

Numerieke Wiskunde voor technici

ir. J. van Kan

Delft University Press

CIP-gegevens Koninklijke Bibliotheek, Den Haag

Kan, J. van

Numerieke wiskunde voor technici / J. van Kan. - Delft :
Delftse Universitaire Pers. — III.

Uitg. in opdracht van: Vereniging voor Studie- en
Studentenbelangen. - 1e dr.: Delft : Delftse U.M., 1988.

- Met lit. opg., reg.

ISBN 90-407-1151-8

Trefw.: numerieke wiskunde.

© VSSD

Eerste druk 1988

Derde druk 1996, 2000

Uitgegeven door:

Delft University Press

Postbus 98, 2600 MG Delft

tel. 015 278 3254, telefax 015 278 1661, e-mail info@library.tudelft.nl

website: <http://www.library.tudelft.nl/dup>

In opdracht van:

Vereniging voor Studie- en Studentenbelangen te Delft

Poortlandplein 6, 2628 BM Delft

tel. 015 - 2782124, telefax 015 - 2787585, e-mail: hlf@vssd.nl

internet: <http://www.vssd.nl/hlf>

Alle rechten voorbehouden. Niets uit deze uitgave mag worden verveelvoudigd, opgeslagen in een geautomatiseerd gegevensbestand, of openbaar gemaakt, in enige vorm of op enige wijze, hetzij elektronisch, mechanisch, door fotokopieën, opnamen, of op enige andere manier, zonder voorafgaande schriftelijke toestemming van de uitgever.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher.

ISBN 90-407-1151-8

Voorwoord

Het gebruik van numerieke methoden door technici beperkt zich in de praktijk vaak door het aanroepen van subroutines uit een bestaand programmapakket. Daar dit vaak 'black-box' pakketten zijn is het noodzakelijk dat men uit de resultaten zelf kan zien of de vereiste nauwkeurigheid gehaald is en of het probleem gevoelig is voor verstoringen. Om deze reden wordt in dit boek naast de uiteenzetting van de numerieke methoden ook ruimschoots aandacht besteed aan zaken als foutenanalyse, stabiliteit en conditie.

De methoden worden gepresenteerd in een technische context in de verwachting dat dit motivatieverhogend zal werken.

Dit boek bevat (in iets gewijzigde vorm) de stof zoals die in het college Numerieke Analyse C1 (a105) aan de TU Delft wordt onderwezen. Het bevat als onderwerpen gewone differentiaalvergelijkingen en numerieke lineaire algebra. Elementaire onderwerpen als interpolatie, numerieke integratie en nulpuntsbepaling worden bekend verondersteld. In die zin bouwt het voort op Analyse door J.H.J. Almering e.a., ook bij deze uitgever verschenen.

De schrijver houdt zich aanbevolen voor opmerkingen.

Augustus 1988

J. van Kan

Opmerking bij de derde druk

Naast enkele kleine verbeteringen zijn in deze nieuwe druk twee bijlagen toegevoegd.

Januari 1996

J. van Kan

Inhoud

VOORWOORD	5
DEEL I	
GEWONE DIFFERENTIAALVERGELIJKINGEN	
1 BEGINWAARDENPROBLEMEN	13
1.1. Inleiding	13
1.1.1. Type probleemstelling	13
1.1.2. Begin(voor)waardenproblemen	14
1.1.3. Randwaardenproblemen	14
1.2. Numerieke integratie	15
1.2.1. Analytische en numerieke oplossing	15
1.2.2. Principe van een numerieke oplosmethode	15
1.2.3. Het begrip orde (O)	18
1.2.4. Lokale afbreekfout	19
1.2.5. Hogere-orde methoden	22
1.2.6. De methode van Runge-Kutta	23
1.3. Stabiliteit	25
1.3.1. Versterkingsfactor	26
1.3.2. Conclusies	30
1.3.3. Het belang van stabiliteit van numerieke processen	30
1.3.4. Toepasbaarheid op $y' = f(x,y)$	32
1.4. Globale fout	32
1.4.1. Schatting van de fout in de praktijk	34
1.5. Stelsels eerste-orde beginwaardenproblemen	35
1.5.1. Algemene gedaante	35
1.5.2. Numerieke methoden voor stelsels	36
1.5.3. Hogere-orde beginwaardenproblemen	38
1.6. Stabiliteit van numerieke methoden voor stelsels vergelijkingen	40
1.6.1. Versterkingsmatrix	42
1.6.2. Het algemene geval	45
1.6.3. Stabiliteit en nauwkeurigheid	46
1.6.4. Impliciete methoden	47

DEEL II

NUMERIEKE LINEAIRE ALGEBRA

2	HET OPLOSSEN VAN STELSELS LINEAIRE VERGELIJKINGEN	53
2.1.	Inleiding	53
2.2.	Probleemstelling	53
2.2.1.	Geheugenruimte en rekentijd	54
2.2.2.	Hoe het niet moet (I)	54
2.2.3.	Hoe het niet moet (II)	55
2.3.	Gauss-eliminatie	55
2.3.1.	Floating-point aritmetiek	60
2.3.2.	Pivotstrategieën	61
2.4.	LU decompositie	62
2.4.1.	Verband tussen L , U en A	64
2.4.2.	De inverse in LU -vorm	65
2.4.3.	Varianten van de LU decompositie	68
2.4.4.	Varianten voor symmetrische matrices	68
2.4.5.	Choleski decompositie	69
2.5.	Conditie	73
2.5.1.	Relatie tussen conditie en grootte van de determinant?	74
2.5.2.	Relatie tussen conditie en eigenwaarden	74
2.5.3.	Eigenschappen van de norm	75
2.5.4.	Het verstoorde systeem	75
2.5.5.	Numerieke singulariteit	77
2.5.6.	Praktische bepaling van de conditie	77
	Toepassingen	
2.6.	Lineaire randwaarden problemen	77
2.6.1.	De kabelvergelijking	77
2.6.2.	Differentiemethoden	78
2.6.3.	Globale fout	80
2.6.4.	Conditie	81
2.6.5.	Andere randvoorwaarden	82
2.6.6.	Conditie van het probleem met betrekking tot de randvoorwaarden	84
2.6.7.	De buigende balk	85
2.7.	Kleinste kwadraten methode	87
2.7.1.	n vergelijkingen met m onbekenden ($n > m$)	89
2.7.2.	m -de graadspolynoom door $n + 1$ steunpunten	90
2.7.3.	Conditie	92

3	EIGENWAARDENPROBLEMEN	95
3.1.	Inleiding	95
3.2.	Herhaling van enige begrippen	95
3.2.1.	Een eigenwaardenprobleem uit de techniek	96
3.3.	Numerieke methoden	97
3.3.1.	De powermethode	97
3.3.2.	Praktische uitvoering van de powermethode	99
3.3.3.	Convergentiesnelheid	100
3.3.4.	Grootste eigenwaarden van gelijke modulus	103
3.4.	Hotelling-deflatie	103
3.4.1.	Praktische beperking van de Hotelling deflatie	105
3.4.2.	Het niet-symmetrische geval	105
3.5.	Bandmatrices. Vectordeflatie	107
3.6.	Inverse iteratie	108
3.6.1.	Bandmatrices	109
3.7.	Het gegeneraliseerde eigenwaardenprobleem	109
3.7.1.	Symmetrische A en B	110
3.7.2.	A en B grote bandmatrices	111
3.8.	Toepassingen	111
3.8.1.	De knikkende staaf	111
3.8.2.	Numerieke behandeling	112
	APPENDIX A: INTERPOLATIE EN INTEGRATIE	115
A.1.	Het Taylor polynoom	115
A.2.	Integratie	119
	APPENDIX B: STELLING VAN GERSCHGORIN	124
	LITERATUUR	126
	TREFWOORDEN	127



NUMERIEKE LINEAIRE ALGEBRA

2 | Het oplossen van stelsels lineaire vergelijkingen

2.1. Inleiding

De numerieke lineaire algebra houdt zich bezig met twee hoofdproblemen:

1. het oplossen van stelsels lineaire vergelijkingen;
2. het bepalen van eigenwaarden en eigenvectoren van $n \times n$ matrices.

De technische toepassingen van het eerste hoofdprobleem vindt men onder andere in het numeriek oplossen van lineaire randwaardenproblemen (gewone of partiële differentiaalvergelijkingen) en in kleinste kwadraten benaderingen. In dit hoofdstuk zullen hiervan enkele voorbeelden getoond worden.

De technische toepassingen van het tweede hoofdprobleem zijn te vinden in trillings- en knikproblemen. Voorbeelden hiervan komen in hoofdstuk 3 aan de orde.

2.2. Probleemstelling

We formuleren het eerste hoofdprobleem. Zij gegeven het stelsel van n vergelijkingen met n onbekenden.

$$\begin{aligned}
 a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n &= b_1 \\
 a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n &= b_2 \\
 \vdots & \\
 a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n &= b_n
 \end{aligned} \tag{2.1}$$

We zoeken n getallen x_1, x_2, \dots, x_n zó dat aan (2.1) is voldaan. Uit de lineaire algebra kennen we een korte notatie voor (2.1), namelijk zij $\underline{b} \in \mathbb{R}^n$ en A een $n \times n$ matrix dan wordt een $\underline{x} \in \mathbb{R}^n$ gezocht zó dat

$$A\underline{x} = \underline{b}. \tag{2.2}$$

Uit de theorie is een nodige en voldoende voorwaarde bekend dat (2.2) precies één oplossing heet, namelijk dat de determinant van A niet nul is. In formulevorm:

$$\det(A) \neq 0. \tag{2.3}$$

Indien aan (2.3) niet voldaan is, dus indien $\det(A) = 0$, dan heeft (2.2) òf geen enkele oplossing, òf oneindig veel oplossingen, afhankelijk van het rechterlid \underline{b} (zie [2], 6.2.4).

2.2.1. Geheugenruimte en rekentijd

We zullen ons bezighouden met het onderzoeken van oplossingsmethodieken van (2.2) die geschikt zijn voor een computer. Dat wil zeggen dat we onderworpen zijn aan beperkingen ten aanzien van de opslagcapaciteit (het geheugen) en ten aanzien van het aantal rekenkundige bewerkingen (de rekentijd).

Op het eerste gezicht lijkt dat wel mee te vallen: een supercomputer heeft een geheugencapaciteit van 256 M (= $256 \times 2^{20} \approx 256 \times 10^6$) bytes, hetgeen overeenkomt met 32 miljoen dubbele-lengte getallen en een rekensnelheid van 10^9 vermenigvuldigingen per seconde. Echter zowel geheugengebruik als rekentijd kosten in de barre praktijk van de consumptiemaatschappij geld. Het is dus voordelig om zo min mogelijk rekentijd en zo weinig mogelijk geheugen te gebruiken. We zullen daarom onze algoritmen aan die criteria toetsen.

