0}(0)$ en dus ook $f'(x_0)$ benaderd worden als

$$f'(x_0) = f^1_{x_0}(0) \approx p(0) = 2f^1_{x_0}(h) - f^1_{x_0}(2h).$$

Dit is eigenlijk een interpolatie maar omdat 0, het punt waarin de waarde wordt benaderd, buiten het interpolatie interval $[h, 2h]$ ligt heet de procedure "extrapolatie". Het is ook een simpele manier om de nauwkeurigheid van de benadering te verbeteren. Dit volgt uit (merk op, $f \in C^3$!)

$$f'(x_0) = 2f^1_{x_0}(h) - f^1_{x_0}(2h) + O(h^2). \tag{34}$$

Geef een bewijs voor (34). Geef de formule voor de benadering van $f'(x_0)$.

124. Zij $f : (0, \infty) \to \mathbb{R}$, $f(x) = \ln x$. We benaderen $f'(1)$ gebruik makend van de formules (32) en (34). Inspecteer de (afname van) de fout voor $h_k = 2^{-k}$ met $k = 0, 1, \ldots, 4$. Doe dit in een MATLAB programma en plot de twee fouten in een grafiek op de logaritmische schaal.

Les 9

Numerieke integratie

In dit hoofdstuk leren we bepaalde integralen te benaderen gebaseerd op gegeven informatie over een (integreerbare) functie f. Als voorbeeld beschouwen we het volgende probleem. Gegeven de punten (x_i, y_i) $(i = 0, \ldots, n,\ n \in \mathbb{N})$ met $x_i \in [a, b]$ en wetende dat de functie $f : [a, b] \to \mathbb{R}$ integreerbaar is en voldoet aan $f(x_i) = y_i$ voor alle i, benader de integraal $\int_a^b f(x)dx$.
De algemene vorm van deze benadering luidt

$$\int_a^b f(x)dx \approx I^n(f) = \sum_{k=0}^{n} \alpha_k f(x_k). \tag{35}$$

Dit is een *kwadratuurformule* met *gewichten/weights* α_k en de *abscissen/nodes* x_k $(k = 0, \ldots, n)$. De coëfficiënten α_k en soms zelfs de punten x_k worden bepaald aan de hand van zekere criteria. Een voorbeeld is de *nauwkeurigheidsgraad*. Een kwadratuurformule heeft de nauwkeurigheid/exactness $m \in \mathbb{N}$ als de formule exact is voor alle veeltermen met de graad m of lager en dat er ten minste een $m + 1$-de graad veelterm p bestaat waarvoor de formule niet exact is:

$$I^n(p) = \int_a^b p(x)dx \text{ voor alle } p \in \mathbb{P}_m \text{ en er bestaat } q \in \mathbb{P}_{m+1} \text{ z.d. } I^n(q) \neq \int_a^b q(x)dx.$$

125. Bestudeer hoofdstuk 9.1 over *kwadratuurformules*.

126. Zij $a, b \in \mathbb{R}$, $a < b$, $f \in C[a, b]$ en $f_n \in C[a, b]$ een benadering van f. Zij $I(f) = \int_a^b f(x)dx$ en $I^n(f) = \int_a^b f_n(x)dx$. Met $\|f\|_\infty = \max_{x\in[a,b]} |f(x)|$ geef een bewijs voor de foutschatting
$$\big|I(f) - I^n(f)\big| \leq (b - a)\|f - f_n\|_\infty.$$

127. Bestudeer hoofdstuk 9.1 over *interpolerende kwadratuurformules* (zie ook hoofdstuk 6, paragraaf 1–3 in het boek van A. Bultheel). Laat de theorie over stuksgewijze interpolatie/composite interpolerende veeltermen buiten beschouwing.

128. Merk op dat de interpolatiefout voor interpolerende veeltermen volgens Newton (zie hoofdstuk 8, paragraaf 8.2) geldt
$$I(f) - I^n(f) = \int_a^b f[x_0, x_1, \ldots, x_n, x]\omega_{n+1}(x)dx.$$
Dit rechtwaardigd de continuiteit (in x) van de functie $f^{(n+1)}(\xi)$ die voorkomt in de restterm.

129. Zij $f : [a, b] \to \mathbb{R}$ continu differentiëerbaar. Geef een bewijs voor de kwadratuurformule (met integratiefout)
$$\int_a^b f(x)dx = (b - a)f(a) + \frac{(b - a)^2}{2} f'(\zeta), \quad \text{voor een } \zeta \in (a, b).$$

Hint: Gebruik de volgende middelwaarde stelling voor integralen:
Zij $g, h : [a, b] \to \mathbb{R}$ continu en veronderstel dat $h(x) \geq 0$ voor alle $x \in [a, b]$. Dan bestaat er een $\zeta \in (a, b)$ z.d.

$$\int_a^b g(x)h(x)dx = g(\zeta) \int_a^b h(x)dx.$$

Denk aan de interpolatiefout voor de interpolerende veelterm volgens Newton om de functies g en h geschikt te kiezen.

130. Toon aan dat de n-de graads interpolerende kwadratuurformules die gebaseerd zijn op veelterm interpolatie exact zijn voor n-de graads veeltermen. M.a.w. geldt

$$I(p) = I^n(p), \qquad \text{voor elke } p \in \mathbb{P}_n.$$

131. Zij $h = (b - a)/2$. Bepaal de gewichten $\alpha_0, \alpha_1, \alpha_2 \in \mathbb{R}$ zodanig dat de kwadratuurformule

$$\int_a^b f(x)dx \approx \alpha_0 f(a) + \alpha_1 f(a + h) + \alpha_2 f(a + 2h)$$

een zo hoog mogelijke nauwkeurigheidsgraad heeft.