2.2.2. Hoe het niet moet (I)

Als we (2.2) een beetje onzorgvuldig behandelen dan is het mogelijk dat zelfs de hiervoor genoemde capaciteit onvoldoende blijkt. Een mooi voorbeeld is de regel van Cramer. Deze geeft een expliciete uitdrukking voor de oplossing van (2.1):

$$x_i = \frac{\det(B_i)}{\det(A)}. \tag{2.4}$$

Hierin is B_i een matrix waarin alle kolommen gelijk zijn aan die van de matrix A , met uitzondering van de i -de kolom, die vervangen wordt door de vector b . Zouden we met behulp van uitdrukking (2.4) bijvoorbeeld een stelsel van 100 vergelijkingen met 100 onbekenden oplossen, dan moeten we 101 determinanten van de orde 100 uitrekenen. We berekenen die determinanten bijvoorbeeld door ontwikkeling naar de rijen. In dat geval vergt een orde n determinant $n!$ vermenigvuldigingen. In ons geval hebben we dus $101 \times 100! = 101!$ vermenigvuldigingen te verrichten voor het berekenen van de oplossing. Nu is $101! \approx 10^{160}$. Op een supercomputer zou dit 10^{151} seconden vergen. 3×10^7 seconden is ongeveer een jaar, dus het oplossen van (2.2) zou op die manier ongeveer 3×10^{143} jaar kosten! (De geschatte leeftijd van ons zonnestelsel is 10^{15} jaar).

Nu zijn er wel efficiëntere methoden om determinanten uit te rekenen, maar het uitrekenen van één determinant op die manier vraagt evenveel rekenwerk als het oplossen

van een stelsel vergelijkingen van dezelfde orde.

Op zijn best doen we dus voor ons 100×100 -stelsel $101 \times$ zoveel werk als noodzakelijk is. De conclusie is dan ook dat de regel van Cramer in de praktijk volkomen onbruikbaar is om een stelsel vergelijkingen op te lossen. Wel is deze regel van theoretisch belang.

2.2.3. Hoe het niet moet (III)

De theorie leert dat wanneer de matrix A niet singulier is er een eenduidig bepaalde inverse A^{-1} bestaat zó dat $AA^{-1} = A^{-1}A = I$, waarin I de $n \times n$ eenheidsmatrix is. De oplossing van (2.2) wordt dan gegeven door $\underline{x} = A^{-1}\underline{b}$.

Het is echter geen goed idee om (2.2) op te lossen door eerst A^{-1} te bepalen. Immers het bepalen van A^{-1} is equivalent met het oplossen van n stelsels:

$$A\underline{x}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad A\underline{x}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad \dots \quad A\underline{x}_n = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}$$

waarna A^{-1} bekend is:

$$A^{-1} = (\underline{x}_1, \underline{x}_2, \dots, \underline{x}_n).$$

We doen in dit geval weer veel te veel werk. Ook in het geval dat we (2.2) moeten oplossen voor meerdere rechterleden, bijvoorbeeld

$$\begin{aligned} A\underline{x}_1 &= \underline{b}_1 \\ A\underline{x}_2 &= \underline{b}_2 \\ &\vdots \\ A\underline{x}_m &= \underline{b}_m \end{aligned}$$

met $m > n$ is het niet nodig de inverse te bepalen.

Later (bij de LU decompositie) zullen we zien dat er alternatieve mogelijkheden zijn die de voorkeur verdienen. Het bepalen van A^{-1} kan in de praktijk bijna altijd met voordeel vermeden worden.

2.3. Gauss-eliminatie

Een methode die zeer geschikt is voor gebruik op een computer is de eliminatiemethode van Gauss. Het principe daarvan is bekend vanuit de lineaire algebra (zie [2], 3.5.13). We geven hier een enigszins andere behandeling.

Definitie 2.1

Een onderdriehoeksmatrix (aangeduid met de letter L van Lower) is een matrix, waarvan alle elementen *boven* de hoofddiagonaal nul zijn: $l_{ij} = 0$ als $i < j$. Een bovendriehoeksmatrix (aangeduid met de letter U van Upper) is een matrix, waarvan alle elementen *onder* de hoofddiagonaal nul zijn: $u_{ij} = 0$ als $j < i$.

Het zal duidelijk zijn dat wanneer de matrix A in (2.2) een onder- of bovendriehoeksvorm heeft dit stelsel door directe substitutie op te lossen is. \square

Voorbeeld 2.1

Gegeven is het stelsel vergelijkingen:

$$\begin{aligned} x_1 + 2x_2 + 3x_3 &= 8 \\ x_2 + 2x_3 &= 5 \\ 2x_3 &= 4 \end{aligned} \quad \text{of} \quad \begin{bmatrix} 1 & 2 & 3 \\ 0 & 1 & 2 \\ 0 & 0 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 8 \\ 5 \\ 4 \end{bmatrix}.$$

De oplossing volgt door substitutie van beneden naar boven:

$$\begin{aligned} x_3 &= 2 \\ x_2 &= 5 - 2x_3 = 5 - 4 = 1 \\ x_1 &= 8 - 2x_2 - 3x_3 = 8 - 2 - 6 = 0. \end{aligned} \quad \triangle$$

De eliminatiemethode van Gauss vormt het stelsel (2.2) door elementaire rijbewerkingen om tot een stelsel waarvan de matrix een bovendriehoekstructuur heeft. Daartoe gaan we als volgt te werk. We beschouwen de aangevulde matrix $A^{(0)}$ van het stelsel (2.2)

$$A^{(0)} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ a_{21} & a_{22} & \dots & a_{2n} & b_2 \\ \vdots & \vdots & & \vdots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} & b_n \end{bmatrix}.$$

We trekken nu de eerste rij een aantal malen van alle andere rijen af zó dat in de eerste kolom nullen ontstaan vanaf de tweede rij. We definiëren

$$m_{j1} = \frac{a_{j1}}{a_{11}}, \quad j = 2, 3, \dots, n,$$

$$a_{jk}^{(1)} = a_{jk} - m_{j1}a_{1k}, \quad k = 1, \dots, n$$

$$\text{en} \quad b_j^{(1)} = b_j - m_{j1}b_1. \quad (2.5)$$

Men verifieert gemakkelijk dat $a_{j1}^{(1)} = 0$, $j = 2, 3, \dots, n$. Dit leidt tot een nieuwe

aangevulde matrix

$$A^{(1)} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ 0 & a_{22}^{(1)} & \dots & a_{2n}^{(1)} & b_2^{(1)} \\ \vdots & a_{32}^{(1)} & & a_{3n}^{(1)} & b_3^{(1)} \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & a_{n2}^{(1)} & \dots & a_{nn}^{(1)} & b_n^{(1)} \end{bmatrix}.$$

Eenvoudig ziet men in dat het stelsel dat beschreven wordt door deze nieuwe, aangevulde matrix dezelfde oplossing heeft als het oorspronkelijke stelsel (zie ook [2], 3.5.12).

Trek nu de tweede rij een aantal malen van de volgende rijen af zó dat in de tweede kolom nullen ontstaan vanaf de derde rij. Dus:

$$m_{j2} = \frac{a_{j2}^{(1)}}{a_{22}^{(1)}}, \quad j = 3, 4, \dots, n,$$

$$a_{jk}^{(2)} = a_{jk}^{(1)} - m_{j2} a_{2k}^{(1)}, \quad k = 2, \dots, n$$

en
$$b_j^{(2)} = b_j^{(1)} - m_{j2} b_2^{(1)}. \quad (2.6)$$

Dit leidt tot

$$A^{(2)} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} & b_1 \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & \dots & a_{2n}^{(1)} & b_2^{(1)} \\ 0 & 0 & a_{33}^{(2)} & \dots & a_{3n}^{(2)} & b_3^{(2)} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & a_{n3}^{(2)} & \dots & a_{nn}^{(2)} & b_n^{(2)} \end{bmatrix}.$$

De i -de stap van het proces:

$$m_{ji} = \frac{a_{ji}^{(i-1)}}{a_{ii}^{(i-1)}}, \quad j = i+1, i+2, \dots, n,$$

$$a_{jk}^{(i)} = a_{jk}^{(i-1)} - m_{ji} a_{ik}^{(i-1)}, \quad k = i, i+1, \dots, n$$

$$\text{en} \quad b_j^{(i)} = b_j^{(i-1)} - m_{ji} b_i^{(i-1)}. \quad (2.7)$$

Zetten we dit proces voort tot $i = n - 1$ dan resulteert tenslotte:

$$A^{(n-1)} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} & b_1 \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & \dots & a_{2n}^{(1)} & b_2^{(1)} \\ 0 & 0 & a_{33}^{(2)} & \dots & a_{3n}^{(2)} & b_3^{(2)} \\ \vdots & \vdots & 0 & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & a_{nn}^{(n-1)} & b_n^{(n-1)} \end{bmatrix}.$$

Dit is een bovendriehoekssysteem, waarvan de oplossing door directe terugsubstitutie te bepalen is (zie voorbeeld 2.1).

De grootheden m_{ji} heten multiplicatoren, de grootheden $a_{ii}^{(i-1)}$ heten pivots. Daar in de i -de stap door de pivot $a_{ii}^{(i-1)}$ gedeeld moet worden om de multiplicatoren m_{ij} te bepalen (zie (2.7)), is een noodzakelijke en voldoende voorwaarde voor het uitvoerbaar zijn van het Gauss proces dat geen der pivots $a_{ii}^{(i-1)}$ nul is. Alvorens daarover een stelling te formuleren wordt eerst opgemerkt dat het in het Gauss proces is toegestaan om rijen en kolommen te verwisselen. Het verwisselen van rijen komt neer op het veranderen van de volgorde van de vergelijkingen, het verwisselen van kolommen op het omnummeren van de onbekenden.

Stelling 2.1

Indien de rang van A gelijk is aan n ($r_A = n$) of equivalent hiermee indien $\det(A) \neq 0$, dan bestaat er een ordening van de vergelijkingen zó dat geen der pivots $a_{ii}^{(i-1)}$ gelijk is aan 0.

Bewijs

We beschouwen het Gauss proces toegepast op matrix A alleen (dus niet op de aangevulde matrix).

Na k stappen in het proces zijn k rijen ‘behandeld’ en moeten $n - k$ rijen nog behandeld worden. Noem de $(n-k) \times (n-k)$ submatrix die nog behandeld moet worden B_k . Dus:

$$A^{(k)} = \begin{bmatrix} a_{11} & a_{12} & \dots & \dots & \dots & a_{1n} \\ 0 & a_{22}^{(1)} & \dots & \dots & \dots & a_{2n}^{(1)} \\ \vdots & 0 & \dots & \dots & \dots & \vdots \\ \vdots & \vdots & 0 & a_{k+1,k+1}^{(k)} & \dots & a_{k+1,n}^{(k)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & a_{n,k+1}^{(k)} & \dots & a_{n,n}^{(k)} \end{bmatrix}.$$

$$B_k = \begin{bmatrix} a_{k+1,k+1}^{(k)} & \dots & a_{k+1,n}^{(k)} \\ \vdots & & \vdots \\ a_{n,k+1}^{(k)} & \dots & a_{n,n}^{(k)} \end{bmatrix}. \quad (2.8)$$

Daar elementaire rij-operaties de waarde van de determinant niet veranderen geldt
blijkbaar

$$\det(A) = \det(A^{(k)}) = a_{11}^{(1)} \cdot a_{22}^{(2)} \cdot a_{33}^{(3)} \cdot \dots \cdot a_{kk}^{(k-1)} \cdot \det(B_k). \quad (2.9)$$

We bewijzen nu de stelling uit het ongerijmde.

Er is kennelijk een ordening van de vergelijkingen mogelijk zó dat $a_{11} \neq 0$. Immers, was dit niet zo, dan zou de hele eerste kolom van A gelijk zijn aan 0 en dit is in strijd met het gegeven dat $\det(A) \neq 0$. Stel nu dat na k stappen geen verwisseling der resterende rijen mogelijk is, zó dat $a_{k+1,k+1}^{(k)} \neq 0$. Dat betekent dat de hele eerste kolom van B_k nul is, dus $\det(B_k) = 0$. Maar wegens (2.9) is dan ook $\det(A) = 0$, in strijd met het gegeven. \circ

Stelling 2.2

Indien de rang van A gelijk is aan n ($r_A = n$) of equivalent hiermee indien $\det(A) \neq 0$, dan bestaat er een nummering van de onbekenden zó dat geen der pivots $a_{ii}^{(i-1)}$ gelijk is aan nul.

Bewijs

Het bewijs van deze stelling is analoog aan het bewijs van stelling 2.1. \circ

2.3.1. Floating-point aritmetiek

Een computer werkt niet met oneindige precisie. Reële getallen worden gerepresenteerd in de vorm $0.ddd\dots d \times 10^{dd}$. Het eerste stuk noemt men de mantisse, het tweede de exponent (de interne representatie is binair, maar dat doet hier niet ter zake). De lengte van de mantisse hangt af van het fabrikaat van de machine en (soms) van de keuze van de programmeur.

Een normale keuze is een mantisse van 16 cijfers, maar in sommige programmeertalen is een keuze mogelijk tussen 6 of 16 cijfers. Het eerste beperkt natuurlijk het geheugengebruik. Dit type getallen wordt *floating-point getallen* genoemd.

Het werken met dit soort getallen heet enkele nare consequenties: soms kan er informatie verloren gaan.

Voorbeeld 2.2

Werkend met een mantisse van 6 cijfers hebben we de volgende floating-point optelling.

$$\begin{aligned} 102 + 0.25 &= 0.102000 \times 10^3 + 0.250000 \times 10^0 = \\ &= 0.102000 \times 10^3 + 0.000250 \times 10^3 = \\ &= 0.102250 \times 10^3. \end{aligned}$$

Dit resultaat is exact. Met een mantisse van 3 cijfers wordt dit echter:

$$\begin{aligned} 102 + 0.25 &= 0.102 \times 10^3 + 0.250 \times 10^0 = \\ &= 0.102 \times 10^3 + 0.000 \times 10^3 = \\ &= 0.102 \times 10^3. \end{aligned}$$

Het optellen van zo'n kleine grootheid bij een groot getal heeft geen enkel effect.

Beschouw de volgende vectoren:

$$\underline{x}_1 = \begin{bmatrix} 100 \\ 100 \\ 100 \end{bmatrix} \quad \underline{x}_2 = \begin{bmatrix} 0 \\ 0.25 \\ 0 \end{bmatrix} \quad \underline{x}_3 = \begin{bmatrix} 0 \\ 0 \\ 0.25 \end{bmatrix}.$$

\underline{x}_1 , \underline{x}_2 en \underline{x}_3 zijn lineair onafhankelijk. Volgens de theorie zijn dan ook de vectoren $\underline{x}_1 + \underline{x}_2$ en $\underline{x}_1 + \underline{x}_3$ onafhankelijk. Bij floating-point optelling met mantisse van 3 cijfers is echter

$$\underline{x}_1 + \underline{x}_3 = \begin{bmatrix} 100 \\ 100 \\ 100 \end{bmatrix} \quad \text{en} \quad \underline{x}_1 + \underline{x}_2 = \begin{bmatrix} 100 \\ 100 \\ 100 \end{bmatrix}.$$

Deze vectoren zijn gelijk.

△

2.3.2. Pivotstrategieën

In het Gauss-eliminatie proces worden (rij) vectoren floating-point van elkaar afgetrokken. Men wil voorkomen dat door deze floating-point operaties lineaire afhankelijkheid ontstaat, zoals geschetst in voorbeeld 2.2.

Uiteraard is het optreden van lineaire afhankelijkheid fataal, omdat dit zou betekenen dat de determinant van submatrix B_k (zie (2.8)) nul zou worden voor zekere k , zodat het Gauss proces niet kan worden voortgezet. Eén mogelijkheid is om steeds met een mantisse met zoveel mogelijk cijfers te rekenen. In de praktijk ziet men dan ook dat, wanneer een keuzemogelijkheid aanwezig is (zoals in de programmeertaal FORTRAN), men steeds werkt met floating-point getallen met zo groot mogelijke mantisse (zogenaamde *dubbele precisie* of *dubbele lengte getallen*), wanneer men met matrixoperaties te maken heeft.

Een tweede mogelijkheid is ervoor te zorgen dat de multiplicatoren m_{ji} in absolute waarde zo klein mogelijk zijn. Dit vermindert de kans op het optreden van lineaire afhankelijkheid door floating-point aritmetiek. (Beschouw bijvoorbeeld in voorbeeld 2.2 de lineaire (on)afhankelijkheid van $0.1\underline{x}_1 + \underline{x}_2$ en $0.1\underline{x}_1 + \underline{x}_3$ floating-point optellingen met mantisse van 3 cijfers). Daar de multiplicatoren ontstaan na deling door de pivots kiest men de ordening van vergelijkingen en onbekenden zó dat de pivot-elementen steeds zo groot mogelijk zijn.

De twee meest gebruikte methoden zijn:

1. *Partial pivoting.*

Alleen de rijen worden verwisseld. In de restmatrix B_k (zie (2.8)) wordt die vergelijking op de bovenste rij gezet die het in absolute waarde grootste element in de eerste kolom heeft.

2. *Complete pivoting.*

Zowel rijen als kolommen worden verwisseld. Men zoekt het in absolute waarde grootste element van de restmatrix B_k , zeg $a_{ij}^{(k)}$. Nu wisselt men de $(k+1)$ -ste en de i -de rij om, benevens de $(k+1)$ -ste en de j -de kolom. Nu staat het grootste element op de pivot-plaats.

Zowel methode 1 als methode 2 vinden plaats in iedere stap van het Gauss-eliminatie proces en worden pivot strategieën genoemd. In de praktijk blijkt dat partial pivoting net zo goed voldoet als complete pivoting, zodat complete pivoting, dat uiteraard veel meer werk vereist, bijna nooit wordt toegepast.

Voorbeeld 2.3

Een voorbeeld van het effect van pivoting met floating-point aritmetiek en 3-cijfer mantisse.

$$0.1 \times 10^{-3}x_1 + 0.1 \times 10^1x_2 = 0.1 \times 10^1$$

$$0.1 \times 10^1x_1 + 0.1 \times 10^1x_2 = 0.2 \times 10^1.$$

Zonder pivoting:

$$A^{(0)} = \begin{bmatrix} 0.1 \times 10^{-3} & 0.1 \times 10^1 & 0.1 \times 10^1 \\ 0.1 \times 10^1 & 0.1 \times 10^1 & 0.2 \times 10^1 \end{bmatrix}, \quad m_{21} = \frac{a_{21}}{a_{11}} = 0.1 \times 10^5,$$

$$A^{(1)} = \begin{bmatrix} 0.1 \times 10^{-3} & 0.1 \times 10^1 & 0.1 \times 10^1 \\ 0 & -0.1 \times 10^5 & -0.1 \times 10^5 \end{bmatrix}.$$

Merk op dat zowel $0.1 \times 10^1 - 0.1 \times 10^5 = -0.1 \times 10^5$
als $0.2 \times 10^1 - 0.1 \times 10^5 = -0.1 \times 10^5$ met de 3-cijfer mantisse.

Dit geeft $x_1 = 0.00$ en $x_2 = 1.00$.

Met pivoting:

$$A^{(0)} = \begin{bmatrix} 0.1 \times 10^1 & 0.1 \times 10^1 & 0.2 \times 10^1 \\ 0.1 \times 10^{-3} & 0.1 \times 10^1 & 0.1 \times 10^1 \end{bmatrix}, \quad m_{21} = \frac{a_{21}}{a_{11}} = 0.1 \times 10^{-3},$$

$$A^{(1)} = \begin{bmatrix} 0.1 \times 10^1 & 0.1 \times 10^1 & 0.2 \times 10^1 \\ 0 & 0.1 \times 10^1 & 0.1 \times 10^1 \end{bmatrix},$$

want $0.1 \times 10^1 - 0.1 \times 10^{-3} = 0.1 \times 10^1$ met de 3-cijfer mantisse.

Dit geeft $x_1 = 0.100 \times 10^1$ en $x_2 = 0.100 \times 10^1$.

Terugsubstitutie in het oorspronkelijke stelsel laat zien dat dit antwoord aanzienlijk beter is dan het zonder pivoting verkregen antwoord. \triangle

2.4. LU decompositie

Het Gauss-eliminatie proces vormt de matrix A via elementaire rij-bewerkingen om in een bovendriehoeksmatrix U :

$$U = \begin{bmatrix} a_{11} & a_{12} & \dots & \dots & a_{1n} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & \dots & a_{2n}^{(1)} \\ \vdots & 0 & a_{33}^{(2)} & \dots & a_{3n}^{(2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & a_{nn}^{(n-1)} \end{bmatrix}. \quad (2.10)$$

Tevens worden multiplicatoren m_{ji} berekend die kunnen worden opgezameld in een onderdriehoeksmatrix L (zie (2.5), (2.6) en (2.7)):

$$L = \begin{bmatrix} 1 & 0 & \dots & \dots & 0 \\ m_{21} & 1 & & & \vdots \\ \vdots & m_{32} & \ddots & & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ m_{n1} & m_{n2} & \dots & m_{n,n-1} & 1 \end{bmatrix}. \quad (2.11)$$

Stelling 2.3

Indien geldt $L\underline{s} = \underline{b}$ dan is, met de notaties van (2.5), (2.6) en (2.7)

$$s_1 = b_1, \quad s_2 = b_2^{(1)}, \quad s_3 = b_3^{(2)}, \quad \dots, \quad s_n = b_n^{(n-1)}.$$

Bewijs

Beschouw het stelsel

$$\begin{bmatrix} 1 & 0 & \dots & \dots & 0 \\ m_{21} & 1 & & & \vdots \\ \vdots & m_{32} & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ m_{n1} & m_{n2} & \dots & m_{n,n-1} & 1 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ \vdots \\ s_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ \vdots \\ b_n \end{bmatrix}.$$

Veeg je hiervan de eerste kolom met Gauss-eliminatie, dan gebruik je *precies* dezelfde multiplicatoren als voor het vegen van de matrix A .