Newton-Cotes formules

132. Bestudeer paragraaf 9.3 (zie als alternatief ook paragraaf 6.6 in het boek van A. Bultheel). Vaak worden *Newton-Cotes* formules in twee categoriën verdeeld, open en gesloten. Deze verdeling hangt van de abscissen af, namelijk

$$\begin{array}{ll} gesloten\ n-de\ graads formule: & x_k = a + k\frac{b-a}{n}, k = 0, \ldots, n \\ open\ n-de\ graads formule: & x_k = a + (k+1)\frac{b-a}{n+2}, k = 0, \ldots, n. \end{array}$$

Je hoeft het bewijs van Stelling 9.2 niet te leren.

133. Leid de volgende kwadratuurformule (de 3/8 formule) af:
Zij $f \in C^4([a, b])$ en $x_k = a + kh$ ($k \in \{0, 1, 2, 3\}$) met $h = \frac{b-a}{3}$. Dan bestaat er een $\zeta \in (a, b)$ z.d.

$$\int_a^b f(x)dx = \frac{3h}{8}\big(f(x_0) + 3f(x_1) + 3f(x_2) + f(x_3)\big) - \frac{3}{80}h^5 f^{(4)}(\zeta).$$

134. Bepaal de open Newton Cotes formule voor $n = 1$ (dus met twee abscissen). M.a.w. bepaal de gewichten en de abscissen voor deze kwadratuurformule en geef ook de integratiefout. Hiervoor kunnen de formules voor de integratiefout in de stelling 9.2 op blz. 388 gebruikt worden.

135. Zij $f : [0,1] \to \mathbb{R}$, $f(x) = x^{5/2}$. Bestudeer het gedrag van de benadering van $\int_0^1 f(x)dx$ m.b.v. gesloten Newton-Cotes formules van nauwkeurigheid n. Neem bijv. $n = 1, 2, 3, 4, 5$ (je mag ook MATLAB hiervoor gebruiken). Wat valt hier op? Leg het gedrag uit.

136. Voor de functie $f \in C^1[a,b]$ zijn de waarden $f(a)$, $f(b)$, $f'(a)$ en $f'(b)$ gegeven. Gebruik deze informatie om een kwadratuurformule te bepalen voor de benadering van $\int_a^b f(x)dx$. Geef ook een schatting van de fout.
Hint: Zie ook paragraaf 9.5.

137. Alle voorbeelden zijn gebaseerd op interpolerende veeltermen. In les 6 hebben we ook andere interpolerende functies bestudeerd, nl. splines. De natuurlijke kubische splines hebben een bijzondere eigenschap, ze zijn C^2 interpolerende functies die de norm van de 2-de afgeleide minimaliseren. Deze splines kunnen natuurlijk ook gebruikt worden om kwadratuurformules te definiëren. Zoals voor bijv. de Lagrange veeltermen ℓ_k kunnen eerst "basis splines" s_k bepaald worden die voldoen aan $s_k \in S_h^{3,2}[a,b]$, $s_k(x_j) = \delta_{kj}$, $k, j \in \{0, \ldots, n\}$. Voor de praktische implementatie kan het algoritme bij punt 103 gebruikt worden. Merk op dat het stelsel (18) steeds dezelfde matrix heeft en daarom is de LU decompositie uiterst geschikt.

    Gebruik dit idee om een kwadratuurformule te bepalen die de natuurlijke splines gebruiken met slechts drie punten, a, $\frac{a+b}{2}$ en b (de spline functie kan ook direct bepaald worden zonder dat er eerst de basis splines zoals boven bepaald zijn). Wat is de fout?

Samengestelde formules/composite formulae

Net als bij veelterminterpolatie is een hogere nauwkeurigheidsgraad geen garantie voor een betere benadering van de integraal $I(f) = \int_a^b f(x)dx$. Denkend aan de vorm van de integratiefout, maar ook intuïtief kan het voordeliger zijn het oorspronkelijke interval in kleinere intervallen te delen en vervolgens de integraal te berekenen.

138. Bestudeer paragraaf 9.4 (zie ook hoofdstuk 6, paragraaf 8–12 in het boek van A. Bultheel).

139. Beschouw de kwadratuurformule bij punt 133. Leid de bijbehorende samengestelde formule af. Wat is de integratiefout?

140. Door het interval (a,b) op te splitsen in in meerdere deelintervallen, kan de integraal steeds nauwkeuriger benaderd worden. Een nadeel hiervan is het feit dat er steeds meer functie-berekeningen $f(x)$ nodig zijn en deze zijn meestal tijdrovend. De volgende ideëen zijn handig voor een efficiënte kwadratuurformule: ten eerste moeten de reeds berekende functiewaarden opnieuw gebruikt worden om het aantal functie-evaluaties te verminderen en ten tweede moet de fout aan de hand van de

berekeningen geschat worden om onnodige verfijning en berekeningen te vermijden. Hieronder gebruiken we de samengestelde trapeziumregel om een voorbeeld te geven hoe de twee ideëen toegepast kunnen worden.

141. *Verminderen van functieberekeningen*
Beschouw de samengestelde trapeziumregel

$$I_m = \frac{h}{2}\big\{f(a) + f(b) + 2\sum_{k=1}^{m-1} f(x_k)\big\}, \tag{36}$$

met $h = (b-a)/m$ en $x_k = a+kh$. Bij een verdubbeling van het aantal deelintervallen krijgen we met $\tilde{h} = h/2$ en $\tilde{x}_k = a + k\tilde{h}$, $k = 0, \ldots, 2m$:

$$I_{2m}(f) = \frac{\tilde{h}}{2}\big\{f(a) + f(b) + 2\sum_{k=1}^{2m-1} f(\tilde{x}_k)\big\}. \tag{37}$$

Merk op dat $\tilde{x}_{2k} = x_k$ voor alle $k = 0, \ldots, m$. Daarom volgt uit (36) en (37)

$$I_{2m}(f) = \frac{1}{2}I_m(f) + \frac{h}{2}\sum_{k=1}^{m} f(\tilde{x}_{2k-1}). \tag{38}$$

Merk op dat bij het berekenen van I_{2m} slechts m nieuwe functiewaarden bepaald moeten worden.

142. *A posteriori foutschatting*
De integratiefout bij de samengestelde trapeziumregel met m deelintervallen is (zie ook de kwadratuurfout na (9.14) op blz. 384 in het boek)

$$I(f) - I_m(f) \leq -\frac{b-a}{12}h^2 f''(\xi),$$

met $h = (b-a)/m$ en voor een zekere $\xi \in (a, b)$. Dat kan geschreven worden als

$$I(f) - I_m(f) = C_f h^2 + O(h^3), \tag{39}$$

met een $C_f \in \mathbb{R}$ afhankelijk van f, a, b en natuurlijk h. Als het aantal deelintervallen verdubbeld wordt dan geldt

$$I(f) - I_{2m}(f) = \tilde{C}_f \left(\frac{h}{2}\right)^2 + O(h^3). \tag{40}$$

Voor gladde f en kleine h kunnen we aannemen dat $C_f \approx \tilde{C}_f$. Dan kan C_f en vervolgens de integratiefout zoals volgt geschat worden. Als we (39) van (40) aftrekken kan C_f geschat worden als

$$C_f = \frac{I_{2m}(f) - I_m(f)}{\frac{3}{4}h^2} + O(h)$$

en vervolgens de integratiefout door

$$I(f) - I_{2m}(f) \approx \frac{I_{2m}(f) - I_m(f)}{3}. \tag{41}$$

Merk op: De schatting in (41) hangt alleen van twee opeenvolgende benaderingen van $I(f)$ af. M.a.w. om deze schatting te krijgen hoeft men de exacte waarde van de integraal niet te kennen maar benaderingen hiervan. Tegelijkertijd kan men de schatting verkrijgen niet voor maar pas na het berekenen van de benadering. Daarom heet deze schatting "a posteriori".

Praktisch kan (41) ook als stopcriterium gebruikt worden zoals volgt: zij TOL de maximaal toegestane fout bij de benadering van $I(f)$, nl. er moet gelden $|I(f) - I_m(f)| < TOL$. Dan moet de rij benaderingen $I_m(f)$, $I_{2m}(f)$, ..., $I_{2^k m}(f)$ bepaald worden totdat $|I_{2^k m}(f) - I_{2^{k+1} m}(f)| < 3TOL$. Dan is te verwachten dat $I_{2^{k+1} m}(f)$ de integraal $I(f)$ benaderd met de gewenste tolerantie TOL.

143. Het criterium van boven hangt van de methode af. Bedenk een gelijkaardig stopcriterium voor de samengestelde Simpson-regel.

144. *Supplementair*: Gebruik de 4^e graads Hermite interpolerende veelterm om een kwadratuurformule op te stellen van de vorm

$$\int_a^b f(x)dx = \alpha_0 f(a) + \gamma f\big(\frac{a+b}{2}\big) + \beta_0 f(b) + \alpha_1 f'(a) + \beta_1 f'(b).$$

Toon aan dat de nauwkeurigheidsgraad ervan 5 is en construeer de samengestelde regel.

Les 10

Niet-lineaire vergelijkingen

In deze les worden er methoden behandeld voor de benadering van oplossingen van niet-lineaire vergelijkingen, $f(x) = 0$ voor een continue functie $f : \mathbb{R} \to \mathbb{R}$.

Theorie

Deze resultaten zijn voorbereidend op de analyse van de methoden die later worden uitgelegd.

145. Bestudeer blz. 247 (zie ook hoofdstuk 2, paragraaf 15 in het boek van A. Bultheel). Leer de begrippen *convergentie* (lokaal/globaal), *convergentieorde* en *convergentiefactor*.

146. Geef een bewijs voor de volgende:
Stelling. Zij $x_k, k \in \mathbb{N}$ een rij benaderingen van een nulpunt x^* van de functie f. Veronderstel dat de rij convergeert met de convergentieorde 1 en een convergentiefactor $\rho \in (0, 1)$. Toon aan dat voor elke $\varepsilon \in (0, 1 - \rho)$ geldt: er bestaat $C_\varepsilon > 0$ z.d. voor alle $k \in \mathbb{N}$ geldt
$$|x_k - x^*| < C_\varepsilon(\rho + \varepsilon)^k.$$
Leg uit waarom dit ook meteen de convergentie impliceert.