Voor het rechterlid \underline{b} betekent dat, dat na één stap ontstaat $\begin{bmatrix} b_1 \\ b_2^{(1)} \\ \vdots \\ b_n^{(1)} \end{bmatrix}$. Dit argument kan

voor iedere volgende kolom worden herhaald (deze worden immers niet aangetast door vorige Gauss-stappen).

Na twee stappen ontstaat dus $\begin{bmatrix} b_1 \\ b_2^{(1)} \\ b_3^{(2)} \\ \vdots \\ b_n^{(2)} \end{bmatrix}$ en na $(n - 1)$ stappen $\begin{bmatrix} b_1 \\ b_2^{(1)} \\ b_3^{(2)} \\ \vdots \\ b_n^{(n-1)} \end{bmatrix}$.

Hiermee is de stelling bewezen. ○

Stelling 2.3 laat zien dat het niet nodig is om het rechterlid \underline{b} in (2.2) mee te nemen in de Gauss-eliminatie, mits alle multiplicatoren bewaard blijven. Het stelsel $L\underline{x} = \underline{b}$ kan immers door directe substitutie worden opgelost (zie voorbeeld 2.1). Nu zijn alle $b_i^{(i-1)}$ bekend. Zij nu U de bovendriehoeksmatrix die uit het Gauss-eliminatie proces resulteert, dan geeft de oplossing van het bovendriehoekssysteem

$$U\underline{x} = \underline{s}$$

de oplossing van het oorspronkelijke systeem $A\underline{x} = \underline{b}$ (zie ook (2.7) en verder). Het bewaren van de L -matrix is voordelig in die gevallen, waarin het stelsel $A\underline{x} = \underline{b}$ moet worden opgelost voor een groot aantal rechterleden.

Voorbeeld 2.4

We willen het stelsel differentiaalvergelijkingen

$$\underline{x}' = A\underline{x} + \underline{f}(t), \quad \underline{x}(t_0) = \underline{x}_0$$

oplossen met behulp van Crank-Nicolson (zie (1.38)), A hangt niet af van t .

De numerieke oplossing u_{i+1} wordt gekarakteriseerd door

$$(I - \frac{1}{2}hA)\underline{u}_{i+1} = (I + \frac{1}{2}hA)\underline{u}_i + \frac{1}{2}h(\underline{f}_i + \underline{f}_{i+1}), \quad \underline{u}_0 = \underline{x}_0.$$

We bepalen nu eenmalig de L - en U -matrix behorend bij de matrix $I - \frac{1}{2}hA$.

In iedere tijdstap lossen we dan op:

$$L\underline{s}_{i+1} = (I + \frac{1}{2}hA)\underline{u}_i + \frac{1}{2}h(\underline{f}_i + \underline{f}_{i+1})$$

en
$$U\underline{u}_{i+1} = \underline{s}_{i+1}.$$

Hoewel we dus in iedere tijdstap een stelsel vergelijkingen moeten oplossen, reduceert dit tot een eenvoudige heen- en terugsubstitutie, hetgeen natuurlijk veel efficiënter is dan het uitvoeren van het hele eliminatieproces in iedere tijdstap. Dit kan natuurlijk alleen maar omdat de matrix A niet van t afhangt. △

2.4.1. Verband tussen L , U en A

Aangezien de matrices L en U niet van het rechterlid b , maar uitsluitend van de matrix A afhangen, ligt het voor de hand om te zoeken naar een verband tussen L , U en A . Dit wordt gegeven door de volgende stelling.

Stelling 2.4

Laat A een inverteerbare matrix zijn en laat L en U gedefinieerd zijn als in (2.10) respectievelijk (2.11). Dan geldt: $A = LU$.

Bewijs

Voor elke $\underline{b} \in \mathbb{R}^n$ is de oplossing van $A\underline{x} = \underline{b}$ gelijk aan de oplossing van het systeem

$$\begin{aligned} L\underline{s} &= \underline{b} \\ U\underline{x} &= \underline{s}, \end{aligned}$$

dus ook gelijk aan de oplossing van het systeem $(LU)\underline{x} = \underline{b}$ (volgt door substitutie van $\underline{s} = U\underline{x}$ in $L\underline{s} = \underline{b}$, zie stelling 2.3).

Neem nu voor \underline{b} de eerste kolom van de matrix A , \underline{a}_1 . Men verifieert eenvoudig dat de oplossing van $A\underline{x} = \underline{a}_1$ wordt gegeven door

$$\underline{x} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Omgekeerd is waar dat wanneer men een matrix vermenigvuldigt met de vector

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

het resultaat juist de eerste kolom van die matrix is.

Daar $\begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$ de oplossing is van het systeem $(LU)\underline{x} = \underline{a}_1$ (de oplossing hiervan valt

immers samen met de oplossing van $A\underline{x} = \underline{b}$), volgt dus dat \underline{a}_1 de eerste kolom is van de matrix LU .

Door nu voor \underline{b} achtereenvolgens alle kolommen van de matrix A te nemen, volgt met een analoge redenering dat deze gelijk zijn aan de overeenkomstige kolommen van de matrix LU . Dus $A = LU$. \circ

Op grond van stelling 2.4 spreekt men wel van LU decompositie of driehoeksontbinding van de matrix A .

2.4.2. De inverse in LU-vorm

Uit het voorgaande volgt dat het bekend zijn van de LU decompositie van A equivalent is met het bekend zijn van de inverse A^{-1} . Ook is het aantal rekenkundige bewerkingen om \underline{x} uit te rekenen via

$$\underline{x} = A^{-1}\underline{b}$$

of $L\underline{s} = \underline{b}$

$$U\underline{x} = \underline{s}$$

2.4.3. Varianten van de LU decompositie

In de praktijk treden een aantal varianten van de LU decompositie op die weliswaar mathematisch equivalent zijn met Gauss-eliminatie, maar vanuit programmeer-efficiëntie soms verkieslijker zijn.

Een van de meer populaire is de methode van Crout, of het oprollen van de matrix van linksboven af. Dit is gebaseerd op het volgende.

Veronderstel dat van de matrix A de LU decompositie bepaald is, maar deze matrix wordt uitgebreid met een extra rij en een extra kolom. Hoe ziet de LU decompositie van de nieuwe matrix eruit? Na enige contemplatie zal het duidelijk zijn dat ook aan de L en de U een rij en een kolom moeten worden toegevoegd (de reeds gevonden L en U worden immers door het Gauss proces niet aangetast) zodat het geheel de gedaante krijgt:

$$\begin{bmatrix} A & \underline{a}_c \\ \underline{a}_r^T & a_d \end{bmatrix} = \begin{bmatrix} L & 0 \\ \underline{l}^T & 1 \end{bmatrix} \begin{bmatrix} U & \underline{u} \\ 0 & u_d \end{bmatrix} \quad (2.12)$$

Hierin is \underline{a}_c de aan A toegevoegde kolom, \underline{a}_r^T de aan A toegevoegde rij etcetera. Uitvermenigvuldigen van (2.12) geeft:

1. $A = LU$
2. $\underline{a}_c = L\underline{u}$
3. $\underline{a}_r^T = \underline{l}^T U$
4. $a_d = \underline{l}^T \underline{u} + u_d$

(2.13)

Door de terugsubstituties

$$L\underline{u} = \underline{a}_c$$

$$U^T \underline{l} = \underline{a}_r$$

en
$$u_d = a_d - \underline{l}^T \underline{u}$$

kunnen dus de aan L toe te voegen rij en de aan U toe te voegen kolom worden bepaald, alsmede het aan U toe te voegen diagonaalelement u_d . Het Crout-proces maakt systematisch gebruik van (2.12). Beginnend met het linker bovenelement a_{11} ($l_{11} = 1, u_{11} = a_{11}$) wordt (2.12) herhaald toegepast en steeds een rij en een kolom toegevoegd aan de reeds gevonden L en U totdat de gehele matrix is ontbonden.

2.4.4. Varianten voor symmetrische matrices

Als de $n \times n$ matrix A symmetrisch is, heeft men in principe maar $\frac{1}{2}(n+1)n$ plaatsen nodig om A op te slaan. De onderdriehoek is immers gespiegeld ten opzichte van de

bovendriehoek. Men heeft gezocht naar methoden van decompositie, zodat ook voor de decompositie maar $\frac{1}{2}(n+1)n$ plaatsen nodig is. Hoe dit kan vertelt de volgende stelling.

Stelling 2.5

Een niet-singuliere symmetrische matrix A is (eventueel met rij- en kolomverwisseling) te ontbinden in $A = LDL^T$. Hierbij is D diagonaal en heeft L de vorm van (2.11). Deze ontbinding is eenduidig.

Bewijs

We gebruiken het Crout algoritme: $l_{11} = 1, d_{11} = a_{11}$. Dit is de eenduidige ontbinding van de 1×1 matrix. Stel dat we voor een $k \times k$ symmetrische matrix A een LDL^T ontbinding hebben. Voeg nu aan A een rij en een kolom toe. Er moet gelden

$$\begin{bmatrix} A & \underline{a}_c \\ \underline{a}_c^T & a_d \end{bmatrix} = \begin{bmatrix} L & 0 \\ \underline{l}^T & 1 \end{bmatrix} \begin{bmatrix} D & 0 \\ 0 & d \end{bmatrix} \begin{bmatrix} L^T & \underline{l} \\ 0 & 1 \end{bmatrix}. \quad (2.14)$$

Uitvermenigvuldigen geeft:

1. $A = LDL^T$
2. $\underline{a}_c = LD\underline{l}$
3. $\underline{a}_c^T = \underline{l}^TDL^T$
4. $a_d = \underline{l}^TD\underline{l} + d$

(2.15)

(2) en (3) zijn elkaars getransponeerde, dus inderdaad levert dit iets symmetrisch op. De terugsubstitutie

$$\begin{aligned} L\underline{s} &= \underline{a}_c \\ D\underline{l} &= \underline{s} \end{aligned}$$

geeft de aan L toe te voegen kolom. ○

2.4.5. Choleski decompositie

Een andere populaire variant is de Choleski decompositie. Deze is alleen mogelijk indien de diagonaalelementen van D positief zijn.

Stelling 2.6

Als A symmetrisch is en de diagonaalelementen van D in de LDL^T ontbinding zijn positief, dan is er een onderdriehoeksmatrix G zó dat $A = GG^T$.

Bewijs

Laat T een diagonaalmatrix zijn met $t_{ii} = \sqrt{d_{ii}}$ (d_{ii} zijn de diagonaalelementen van D). Stel $G = LT$, dan is $GG^T = (LT)(LT)^T = LTT^T L^T = LDL^T$. Hiermee is de stelling bewezen. \circ

Praktisch zal men de GG^T decompositie weer op de Crout-manier bepalen:

$$\begin{bmatrix} A & \underline{a}_c \\ \underline{a}_c^T & a_d \end{bmatrix} = \begin{bmatrix} G & 0 \\ \underline{g}^T & t \end{bmatrix} \begin{bmatrix} G^T & \underline{g} \\ 0 & t \end{bmatrix}$$

$$A = GG^T$$

$$\underline{a}_c = G\underline{g}$$

$$a_d = \underline{g}^T \underline{g} + t^2$$

Men ziet dat wil dit werken in ieder geval alle diagonaalelementen van A positief moeten zijn, maar dit is lang niet voldoende: in iedere stap moet $a_d - \underline{g}^T \underline{g}$ positief zijn. (Dit is precies het element d in de LDL^T decompositie).

Hoe kan men van te voren weten of de Choleski decompositie werkt? Daartoe dragen we eerst wat apparatuur uit de lineaire algebra aan:

Stelling 2.7

$$(\underline{x}, B\underline{y}) = (B^T \underline{x}, \underline{y}).$$

Bewijs

Daar $\underline{x}^T \underline{x} = \sum_{i=1}^n x_i^2 = (\underline{x}, \underline{x})$,

geldt: $(\underline{x}, B\underline{y}) = \underline{x}^T B\underline{y} = \underline{x}^T (B^T)^T \underline{y} = (B^T \underline{x})^T \underline{y} = (B^T \underline{x}, \underline{y})$.

Hierbij is gebruik gemaakt van het feit dat de getransponeerde van een produkt van twee matrices wordt verkregen door de factoren te transponeren en de vermenigvuldigingsvolgorde om te draaien (\underline{x}^T wordt dan beschouwd als een $1 \times n$ matrix).

Veronderstel dat de niet-singuliere matrix A een ontbinding GG^T heeft. A is dan in ieder geval symmetrisch (waarom?), maar er moet meer gelden. Beschouw $(\underline{x}, A\underline{x})$, met $\underline{x} \neq \underline{0}$. Wegens $A = GG^T$ geldt

$$(\underline{x}, A\underline{x}) = (\underline{x}, GG^T \underline{x})$$

en wegens stelling 2.7:

$$(\underline{x}, A\underline{x}) = (G^T \underline{x}, G^T \underline{x}).$$

Omdat A niet singulier is, is G^T niet singulier en dus is $G^T \underline{x} \neq 0$. Stel $G^T \underline{x} = \underline{z} \neq 0$, dan is $(G^T \underline{x}, G^T \underline{x}) = (\underline{z}, \underline{z}) = \sum_{i=1}^n z_i^2 > 0$. \circ

Dit is een zeer opmerkelijke eigenschap: voor alle vectoren $\underline{x} \neq \underline{0}$ geldt dus $(\underline{x}, A\underline{x}) > 0$.

Definitie 2.3

Een $n \times n$ matrix A heet *positief definitief* indien

1. de matrix A symmetrisch is,
2. voor iedere $\underline{x} \neq \underline{0}$ geldt: $(\underline{x}, A\underline{x}) > 0$. \square

Stelling 2.8

De Choleski decompositie van een niet-singuliere matrix A bestaat dan en slechts dan wanneer A positief definitief is.

Bewijs

Dat het positief definitief zijn van A noodzakelijk is voor het bestaan van de Choleski decompositie zagen we reeds. We zullen nu bewijzen dat het ook voldoende is.

Omdat A positief definitief is, is A symmetrisch en bestaat de LDL^T decompositie. Wegens $(\underline{x}, LDL^T \underline{x}) > 0$ voor alle $\underline{x} \neq \underline{0}$ geldt (met behulp van stelling 2.7)

$$(L^T \underline{x}, DL^T \underline{x}) > 0 \text{ voor alle } \underline{x} \neq \underline{0}$$

en omdat L niet-singulier is, is er voor alle $\underline{z} \neq \underline{0}$ een $\underline{x} \neq \underline{0}$ zodanig dat $L^T \underline{x} = \underline{z}$.

Kies nu \underline{z} achtereenvolgens

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \dots \text{ enz.}$$

Dit geeft, wegens $(\underline{z}, D\underline{z}) > 0$: $d_{11} > 0$, $d_{22} > 0$, ... enz.

Dus de diagonaalelementen van D zijn positief en de Choleski decompositie bestaat. Hiermee is de stelling bewezen. \circ

Het verifiëren of een matrix positief definitief is, is niet zo eenvoudig. De volgende stelling geeft een voldoende (maar niet noodzakelijke) voorwaarde voor het positief semi-definitief zijn van een matrix A , d.w.z. $(\underline{x}, A\underline{x}) \geq 0$ voor iedere $\underline{x} \in \mathbb{R}^n \setminus \{\underline{0}\}$. Voor een semi-definiete A kunnen enkele (of alle) diagonaalelementen van de matrix D nul worden. Wanneer echter A niet-singulier is, zijn ook alle diagonaalelementen van D ongelijk aan nul. Immers

$$\det(A) = \det(LDL^T) = \det(L)\det(D)\det(L^T) = a_{11}a_{22}^{(1)} \dots a_{nn}^{(n-1)}.$$

Is A daarnaast positief semi-definiet, dan is A ook positief definiet en kan de Choleski decompositie worden toegepast.

Stelling 2.9 (Gerschgorin)

Zij A een reële symmetrische $n \times n$ matrix met positieve diagonaalelementen. Zij voorts

$$a_{ii} \geq \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}| \quad i = 1, 2, \dots, n.$$

Dan is A positief semi-definiet.

Bewijs

Zie Appendix B.

Toepassing

Beschouw de matrix

$$\begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}$$

We berekenen de LDL^T decompositie met behulp van Crout. $l_{11} = 1$, $d_{11} = 2$. De eerste Crout-stap geeft: $LDl = -1$ (de vector \underline{d}_c heeft maar 1 component) dus: $l = -\frac{1}{2}$.

$$-\frac{1}{2} \cdot 2 - \frac{1}{2} + d = 2 \Rightarrow d = \frac{3}{2}.$$

Na de eerste Crout-stap is de ontbinding:

$$L = \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & 1 \end{bmatrix} \quad D = \begin{bmatrix} 2 & 0 \\ 0 & \frac{3}{2} \end{bmatrix}.$$

De tweede Crout-stap geeft:

$$LDl = \begin{bmatrix} 0 \\ -1 \end{bmatrix} \Rightarrow Dl = \begin{bmatrix} 0 \\ -1 \end{bmatrix}, \quad l = \begin{bmatrix} 0 \\ -\frac{2}{3} \end{bmatrix}.$$

$$(0, -\frac{2}{3}) \begin{bmatrix} 2 & 0 \\ 0 & \frac{3}{2} \end{bmatrix} \begin{bmatrix} 0 \\ -\frac{2}{3} \end{bmatrix} + d = 2 \Rightarrow d = 2 - \frac{2}{3} = \frac{4}{3}.$$

De totale ontbinding is dus:

$$L = \begin{bmatrix} 1 & 0 & 0 \\ -\frac{1}{2} & 1 & 0 \\ 0 & -\frac{2}{3} & 1 \end{bmatrix} \quad D = \begin{bmatrix} 2 & 0 & 0 \\ 0 & \frac{3}{2} & 0 \\ 0 & 0 & \frac{4}{3} \end{bmatrix}.$$

De matrix is positief semi-definiet volgens stelling 2.9. De Choleski decompositie bestaat dus ook. Bereken deze zelf.

2.5. Conditie

Beschouw het stelsel:

$$\begin{aligned} x_1 + 1.001 x_2 &= 2 \\ 1.001 x_1 + x_2 &= 2 \end{aligned} \tag{2.17}$$

en het stelsel

$$\begin{aligned} x_1 + 1.001 x_2 &= 2.1 \\ 1.001 x_1 + x_2 &= 2. \end{aligned}$$

Het eerste heeft de oplossing

$$x_1 = x_2 = \frac{2000}{2001} \approx 0.9995$$

het tweede

$$x_1 = \frac{-98000}{2001} \approx -48.976 \quad x_2 = \frac{102100}{2001} \approx 51.024.$$

Dus een verandering van slechts 5% in het rechterlid geeft een verandering in de oplossing van een factor 50. Het zal duidelijk zijn dat we in de praktijk dit soort stelsels moeten trachten te vermijden. Immers, wanneer de gegevens uit het rechterlid bijvoorbeeld uit metingen bepaald moeten worden, zullen zij niet volkomen nauwkeurig zijn, maar een kleine afwijking (bijvoorbeeld 1%) kunnen vertonen. Als daardoor de oplossing enkele orden van grootte kan veranderen is de betrouwbaarheid van de uitkomsten nihil. Zulke stelsels heten slecht geconditioneerd: een kleine verstoring in de gegevens veroorzaakt een grote verstoring in de oplossing (vergelijk dit met de opmerkingen ten aanzien van stabiliteit in paragraaf 1.2.2).

De volledige analyse van de conditie van systemen valt enigszins buiten het bestek van dit boek. We zullen ons beperken tot symmetrische matrices. Eerst beschouwen we het 2×2 geval.

Een grafische weergave van (2.17) toont direct wat er aan de hand is: we trachten het snijpunt te bepalen van twee (nagenoeg) evenwijdige lijnen. Zowel een kleine verandering in de matrix (d.w.z. verdraaiing van een der lijnen) of in het rechterlid (verschuiving) veroorzaakt een grote verandering in het snijpunt.

Een andere manier om er tegenaan te kijken is op te merken dat de rij vectoren in (2.17) van de matrix A 'bijna' lineair afhankelijk zijn. De matrix is bijna singulier.

2.5.1. Relatie tussen conditie en grootte van de determinant?

Uit de theorie weten we dat de matrix A singulier is wanneer $\det(A) = 0$. Dit wekt de suggestie dat bij een slecht geconditioneerd systeem de determinant klein zou moeten zijn. Dit is echter niet het geval. De determinant van systeem (2.17) is

$$(1 - (1.001)^2) = -2.01 \times 10^{-3}$$

en dit is welliswaar klein (in absolute waarde) ten opzichte van de coëfficiënten, maar als we beide vergelijkingen met 1000 vermenigvuldigen, dan wordt de determinant

$$((1000)^2 - (1001)^2) = -2.001 \times 10^3$$

en dat is niet klein. *Toch is de conditie van het systeem niet veranderd!* De grootte van de determinantwaarde geeft geen uitsluitsel ten aanzien van de conditie van het probleem.

2.5.2. Relatie tussen conditie en eigenwaarden

Alvorens dieper in te gaan op de conditie van systemen zullen we eerst het begrip zelf wat preciseren.

Allereerst brengen wij het begrip *norm* in herinnering (zie [2], 1.7.4).