147. Een manier om het nulpunt x^* van de functie f te benaderen is de functie f zelf te benaderen met een functie $\tilde{f}$ waarvoor nulpunten makkelijker te berekenen zijn. Voorbeelden hiervan zijn interpolerende veeltermen (meest gebruikelijk zijn lineaire of kwadratische veeltermen) of interpolerende splines. Veronderstel dat er geldt $\|f - \tilde{f}\|_\infty < \varepsilon$ (het maximum van het verschil $f - \tilde{f}$ over het definitiegebied). De verwachting is dat het nulpunt $\tilde{x}$ van $\tilde{f}$ een goede benadering is van x^*. Deze bewering wordt hieronder preciezer gemaakt, nl. we bestuderen hoe $|\tilde{x} - x^*|$ geschat kan worden in termen vam ε. Daarvoor herhalen we het begrip multipliciteit van een nulpunt:
Definitie. Zij $f : \mathbb{R} \to \mathbb{R}$ een gegeven functie in $C^p(\mathbb{R})$ en x^* een nulpunt van f. x^* heeft multipliciteit $m \leq p$ als geldt
$$f^{(j)}(x^*) = 0 \text{ voor alle } j \in \{0, \ldots, m-1\} \text{ en } f^{(m)}(x^*) \neq 0.$$

148. Geef een bewijs voor de volgende:
Stelling. Zij $f \in C^p(\mathbb{R})$ $(p \geq 1)$ en x^* een nulpunt met multipliciteit $m \leq p$. Veronderstel dat er een $C_f > 0$ bestaat z.d. $|f^{(m)}(x)| \geq C_f > 0$ voor alle $x \in \mathbb{R}$. Zij $\tilde{f} \in C(\mathbb{R})$ een benadering van f z.d. $\|f - \tilde{f}\|_\infty < \varepsilon$ voor een zekere $\varepsilon > 0$. Dan geldt
$$|x^* - \tilde{x}| \leq \left(\frac{m!}{C_f}\,\varepsilon\right)^{\frac{1}{m}}. \tag{42}$$

Hint: Je hebt enerzijds $f(\tilde{x}) = f(\tilde{x}) - \tilde{f}(\tilde{x})$ met $|f(\tilde{x}) - \tilde{f}(\tilde{x})| < \varepsilon$. Anderzijds kun je voor $f(\tilde{x})$ een Taylor benadrring gebruiken rond x^*.

149. Het bovenstaande resultaat geeft een indicatie over de stabiliteit van de methode. Leg uit waarom het geval $m > 1$ minder stabiel is dan $m = 1$.

150. *Supplementair*: Bestudeer hoofdstuk 6, paragraaf 1 (zie ook Deel II, hoofdstuk 2, paragraaf 18 in het boek van A. Bultheel).

Vast punt/substitutie methoden

Zij I een gesloten interval van $\mathbb{R}$ (eindig of oneindig) en $f \in C(I)$ een gegeven functie met nulpunt x^*. Merk op dat voor elke $\alpha \neq 0$ kan de functie $\Phi \in C(I)$ gedefinieerd worden als

$$\Phi(x) = x + \alpha f(x). \tag{43}$$

Dan is x^* een *vast punt* van Φ, nl. er geldt $\Phi(x^*) = x^*$. Sterker nog, x^* is een nulpunt van f a.e.s.a. x^* vast punt van Φ is. De benadering van nulpunten van f is gelijk aan de benadering van vaste punten van Φ (merk op, hier is de keuze $\alpha \neq 0$ essentieel!). Daarom is het zinvol om zgn. *vast punt methoden* te bestuderen. Hier beperken we de analyse tot de simpelste methode die gedefinieerd is als

$$x_{k+1} = \Phi(x_k), k \geq 0,$$

met x_0 een gegeven startwaarde. De voor de hand liggende vraag is of deze iteratie wel convergeert en hoe snel.

De resultaten in deze paragraaf gelden voor *zelfafbeeldingen*, d.w.z. functies $\Phi : M \to M$ (met M een gegeven verzameling).

151. *Existentiestelling*: Zij $a, b \in \mathbb{R}$, $a < b$ en $I = [a, b]$. Verder zij $\Phi : I \to I$ een continue functie. Dan heeft Φ ten minste een vast punt in I.

Bewijs: Definieer de functie $g : I \to \mathbb{R}$, $g(x) = \Phi(x) - x$. Merk op dat g continu is en dat $g(a) \geq 0$ en $g(b) \leq 0$. Als $g(a) = 0$ dan is a een vast punt van Φ, anders geldt $g(a) > 0$. Analoog, als $g(b) = 0$ dan is b een vast punt van Φ, anders geldt $g(b) < 0$. Als $g(a)g(b) \neq 0$ dan wisselt g van teken ten minste een keer in het inwendige van I. Omdat g continu is moet deze functie een nulpunt hebben en dat is meteen een vast punt voor Φ.

152. Bestudeer paragraaf 6.3 (zie ook deel II, hoofdstuk 2.14 in het boek van A. Bultheel). Leg het verschil uit tussen *Lipschitz continuïteit* en *contractie*. Merk op dat in de stelling van boven het interval $I = [a, b]$ **gesloten** is. Dit is echter noodzakelijk. Geef een voorbeeld van een interval I dat niet gesloten is en een continue *zelfafbeelding* (d.w.z. $\Phi : I \to I$) die geen vast punt heeft. Veronderstel verder dat de functie Φ

continu differentiëerbaar is. Onder welke voorwaarden is Φ Lipschitz continu en wat is de Lipschitz constante van Φ?

153. Geef een bewijs voor het volgende eenduidigheidsresultaat:
Zij I een interval in $\mathbb{R}$ (begrensd of niet, gesloten of niet) en $\Phi : I \to I$ een contractie. Dan heeft Φ ten hoogste één vast punt in I.

154. Het belangrijkste resultaat voor vaste punten betreft contracties. Dit resultaat is de onderstaande
Stelling (Vast-punt-stelling van Banach). Zij I een gesloten interval en $\Phi : I \to I$ een contractie met de constante $L \in (0,1)$. Dan geldt

1. Φ heeft een unieke vast punt $x^* \in I$.
2. Voor elk startpunt $x_0 \in I$ is de substitutiemethode $x_{k+1} = \Phi(x_k)$ $(k \in \mathbb{N})$ convergent naar x^* en gelden er de (a-priori en a-posteriori) foutschattingen

$$|x_k - x^*| \leq \frac{L^k}{1-L}|x_1 - x_0|, \quad \text{resp.} \quad |x_k - x^*| \leq \frac{L}{1-L}|x_k - x_{k-1}|.$$

Merk op: De foutschatting laat zien dat hoe kleiner $L < 1$ hoe sneller de vast punt iteratie convergeert.

Bewijs 1. Eerst laten we zien dat $\{x_k, k \in \mathbb{N}\}$ een cauchyrij is. Merk op dat

$$|x_{k+1} - x_k| = |\Phi(x_k) - \Phi(x_{k-1})| \leq L|x_k - x_{k-1}| \leq \ldots L^k|x_1 - x_0|,$$

voor alle $k \in \mathbb{N}$. Zij $k, p \in \mathbb{N}$. Er geldt

$$\begin{aligned} |x_{k+p} - x_k| &\leq |x_{k+p} - x_{k+p-1}| + |x_{k+p-1} - x_{k+p-2}| + \ldots |x_{k+1} - x_k| \\ &\leq L^{k+p-1}|x_1 - x_0| + \ldots L^k|x_1 - x_0| = L^k \tfrac{1-L^p}{1-L}|x_1 - x_0|. \end{aligned}$$

Omdat $L \in (0,1)$ geldt $|x_{k+p} - x_k| \leq L^k \frac{1}{1-L}|x_1 - x_0|$. Toon aan dat voor elke $\varepsilon > 0$ bestaat er een k_ε z.d. $|x_{k+p} - x_k| < \varepsilon$ voor alle $k \geq k_\varepsilon$ en $p \in \mathbb{N}$. M.a.w. de rij is een cauchyrij.

Omdat I gesloten is bestaat er $x^* \in I$ z.d. $\lim_{k\to\infty} x_k = x^*$. De volgende aspecten zijn zelfstudie opdrachten:

a.) Toon aan dat x^* een vast punt is van Φ (denk aan de continuïteit van Φ!).
b.) Toon aan dat Φ geen twee vaste punten kan hebben (gebruik de contractie-conditie, zie ook punt 153).

2. Zij $k \in \mathbb{N}$ en $e_k = |x_k - x^*|$ (de absolute waarde van de fout). Omdat $x_{k+1} = \Phi(x_k)$ en $x^* = \Phi(x^*)$ geldt

$$e_k \leq |x_k - x_{k+1}| + e_{k+1} \leq |x_k - x_{k+1}| + Le_k \leq L^k|x_1 - x_0| + Le_k.$$

Hier hebben we ook de contractie eigenschap gebruikt. Beide foutschattingen volgen nu.

155. In de bovenstaande stelling moet I gesloten zijn. Geef een voorbeeld van een niet gesloten interval I en een contractie $\Phi : I \to I$ die geen vast punt in I heeft.

156. De contractie Φ moet een zelfafbeelding zijn. Geef een voorbeeld van een gesloten interval I en een contractie $\Phi : I \to \mathbb{R}$ die geen vast punt in I heeft.

157. Zij $f : \mathbb{R} \to \mathbb{R}$ een C^1 functie waarvoor constanten $M, m > 0$ bestaan z.d.

$$0 < m = \min_{x \in \mathbb{R}} |f'(x)| \leq \max_{x \in \mathbb{R}} |f'(x)| = M < \infty$$

voor alle $x \in \mathbb{R}$.

a. Toon aan dat er een $\alpha \in \mathbb{R}$ bestaat z.d. de functie $\Phi(x) = x + \alpha f(x)$ een contractie is.

b. De foutschattingen voor de vast punt iteratie tonen aan dat hoe kleiner de contractie constante is hoe sneller de convergentie van de iteratie. Is er een optimale waarde α_{opt} die de snelste convergentie garandeerd?

Hint: Zoek α z.d. er geldt $-1 < \Phi' < 1$. Om de optimale waarde te verkrijgen zoek α z.d. de maximale ondergrens en de minimale bovengrens van Φ' symmetrisch zijn, d.w.z. $-L \leq \Phi' \leq L$ voor een zekere $L < 1$.
Merk op: Dit is een voorbeeld van hoe de substitutiemethode gebruikt kan worden om de nulpunten van een functie te benaderen.

158. We willen het snijpunt van de functies $y = x$ en $y = \cos x$ benaderen.

a. Bewijs dat de functie $f(x) = \cos(x)$ geen contractie is op $\mathbb{R}$.

b. Toon aan dat f wel een contractie is op het interval [0, 1].
Hint: gebruik de middelwaardestelling.