Definitie 2.4

Zij $\underline{x} \in \mathbb{R}^n$. Onder de Euclidische norm (in het vervolg kortweg aangeduid met *norm*) van \underline{x} verstaan we het reële getal $\|\underline{x}\|$ gedefinieerd door:

$$\|\underline{x}\| = \sqrt{\sum_{i=1}^n x_i^2}.$$

□

Merk op dat deze definitie van norm niets anders is dan de *lengte* van de vector \underline{x} .

2.5.3. Eigenschappen van de norm

De norm heeft de volgende eigenschappen:

1. $\|\underline{x}\| \geq 0$, $\|\underline{x}\| = 0$ dan en slechts dan als $\underline{x} = \underline{0}$.
2. $\|\alpha \underline{x}\| = |\alpha| \cdot \|\underline{x}\|$ voor elke $\alpha \in \mathbb{R}$.
3. $\|\underline{x} + \underline{y}\| \leq \|\underline{x}\| + \|\underline{y}\|$ (driehoeksongelijkheid).

Voor het bewijs van deze stellingen wordt u verwezen naar [2], 1.7.8.

2.5.4. Het verstoorde systeem

We willen oplossen het stelsel $A\underline{x} = \underline{b}$. Wordt het rechterlid nu verstoord met een fout $\Delta \underline{b}$, zal tengevolge daarvan in de oplossing \underline{x} een fout $\Delta \underline{x}$ optreden, zodat we in feite oplossen

$$A(\underline{x} + \Delta \underline{x}) = \underline{b} + \Delta \underline{b}. \quad (2.18)$$

We willen nu een verband leggen tussen de relatieve fout in $\|\underline{x}\|$, dus $\frac{\|\Delta \underline{x}\|}{\|\underline{x}\|}$ en de relatieve fout in $\|\underline{b}\|$, dus $\frac{\|\Delta \underline{b}\|}{\|\underline{b}\|}$.

Nu kunnen we $\Delta \underline{x}$ expliciet uitdrukken in $\Delta \underline{b}$, immers trekken we $A\underline{x} = \underline{b}$ af van systeem (2.18) dan vinden we $A\Delta \underline{x} = \Delta \underline{b}$.

We moeten nog een verband vinden voor $\|A\Delta \underline{x}\|$ en $\|\Delta \underline{x}\|$. In het geval dat A een symmetrische matrix is, is dit eenvoudig aan te geven. We zullen ons daarom tot dit geval beperken.

Stelling 2.10

Zij A een symmetrische $n \times n$ matrix, met eigenwaarden gerangschikt naar absolute grootte, dus $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_n|$ en bijbehorende eigenvectoren $\underline{v}_1, \underline{v}_2, \dots, \underline{v}_n$.

Zij $\underline{x} \in \mathbb{R}^n$ willekeurig, dan geldt:

$$|\lambda_n| \cdot \|\underline{x}\| \leq \|A\underline{x}\| \leq |\lambda_1| \cdot \|\underline{x}\|.$$

Bewijs

We merken op dat $\|\underline{x}\|^2 = (\underline{x}, \underline{x})$. Verder vormen de eigenvectoren van een symmetrische matrix een orthonormale basis in \mathbb{R}^n (zie [2], 9.1.11)

$$\begin{aligned} (\underline{v}_i, \underline{v}_j) &= 0 & i \neq j \\ &= 1 & i = j \end{aligned} \quad (2.19)$$

en iedere vector \underline{x} is eenduidig te schrijven als een lineaire combinatie van eigen-

vectoren. Stel nu $\underline{x} = \sum_{i=1}^n \alpha_i \underline{v}_i$, dan geldt:

$$\begin{aligned} \|\underline{x}\|^2 = (\underline{x}, \underline{x}) &= \left(\sum_{i=1}^n \alpha_i \underline{v}_i, \sum_{j=1}^n \alpha_j \underline{v}_j \right) = \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j (\underline{v}_i, \underline{v}_j) = \\ &= \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j (\underline{v}_i, \underline{v}_j) = \sum_{i=1}^n \alpha_i^2, \text{ wegens (2.19).} \end{aligned}$$

$$\|A\underline{x}\|^2 = (A\underline{x}, A\underline{x}) = \left(\sum_{i=1}^n \alpha_i A\underline{v}_i, \sum_{j=1}^n \alpha_j A\underline{v}_j \right)$$

en wegens $A\underline{v}_i = \lambda_i \underline{v}_i$ geldt dus:

$$\begin{aligned} \|A\underline{x}\|^2 &= \left(\sum_{i=1}^n \alpha_i \lambda_i \underline{v}_i, \sum_{j=1}^n \alpha_j \lambda_j \underline{v}_j \right) = \sum_{i=1}^n \sum_{j=1}^n \alpha_i \lambda_i \alpha_j \lambda_j (\underline{v}_i, \underline{v}_j) = \\ &= \sum_{i=1}^n \alpha_i^2 \lambda_i^2, \text{ wegens (2.19).} \end{aligned} \tag{2.20}$$

Uit (2.20) volgt, daar $|\lambda_1| \geq |\lambda_i| \quad i = 2, 3, \dots, n$ en $|\lambda_n| \leq |\lambda_i| \quad i = 1, 2, \dots, n-1$, dat

$$\lambda_n^2 \sum \alpha_i^2 \leq \|A\underline{x}\|^2 \leq \lambda_1^2 \sum \alpha_i^2,$$

dus: $\lambda_n^2 \|\underline{x}\|^2 \leq \|A\underline{x}\|^2 \leq \lambda_1^2 \|\underline{x}\|^2$

en hieruit $|\lambda_n| \cdot \|\underline{x}\| \leq \|A\underline{x}\| \leq |\lambda_1| \cdot \|\underline{x}\|$,

waarmee stelling 2.10 bewezen is. ○

Met behulp van stelling 2.10 vinden we:

$$\|\Delta \underline{b}\| = \|A \Delta \underline{x}\| \geq |\lambda_n| \cdot \|\Delta \underline{x}\| \quad \text{en} \quad \|\underline{b}\| = \|A\underline{x}\| \leq |\lambda_1| \cdot \|\underline{x}\|.$$

Hiermee vindt men de gezochte uitdrukking:

$$\frac{\|\Delta \underline{x}\|}{\|\underline{x}\|} \leq \frac{|\lambda_1|}{|\lambda_n|} \cdot \frac{\|\Delta \underline{b}\|}{\|\underline{b}\|}.$$

De grootheid $|\lambda_1|/|\lambda_n|$ wordt *conditiegetal* van de reële symmetrische matrix A notatie $\text{cond}(A)$, genoemd. Hoe groter het conditiegetal, des te slechter is het systeem geconditioneerd.

2.5.5. Numerieke singulariteit

Echt singuliere matrices komen bij het werken met floating-point getallen haast niet voor. Door afrondingen van getallen wordt meestal een reguliere matrix verkregen. Dat dit niettemin weinig helpt ziet men als volgt in.

Als A singulier is, dan bezit A tenminste één eigenwaarde 0 (immers $A\underline{x} = \underline{0}$ heeft een niet-triviale oplossing). De eigenwaarden hangen continu af van de matrixcoëfficiënten. Dus A in floating-point hoeft niet meer singulier te zijn, maar bezit wel een eigenwaarde zeer dicht in de buurt van 0. Dat betekent echter dat $\text{cond}(A)$ zeer groot is en het systeem zeer slecht geconditioneerd.

2.5.6. Praktische bepaling van de conditie

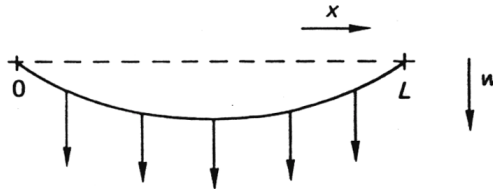
Het uitrekenen van het conditiegetal is vaak te bewerkelijk. Een praktische manier om de conditie van een systeem te weten te komen is het stelsel op te lossen voor verschillende dicht bij elkaar liggende rechterleden. Dit is niet veel extra werk wanneer we met de LU decompositie werken.

Toepassingen

2.6. Lineaire randwaarden problemen

2.6.1. De kabelvergelijking

Op het interval $[0, L]$ wordt een (gewichtsluus gedachte) kabel gespannen, die belast wordt als aangegeven in figuur 2.1.



Figuur 2.1.

De belasting $\psi(x)$ wordt gedacht continu verdeeld te zijn. De zakking $w(x)$ van de kabel wordt dan gegeven door de differentiaalvergelijking

$$-T_0 \frac{d^2 w}{dx^2} = \psi(x), \quad w(0) = w(L) = 0. \quad (2.21)$$

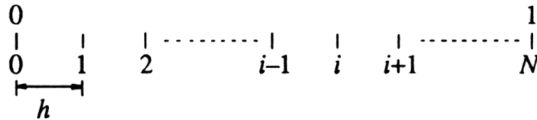
De grootte T_0 is de horizontale component van de trekkracht. We kunnen nu (2.21)

zo dimensioneren (neem $y = \frac{T_0 w}{L^2}$, $\xi = \frac{x}{L}$, $f(\xi) = \psi(L\xi)$), dat (2.21) equivalent is met het modelprobleem.

$$-\frac{d^2y}{dx^2} = f(x), \quad y(0) = y(1) = 0. \quad (2.22)$$

Behalve als de kabelvergelijking kunnen we deze problemen ook interpreteren als de warmteverdeling in een homogene staaf waar de uiteinden op temperatuur $y = 0$ gehouden wordt. De functie f geeft dan warmtebronnen in de staaf weer. Wanneer f een gemakkelijk te integreren functie is, kan (2.22) analytisch worden opgelost. Vaak is dat echter niet het geval. De numerieke oplosmethode die we nu presenteren is daar niet gevoelig voor.

2.6.2. Differentiemethoden



We verdelen het interval $[0, 1]$ in N deel-intervallen met lengte $h = \frac{1}{N}$. We schrijven weer $y_i = y(ih)$ en $f_i = f(ih)$. In het steunpunt x_i geldt (2.22).

$$-\frac{d^2y}{dx^2} \Big|_{x_i} = f_i. \quad (2.23)$$

We trachten nu y_i met behulp van (2.23) uit te drukken in y_{i-1} en y_{i+1} . Daartoe de volgende stelling.

Stelling 2.11

Laat y viermaal continu differentieerbaar zijn. Dan geldt, met de notaties zoals hierboven genoemd:

$$\frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} = \frac{d^2y}{dx^2} \Big|_{x_i} + O(h^2).$$

Bewijs

Volgens de formule van Taylor (zie [1], 4.7.5) geldt:

$$y_{i+1} = y_i + h \frac{dy}{dx} \Big|_{x_i} + \frac{h^2}{2!} \frac{d^2y}{dx^2} \Big|_{x_i} + \frac{h^3}{3!} \frac{d^3y}{dx^3} \Big|_{x_i} + O(h^4)$$

en

$$y_{i-1} = y_i - h \frac{dy}{dx} \Big|_{x_i} + \frac{h^2}{2!} \frac{d^2y}{dx^2} \Big|_{x_i} - \frac{h^3}{3!} \frac{d^3y}{dx^3} \Big|_{x_i} + O(h^4). \quad (2.24)$$

Optelling van de uitdrukkingen in (2.24) geeft:

$$y_{i+1} - 2y_i + y_{i-1} = h^2 \left. \frac{d^2y}{dx^2} \right|_{x_i} + O(h^4)$$

Uit links en rechts delen door h^2 volgt het gestelde.