159. Zij $f : [1, 2] \to \mathbb{R}$, $f(x) = x^3 + 4x^2 - 10$. Deze functie heeft een nulpunt $x^* = 1.3652300134...$ in het gegeven interval. Zoals boven kan x^* tevens vast punt van een functie Φ zijn. Merk op, de keuze van α is niet eenduidig.

a. Geef een conditie waaraan α moet voldoen z.d. Φ een contractie is.

b. Is er een optimale waarde van α (d.w.z. de waarde die de snelste convergentie geeft)?

c. Kies een zekere α die voldoet aan de contractie conditie en pas de vast punt menthode toe om x^* te benaderen met de absolute fout kleiner dan $TOL = 10^{-8}$. Hoeveel iteraties zijn hiervoor nodig? Komt het resultaat overeen met de a-priori en a-posteriori schattingen? Als er een optimale α_{opt} bestaat, herhaal deze procedure voor α_{opt}. Vergelijk de resultaten (aantal iteraties).

160. De functie $f : (0, \infty) \to \mathbb{R}$, $f(x) = x + \ln(x)$ heeft een nulpunt x^* in het interval $[1/2, 3/5]$. Bestudeer de volgende iteratie methoden: zijn deze correct (d.w.z. in geval van convergentie is de limiet x^*), is er sprake van een contractie, welke zal sneller convergeren?

   a. $x_{k+1} = \frac{1}{2}x_k - \frac{1}{2}\ln(x_k)$;
   b. $x_{k+1} = \frac{1}{2}x_k - \ln(x_k)$;
   c. $x_{k+1} = -\ln(x_k)$;
   d. $x_{k+1} = e^{-x_k}$;
   e. $x_{k+1} = \frac{1}{2}x_k + \frac{1}{2}e^{-x_k}$;
   f. $x_{k+1} = \frac{1}{2}x_k + e^{-x_k}$.

161. Maak oefening 6 op blz. 283.

162. Maak oefening 7 op blz. 284.

Bisectie-methode

Dit is een simpele methode die toepasbaar is voor elke continue functie $f : I \to \mathbb{R}$ waarvoor geldt dat $f(a)f(b) < 0$ voor twee punten $a < b$ in het interval I. De convergentie van de methode is gegarandeerd.

163. Bestudeer paragraaf 6.2.1. (zie als alternatief ook paragraaf 2 in het boek van A. Bultheel).

164. Geef een bewijs voor de volgende:
Stelling. Zij I een interval en $a, b \in I$ z.d. $f(a)f(b) < 0$.

   1. De bisectie-methode is convergent met convergentieorde 1 en convergentiefactor 1/2.
   2. Zij x_k de rij van benaderingen in de bisectie-methode en x^* het nulpunt dat de limiet van x_k is. Als $f \in C^1(I)$ en er bestaan $m, M > 0$ z.d. geldt $0 < m \leq |f'(x)| < M$ voor alle $x \in I$ dan geldt

$$\frac{|f(x_k)|}{M} \leq |x_k - x^*| \leq \frac{|f(x_k)|}{m}, \qquad \text{voor alle } k \geq 0.$$

165. Beschouw het voorbeeld bij punt 159. Probeer nu het nulpunt van f te bepalen met behulp van de bisectie-methode. Hoeveel stappen zijn er nu nodig om een absolute fout onder $TOL = 10^{-8}$ te krijgen? Komt dit overeen met de foutschatting?

Secant-methode

166. Bestudeer paragraaf 6.2.2 tot "Regula falsi" op blz. 254 (zie als alternatief ook paragraaf 3 in het boek van A. Bultheel) . Hoeveel nieuwe functie evaluaties zijn er nodig in elke iteratiestap?

167. *Info*:
Voor de substitutie-methode kan men de zgn. *lokale convergentie* bewijzen. M.a.w. de convergentie is alleen gegarandeerd als de startwaarden dicht genoeg zijn bij het nulpunt. Er geldt
Stelling. Zij $f \in C^2[a,b]$ met $x^* \in [a,b]$-nulpunt voor f. Veronderstel dat er $m, M \in (0,\infty)$ bestaan z.d.

$$0 < m \leq |f'(x)|, \quad \text{en} \quad |f''(x)| \leq M, \quad \text{voor alle } x \in [a,b].$$

Zij verder $\rho < \frac{2m}{M}$ en veronderstel dat voor de startpunten geldt $x_0, x_1 \in (x^* - \rho, x^* + \rho)$. Dan is de secant-methode convergent en er gelden de (a-priori en a-posteriori) schattingen

$$|x_k - x^*| \leq \frac{2m}{M}\left(\frac{M\rho}{2m}\right)^{\gamma_k}, \quad |x_k - x^*| \leq \frac{|f(x_k)|}{m} \leq \frac{M}{2m}|x_k - x_{k-1}||x_{k-1} - x_{k-2}|,$$

met γ_k de *Fibonacci* getallen.

Merk op: Als de functie f voldoet aan de voorwaarden in de stelling van boven dan is deze ook monotoon en heeft dus ook maximaal één nulpunt. Verder merk op dat de convergentie alleen is gegarandeerd als de startwaarden dicht genoeg bij het nulpunt liggen.

168. *Info*:
De Fibonacci getallen γ_k zijn lineaire combinaties van $\left(\frac{1\pm\sqrt{5}}{2}\right)^k$ en hangen van de startwaarden γ_0 en γ_1 af. In dit geval geldt $\gamma_0 = \gamma_1 = 1$. Voor grote k geldt $\gamma_k \approx \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^{k+1}$. Daarom is de convergentieorde van de secant-methode $\frac{1+\sqrt{5}}{2}$. M.a.w. de convergentie van de secant-methode is sneller dan die van de bisectie-methode. Daarentegen moeten de startwaarden dicht genoeg bij het nulpunt liggen.

169. *Info*:
Een simpele manier om de convergentieorde γ te schatten is gebaseerd op de aanname dat voor de fouten $e_k = |x_k - x^*|$ geldt $e_{k+1} \approx Ce_k^\gamma$ voor een zekere $C > 0$ en dat er geldt $e_{k+1} \approx \tilde{C}e_k e_{k-1}$ voor een constante $\tilde{C}$ die niet afhangt van k (vergelijk dit met de a-posteriori schatting). Gebruik dit in de a-posteriori schatting zoals gegeven in de convergentiestelling voor de secant-methode om aan te tonen dat $\gamma = \frac{1+\sqrt{5}}{2}$.
N.B.: In het geval van convergentie is de convergentieorde een positief getal.

170. Beschouw alweer het voorbeeld bij punt 159. Probeer nu het nulpunt van f te bepalen met behulp van de secant-methode. Hoeveel stappen zijn er nu nodig om een

(absolute) fout onder $TOL = 10^{-8}$ te krijgen? Komt dit overeen met de foutschatting bij punt 167?

171. Implementeer het algoritme voor de secant-methode in MATLAB. Gebruik deze code om het nulpunt van de functie $f : [0.5, 1.5] \to \mathbb{R}$,

$$f(x) = 5\sin^2(x) - 8\cos^5(x)$$

te benaderen. Controleer hierbij ook de convergentieorde.

172. *Supplementair*: Bestudeer verder paragraaf 6.2.2 tot "Newton" op blz. 255 (zie als alternatief ook paragraaf 4 in het boek van A. Bultheel). Regula falsi is een combinatie van bisectie- en secant-methodes waardoor de afstand $|x_k - x^*|$ steeds kleiner wordt en dus de benadering ook stabiel is. Het nadeel is dat de methode i.h.a. langzamer convergeert dan de secant-methode.

173. *Supplementair*: Implementeer in MATLAB het algoritme voor de "regula falsi"-methode. Gebruik deze code om het nulpunt van de functie $f : [0.5, 1.5] \to \mathbb{R}$,

$$f(x) = 5\sin^2(x) - 8\cos^5(x)$$

te benaderen met een absolute fout kleiner dan 10^{-8}. Controleer hierbij ook de convergentieorde.

Newton-methode

Deze methode heeft een kwadratische convergentie en is dus de snelste van alle methoden die we tot nu toe hebben gezien.

174. Bestudeer paragraaf 6.2.2 tot het eind (zie als alternatief ook paragraaf 6 in het boek van A. Bultheel). Merk op dat in elke iteratiestap er twee functie evaluaties nodig zijn: $f(x_k)$ en $f'(x_k)$.

175. Zoals voor de secant-methode kan men de convergentie van de Newton-methode alleen lokaal bewijzen, nl. als de startwaarde dicht genoeg bij het nulpunt ligt. Er geldt
Stelling. Zij $f \in C^2[a, b]$ met $x^* \in [a, b]$-nulpunt voor f. Veronderstel dat er $m, M \in (0, \infty)$ bestaan z.d.

$$0 < m \leq |f'(x)|, \quad \text{en} \quad |f''(x)| \leq M, \quad \text{voor alle } x \in [a, b].$$

Zij verder $\rho < \frac{2m}{M}$ z.d. $B_\rho := (x^* - \rho, x^* + \rho) \subset [a, b]$ en veronderstel dat voor het startpunt geldt $x_0 \in B_\rho$. Dan is de Newton-methode convergent (naar x^*) en er gelden de (a-priori en a-posteriori) schattingen

$$|x_k - x^*| \leq \frac{2m}{M}\left(\frac{M\rho}{2m}\right)^{2^k}, \quad |x_k - x^*| \leq \frac{|f(x_k)|}{m} \leq \frac{M}{2m}|x_k - x_{k-1}|^2.$$

Merk op: Onder de gegeven voorwaarden is de functie f monotoon en heeft dus slechts één nulpunt. Verder is de convergentie gegarandeerd alleen als de startwaarden dicht genoeg bij het nulpunt liggen.

Bewijs: We tonen eerst aan dat alle punten x_k in het interval B_ρ blijven en vervolgens dat de methode convergeert. Hiervoor gebruiken we de volgende resultaten. Voor elke $x, y \in B_\rho$ geldt

$$\begin{array}{l} |f(x) - f(y)| \geq m|x-y|, \quad \text{en} \\ f(y) = f(x) + (y-x)f'(x) + (y-x)^2 R(y;x), \quad \text{met} \\ \qquad R(y;x) = \int_0^1 f''(x + s(y-x))(1-s)ds. \end{array} \tag{44}$$

Opdracht: Geef een bewijs voor (44)!

Omdat $|f''(x)| \leq M$ geldt $|R(y;x)| \leq \frac{M}{2}$.

Beschouw de functie $g : B_\rho \to \mathbb{R}$, $g(x) = x - \frac{f(x)}{f'(x)}$. Omdat $|f'(x)| \geq m$ en $f(x^*) = 0$ geldt voor elke $x \in B_\rho$

$$|g(x) - x^*| = \left| x - x^* - \frac{f(x)}{f'(x)} \right| \leq \frac{1}{m} \left| f(x^*) - f(x) - (x^* - x)f'(x) \right|.$$

Uit de ongelijkheid in (44) en omdat $|R(x^*;x)| \leq \frac{M}{2}$, $|x^* - x| < \rho$ en $\rho < \frac{2m}{M}$ volgt

$$|g(x) - x^*| \leq \frac{1}{m}(x^* - x)^2 |R(x^*;x)| \leq \frac{\rho^2 M}{2m} < \rho. \tag{45}$$

M.a.w. $g(x) \in B_\rho$. Merk op dat de Newton-iteratie geschreven kan worden als $x_{k+1} = g(x_k)$ en daarom is de iteratie ten eerste wel gedefinieerd en ten tweede blijft de rij benaderingen altijd in het interval B_ρ.