De grootheid $\frac{y_{i+1} - 2y_i + y_{i-1}}{h^2}$ wordt tweede differentiequotiënt of ook wel tweede gedeelde differentie genoemd. (De eerste gedeelde differentie is $\frac{y_{i+1} - y_i}{h}$, de tweede is $\frac{(\frac{y_{i+1} - y_i}{h}) - (\frac{y_i - y_{i-1}}{h})}{h}$). Bij de ongedeelde differenties wordt niet door h gedeeld).

Oefening 2.1

Ga na dat geldt:

$$\frac{y_{i+1} - y_i}{h} = \left. \frac{dy}{dx} \right|_{x_i} + O(h) \quad \text{en} \quad \frac{y_{i+1} - y_{i-1}}{2h} = \left. \frac{dy}{dx} \right|_{x_i} + O(h^2).$$

Door nu in (2.23) het differentiaalquotiënt te vervangen door het differentiequotiënt, met verwaarlozing van de term van $O(h^2)$ (dit is weer de lokale afbreekfout) ontstaat het volgende stelsel vergelijkingen (de numerieke oplossing wordt weer aangegeven met u_i).

$$\begin{aligned} -u_0 + 2u_1 - u_2 &= h^2 f_1 \\ -u_1 + 2u_2 - u_3 &= h^2 f_2 \\ &\vdots \\ -u_{N-2} + 2u_{N-1} - u_N &= h^2 f_{N-1} \end{aligned} \quad (2.25)$$

Met inachtneming van de randvoorwaarden $y(0) = y(1) = 0$, d.w.z. $u_0 = u_N = 0$ gaat dit over in

$$\begin{aligned} 2u_1 - u_2 &= h^2 f_1 \\ -u_1 + 2u_2 - u_3 &= h^2 f_2 \\ &\vdots \\ -u_{N-3} + 2u_{N-2} - u_{N-1} &= h^2 f_{N-2} \\ -u_{N-2} + 2u_{N-1} &= h^2 f_{N-1} \end{aligned} \quad (2.26)$$

of in matrix-vector notatie:

$$A\mathbf{u} = h^2 \mathbf{f}$$

met

$$A = \begin{bmatrix} 2 & -1 & & & 0 \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ 0 & & \ddots & \ddots & -1 \\ & & & -1 & 2 \end{bmatrix}, \quad (2.27)$$

een $(N-1) \times (N-1)$ -matrix.

De oplossing kan nu gevonden worden met behulp van de LU decompositie of om geheugenruimte te sparen met de Choleski decompositie (A is positief definitief, zie ook stelling 2.9).

2.6.3. Globale fout

Uit stelling 2.11 weten we dat in elk der vergelijkingen van (2.26) een fout zit van $O(h^4)$. Stel nu de fout in de i -de vergelijking $h^4 p_i$ en stel $\Delta y_i = y_i - u_i$, waarin y_i de exacte oplossing en u_i de numerieke oplossing voorstelt. Dan geldt:

$$A\underline{y} = h^2 \underline{f} + h^4 \underline{p} \quad \text{en} \quad A\underline{u} = h^2 \underline{f},$$

met A als in (2.27). Aftrekking levert:

$$A \Delta \underline{y} = h^4 \underline{p}.$$

Zijn $\lambda_1, \lambda_2, \dots, \lambda_{N-1}$ de eigenwaarden van A naar absolute grootte gerangschikt, dan geeft stelling 2.10 een schatting voor de globale fout $\|\Delta \underline{y}\|$:

$$\|\Delta \underline{y}\| \leq \frac{1}{|\lambda_{N-1}|} h^4 \|\underline{p}\|. \quad (2.28)$$

In dit eenvoudige geval kunnen de eigenwaarden van A exact bepaald worden.

Stelling 2.12

Zij A een $(N-1) \times (N-1)$ matrix met een structuur als in (2.27). De eigenwaarden van A worden gegeven door $\lambda_j = 2 - 2\cos\left(\frac{N-j}{N}\pi\right)$, $j = 1, 2, \dots, (N-1)$. \circ

We zullen deze stelling niet bewijzen. Merk op dat de eigenwaarden symmetrisch liggen ten opzichte van 2 en dat ze alle groter dan nul zijn. (Dit laatste is equivalent met het positief definitief zijn van A .)

Er geldt namelijk $\lambda_j = 4\sin^2\left(\frac{N-j}{2N}\pi\right)$. De kleinste eigenwaarde $\lambda_{N-1} = 4\sin^2\left(\frac{1}{N}\frac{\pi}{2}\right)$ en dit is voor grote N ongeveer $\lambda_{N-1} \approx \frac{\pi^2}{N^2} = h^2\pi^2$. Dit gesubstitueerd in (2.28) geeft de

globale fout:

$$\|\Delta \underline{y}\| \leq \frac{h^2}{\pi^2} \|\underline{p}\|.$$

De globale fout is dus $O(h^2)$.

2.6.4. *Conditie*

Berekenen we het conditiegetal van A dan vinden we met stelling 2.12 voor grote N :

$$\text{cond}(A) = \frac{|\lambda_1|}{|\lambda_{N-1}|} \approx \frac{4}{h^2 \pi^2}.$$

Dat ziet er niet zo best uit, want naarmate h kleiner wordt verslechtert de conditie van het systeem. De situatie is echter niet zo erg als het lijkt. Stel dat er een fout in de belastingsvector \underline{f} zit, zeg $\Delta \underline{f}$. Hierdoor ontstaat een fout $\Delta \underline{u}$ in de numerieke oplossing en we lossen op

$$A(\underline{u} + \Delta \underline{u}) = h^2(\underline{f} + \Delta \underline{f}) \quad \text{in plaats van} \quad A\underline{u} = h^2 \underline{f}.$$

Aftrekking geeft $A\Delta \underline{u} = h^2 \Delta \underline{f}$ en met eenzelfde soort redenering als in paragraaf 2.6.3 vinden we voor $\|\Delta \underline{u}\|$

$$\|\Delta \underline{u}\| \leq \frac{1}{\pi^2} \|\Delta \underline{f}\| \tag{2.30}$$

onafhankelijk van h . Dat wil zeggen dat een kleine verstoring in de belastingsvector een kleine verstoring in de verplaatsingsvector veroorzaakt. Het systeem is dus in ieder geval goed geconditioneerd met betrekking tot de absolute fout. Of het ook goed geconditioneerd is met betrekking tot de relatieve fout $\|\Delta \underline{u}\|/\|\underline{u}\|$ hangt af van de belastingsvector \underline{f} . Met name in die gevallen, waarin grote belastingen kleine verplaatsingen veroorzaken (niet erg realistisch, maar het is mogelijk zulke belastinggevallen te construeren) is het systeem slecht geconditioneerd met betrekking tot de relatieve fout. Men kan dit echter aan de oplossing zelf zien, uit (2.30) volgt immers:

$$\frac{\|\Delta \underline{u}\|}{\|\underline{u}\|} \leq \frac{1}{\pi^2} \frac{\|\underline{f}\|}{\|\underline{u}\|} \frac{\|\Delta \underline{f}\|}{\|\underline{f}\|}. \tag{2.31}$$

Dit is een veel minder pessimistische schatting dan met behulp van het conditiegetal. De situatie: goede conditie met betrekking tot de absolute fout en mogelijk slechte conditie met betrekking tot de relatieve fout treedt vrij algemeen op bij randwaardenproblemen. Men doet er goed aan dan de analyse van deze paragraaf te volgen in plaats van te werken met het conditiegetal zelf. Merk ook nog op dat wanneer men de conditie bepaalt volgens paragraaf 2.5.6, men in feite ook een minder pessimistische

met A een $N \times N$ matrix van de vorm

$$A = \begin{bmatrix} 2 & -1 & & & 0 \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ 0 & & & -1 & 1 \end{bmatrix}.$$

De eigenwaarden van deze matrix zijn iets anders dan die van (2.27) maar de kleinste is ook weer $O(h^2)$ voor kleine h .

De laatste vergelijking van systeem (2.34) heeft een lokale afbreekfout van $O(h^3)$, alle andere vergelijkingen hebben een lokale afbreekfout van $O(h^4)$. Op basis van de analyse in paragraaf 2.6.2 zouden we nu een globale fout verwachten van $O(h)$. Deze verwachting is echter niet juist. We beschouwen weer de exacte oplossing y . Hiervoor geldt:

$$A\underline{y} = h^2\underline{f} + h^4\underline{p} + h^3\underline{q},$$

met \underline{p} als in paragraaf 2.6.2 en $\underline{q} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ q_n \end{bmatrix}$.

Definieer nu $A(\Delta_1\underline{y}) = h^4\underline{p}$ en $A(\Delta_2\underline{y}) = h^3\underline{q}$.

Kennelijk geldt $\Delta\underline{y} = \underline{y} - \underline{u} = \Delta_1\underline{y} + \Delta_2\underline{y}$. Voor $\Delta_1\underline{y}$ geldt, juist als in paragraaf 2.6.2

$$\|\Delta_1\underline{y}\| \leq \frac{1}{\lambda_N} h^4 \|\underline{p}\|$$

en daar $\lambda_N = Kh^2$ geldt

$$\|\Delta_1\underline{y}\| \leq Kh^2 \|\underline{p}\|.$$

Nu wordt de oplossing van $A\underline{x} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}$ gegeven door: $x_i = i$, $i = 1, 2, \dots, N$.

De oplossing van $A(\Delta_2\underline{y}) = h^3\underline{q}$ wordt dus gegeven door $\Delta_2 y_i = ih^3 q_n$. Nu is $ih = x_i$, dus alle $\Delta_2 y_i$ zijn $O(h^2)$.

Stel bijvoorbeeld $x_{q_n} = s_i$ dan is $\Delta_2 y_i = h^2 s_i$ en $\|\Delta_2\underline{y}\| = h^2 \|\underline{s}\|$.

Dus voor de totale fout $\|\Delta\underline{y}\|$ geldt weer:

$$\|\Delta\underline{y}\| = \|\Delta_1\underline{y} + \Delta_2\underline{y}\| \leq \|\Delta_1\underline{y}\| + \|\Delta_2\underline{y}\| = h^2(K\|\underline{p}\| + \|\underline{s}\|) = O(h^2).$$

2.6.6. Conditie van het probleem met betrekking tot de randvoorwaarden

Het bovenstaande resultaat, dat voor dit eenvoudige geval volledig uitgearbeit is, kan worden, is algemeen geldig voor problemen die goed geconditioneerd zijn met betrekking tot de randvoorwaarden. Dat wil zeggen dat een kleine verstoring in de randvoorwaarden ook een kleine verstoring in de oplossing veroorzaakt.