Het bewijs voor de convergentie volgt dezelfde ideëen. Er geldt

$$|x_{k+1} - x^*| = |g(x_k) - x^*| \leq \frac{1}{m}(x_k - x^*)^2 |R(x^*;x_k)| \leq \frac{M}{2m}(x_k - x^*)^2.$$

Met $e_k := |x_k - x^*|$ kan het bovenstaande herschreven worden als

$$\frac{M}{2m} e_{k+1} \leq \left(\frac{M}{2m} e_k \right)^2 \leq \left(\frac{M}{2m} e_{k-1} \right)^4 \leq \cdots \leq \left(\frac{M}{2m} e_0 \right)^{2^{k+1}}.$$

Omdat $e_0 = |x_0 - x^*| < \rho$ is $\frac{M}{2m} e_0 < 1$ en de convergentie alsmede de a-priori foutschatting volgen.

Voor de a-posteriori foutschatting gebruiken we de ongelijkheid in (44). Omdat $f(x^*) = 0$ geldt

$$\begin{array}{rcl} |x_k - x^*| & \leq & \frac{1}{m}|f(x_k)| = \frac{1}{m}|f(x_{k-1}) + (x_k - x_{k-1})f'(x_{k-1}) + (x_k - x_{k-1})^2 R(x_k; x_{k-1})| \\ & = & \frac{(x_k - x_{k-1})^2}{m} |R(x_k; x_{k-1})|. \end{array}$$

Hier hebben we de gelijkheid in (44) alsmede de definitie van de iteratie gebruikt. De rest volgt uit de bovengrens voor R.

176. Implementeer in MATLAB het Newton-algoritme. Gebruik deze code om het nulpunt van de functie $f : [0.5, 1.5] \to \mathbb{R}$,

$$f(x) = 5\sin^2(x) - 8\cos^5(x)$$

te benaderen met een absolute fout kleiner dan 10^{-8}. Controleer hierbij ook de convergentieorde.

177. De Newton-methode is gebruikt voor computerbenaderingen van $a^{\frac{1}{n}}$ ($a > 0$ en $n \in \mathbb{N}$). Merk op dat $a^{\frac{1}{n}}$ het positieve nulpunt is van $f : [0, \infty) \to \mathbb{R}$, $f(x) = x^n - a$. Hoewel volgens de stelling bij punt 175 de convergentie alleen gegarandeerd is wanneer de startwaarde dicht bij het nulpunt ligt, is in dit geval de methode convergent voor elke starwaarde. Hieronder is $\{x_k, k \in \mathbb{N}\}$ de rij Newton-iteraties, $x_{k+1} = x_k - f(x_k)/f'(x_k)$.

    a. Toon aan dat f stijgend en convex is.
    b. Toon aan dat als $x_k > a^{\frac{1}{n}}$ dan geldt $a^{\frac{1}{n}} < x_{k+1} < x_k$.
    c. Toon aan dat als $x_0 \in (0, a^{\frac{1}{n}})$ dan geldt $x_1 > a^{\frac{1}{n}}$.
    d. Gebruik deze stappen om aan te tonen dat de rij $\{x_k, k \in \mathbb{N}\}$ begrensd is en ook monotoon voor $k \geq 1$. Bepaal vervolgens de limiet van x_k (een ϵ-δ argument is niet nodig).
    e. Schrijf een MATLAB programma om $2^{1/4}$ te benaderen met de absolute fout $TOL = 10^{-8}$. Gebruik de a-priori foutschatting om het aantal iteraties te voorspellen en vervolgens de a-posteriori foutschatting als stopcriterium.

178. Als het nulpunt x^* de multipliciteit m heeft met $m > 1$ dan kan de Newton-methode minder goed convergeren (of helemaal niet). In dit geval zijn er twee mogelijkheden te bedenken:

    a. x^* is een simpel nulpunt voor $h = f^{(m-1)}$ en kan de Newton-methode dus toegepast worden voor h.
    b. Alternatief: voor elke x bestaat er een $\xi(x)$ z.d. $f(x) = \frac{(x-x^*)^m}{m!} f^{(m)}(\xi(x))$. Dan geldt ook $f'(x) = \frac{(x-x^*)^{m-1}}{(m-1)!} f^{(m)}(\xi(x)) + \frac{(x-x^*)^m}{m!} f^{(m+1)}(\xi(x))\xi'(x)$. Toon nu aan dat als $f'(x_k) \neq 0$ geldt asymptotisch (d.w.z. als $x_k \to x^*$):

$$x_{k+1} - x^* = O((x_k - x^*)^2), \text{ voor de methode } x_{k+1} = x_k - x^* - m\frac{f(x_k)}{f'(x_k)}$$

    respectievelijk $x_{k+1} - x^* = O((x_k - x^*))$ voor de standaard Newton-methode.

Implementeer de twee varianten in MATLAB en gebruik deze om $x^* = 1$, het dubbele nulpunt van $f(x) = x^3 - x^2 - x + 1$ te benaderen met de absolute fout $TOL = 10^{-8}$. Vergelijk de resultaten voor de twee algoritmen en voor de standaard Newton-methode.