In het onderhavige geval maakten we een fout van $O(h^2)$ in de randvoorwaarden, we kregen daardoor een fout van $O(h^2)$ in de oplossing. Het heeft geen zin de randvoorwaarden nauwkeuriger te benaderen, want de oplossing heeft toch al een fout van $O(h^2)$ door het vervangen van de tweede afgeleide door de tweede gedeelde differentie, ook met een fout $O(h^2)$ (zie stelling 2.11). Randwaardenproblemen die voortkomen uit technische problemen voldoen bijna altijd aan het goed geconditioneerd zijn met betrekking tot de randvoorwaarden.

We formuleren daarom een vuistregel met betrekking tot het benaderen van randvoorwaarden die eerste of hogere afgeleiden bevatten.

Vuistregel

Worden in een randwaardenprobleem de differentiaalquotiënten vervangen door gedeelde differenties met een fout $O(h^p)$ dan moeten ook de differentiaalquotiënten in de randwaarden vervangen worden door gedeelde differenties met een fout $O(h^p)$. De oplossing heeft dan ook een globale fout $O(h^p)$.

Merk op dat het gaat om de afbreekfout van de gedeelde differenties en niet om de afbreekfout van de stelsels (2.26) en (2.33), want hierin treden ongedeelde differenties op.

Oefeningen

2.2. Men lost probleem (2.32) op met behulp van een discretisatie als in paragraaf 2.6.2 en verwerkt de randvoorwaarde $y'(1) = 0$ door te nemen $u_N - u_{N-1} = 0$.

- Ga na dat de lokale afbreekfout in de laatste vergelijking van het systeem $O(h^2)$ is.
- Laat zien dat niet voldaan is aan de vuistregel en ga na als in (2.34) en verder dat de globale fout nu $O(h)$ is.

2.3. Gegeven is de differentiaalvergelijking $\frac{d}{dx} \left(p(x) \frac{dy}{dx} \right) = f(x)$, $y(0) = 0$, $y(1) = 1$.

- Laat zien met behulp van de formule van Taylor dat geldt:

$$\frac{p_{i-\frac{1}{2}} y_{i-1} - [p_{i-\frac{1}{2}} + p_{i+\frac{1}{2}}] y_i + p_{i+\frac{1}{2}} y_{i+1}}{h^2} = \frac{d}{dx} p(x) \frac{dy}{dx} \Big|_{x=x_i} + O(h^2).$$

- Formuleer het stelsel differentievergelijkingen met een discretisering als in paragraaf 2.6.2 in matrix vectorvorm $A\mathbf{u} = \mathbf{f}$. Besteed in het bijzonder aandacht

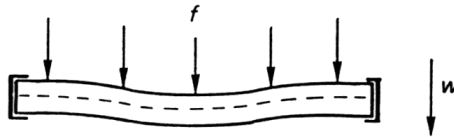
aan f_{N-1} . Laat zien dat A positief semi-definiet is als $p \leq 0$ voor $x \in [0,1]$.

- c. Het probleem $\frac{d}{dx}(p(x)\frac{dy}{dx}) = f(x)$, $y'(0) = y'(1) = 0$ heeft geen eenduidige oplossing. Als y een oplossing is, is $y + C$, C willekeurig $\in \mathbb{R}$ ook een oplossing. Verifieer dat dit ook het geval is met het differentie-analogon als in (2.32). (N.B. Zowel het punt $x = 0$ als $x = 1$ moeten hier in het midden van een interval liggen).

2.6.7. De buigende balk

De door buiging van een balk ter lengte L onder een belasting f (zie figuur 2.2) wordt beschreven door de differentiaalvergelijking

$$EI \frac{d^4 w}{dx^4} = f(x). \tag{2.35}$$



Figuur 2.2.

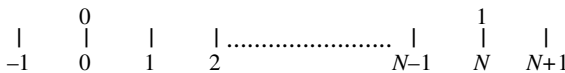
w is de zakking van de zogenaamde *neutrale lijn* (gestippeld in de tekening). De grootte EI hangt af van het materiaal en van de vorm van de dwarsdoorsnede van de balk en wordt buigstijfheid genoemd.

Aan de uiteinden van de balk is de verplaatsing 0. Dit geeft de randvoorwaarden $w(0) = w(L) = 0$. De andere twee randvoorwaarden hangen ervan af hoe de balk aan de uiteinden bevestigd is. Inklemming aan een uiteinde geeft $w' = 0$, vrije oplegging geef $w'' = 0$.

We beschouwen een aan beide zijden ingeklemde balk. Door geschikte dimensionering is (2.35) dan terug te voeren tot het modelprobleem:

$$y^{iv} = f(x), \quad y(0) = y(1) = 0, \quad y'(0) = y'(1) = 0. \tag{2.36}$$

We verdelen het interval $[0,1]$ weer in N intervallen van breedte h , maar om de randvoorwaarden goed in te passen nemen we aan beide uiteinden een extra steunpunt op de volgende manier.



Met de bekende notaties $x_i = ih$, $y_i = y(ih)$ en $f_i = f(ih)$ kan de volgende stelling

V	I (mA)	σ (kg/mm ²)	$\epsilon \times 10^3$
100	1.10	100	0.10
200	2.15	200	0.17
300	3.25	300	0.28
400	4.30	400	0.36
500	5.45	500	0.46
$V = IR$	$R?$	$\sigma = \epsilon E$	$E?$

Tabel 2.2 (a)

(b)

In tabel 2.2a moeten we een oplossing vinden voor het ‘stelsel’ vergelijkingen

$$\begin{aligned}
 1.10R - 100 &= 0 \\
 2.15R - 200 &= 0 \\
 3.25R - 300 &= 0 \\
 4.30R - 400 &= 0 \\
 5.45R - 500 &= 0.
 \end{aligned} \tag{2.39}$$

In plaats daarvan beschouwen we het stelsel

$$\begin{aligned}
 1.10R - 100 &= r_1 \\
 2.15R - 200 &= r_2 \\
 3.25R - 300 &= r_3 \\
 4.30R - 400 &= r_4 \\
 5.45R - 500 &= r_5.
 \end{aligned} \tag{2.40}$$

De grootheden r_i worden residuen genoemd. We proberen nu R zo te bepalen dat

$$r_1^2 + r_2^2 + r_3^2 + r_4^2 + r_5^2 \tag{2.41}$$

minimaal wordt.

Daar alle residuen functies zijn van R moeten we dus minimaliseren:

$$\psi(R) = \sum_1^5 r_i^2. \tag{2.42}$$

Daartoe stellen we $\frac{d\psi}{dR} = 0$. Differentiatie van (2.42) geeft:

$$\frac{d\psi}{dR} = \frac{d}{dR} \left(\sum_1^5 r_i^2 \right) = \sum_1^5 \frac{d}{dR} r_i^2 = 2 \sum_1^5 r_i \frac{dr_i}{dR}. \tag{2.43}$$

Nu is $\frac{dr_1}{dR} = 1.10$, $\frac{dr_2}{dR} = 2.15$ etcetera. Nul stellen van (2.43) geeft:

$$\begin{aligned} & (1.10R - 100) \times 1.10 + (2.15R - 200) \times 2.15 + \\ & + (3.25R - 300) \times 3.25 + (4.30R - 400) \times 4.30 + \\ & + (5.45R - 500) \times 5.45 = 0. \end{aligned}$$

Dit uitgewerkt geeft:

$$R = \frac{5960}{64.5875} = 92.28 \text{ k}\Omega.$$

Deze oplossing wordt kleinste kwadraten oplossing genoemd, omdat de som van de kwadraten van de residuen minimaal gemaakt is.

Oefeningen

2.8. Verifieer dat geldt $\frac{d^2\psi}{dR^2} > 0$.

2.9. Bepaal de kleinste kwadraten oplossing voor E in tabel 2.2b.

2.7.1. n vergelijkingen met m onbekenden ($n > m$)

We beschouwen nu een iets algemener geval namelijk n lineaire vergelijkingen met m onbekenden ($n > m$).

Een oplossing in kleinste kwadraten zin van $A\mathbf{x} = \mathbf{b}$ is een vector \mathbf{x}^* met de eigenschap

$$((\mathbf{b} - A\mathbf{x}^*), (\mathbf{b} - A\mathbf{x}^*)) \leq ((\mathbf{b} - A\mathbf{x}), (\mathbf{b} - A\mathbf{x})) \quad \forall \mathbf{x} \in \mathbb{R}^n$$

Dat wil zeggen, dat van alle mogelijke kandidaten, \mathbf{x}^* zo dicht mogelijk bij een oplossing ligt. In stelling 2.1.4 zullen we laten zien, dat zo een \mathbf{x}^* ook voldoet aan

$$A^T A \mathbf{x}^* = A^T \mathbf{b}$$

Deze vergelijkingen worden de *normaalvergelijkingen* genoemd. Merk op, dat $A^T A$ een $m \times m$ matrix is, zodat er net zoveel normaalvergelijkingen zijn als onbekenden. In sommige gevallen is $A^T A$ singulier en er is dan geen eenduidige oplossing.

Zij gegeven de $n \times m$ matrix A en $\underline{b} \in \mathbb{R}^n$. We zoeken nu een $\underline{x} \in \mathbb{R}^m$ zó dat het stelsel $A\underline{x} = \underline{b}$ wordt opgelost in kleinste kwadraten zin.

Stelling 2.14

De oplossing van het $n \times m$ stelsel $A\underline{x} = \underline{b}$ in kleinste kwadraten zin wordt gegeven door de oplossing van het $m \times m$ stelsel $A^T A \underline{x} = A^T \underline{b}$.

Bewijs

Laat $r = A\bar{x} - \bar{b}$, we zoeken dan een \underline{x} zó dat $(\underline{r}, \underline{r})$ minimaal is.

Veronderstel dat \underline{x}^* de gezochte \underline{x} is. Als nu \underline{v} een willekeurige vector is, betekent dit dat

$$(A(\underline{x}^* + \varepsilon\underline{v}) - \bar{b}, A(\underline{x}^* + \varepsilon\underline{v}) - \bar{b}) \geq (A\underline{x}^* - \bar{b}, A\underline{x}^* - \bar{b})$$

voor elke ε .

Uitwerken van het linker lid geeft:

$$\varepsilon^2(A\underline{v}, A\underline{v}) + 2\varepsilon(A\underline{v}, A\underline{x}^* - \bar{b}) + (A\underline{x}^* - \bar{b}, A\underline{x}^* - \bar{b}) \geq (A\underline{x}^* - \bar{b}, A\underline{x}^* - \bar{b})$$

ofwel:
$$\varepsilon^2(A\underline{v}, A\underline{v}) + 2\varepsilon(A\underline{v}, A\underline{x}^* - \bar{b}) \geq 0$$

voor elke ε en \underline{v} .

Kiezen we $\varepsilon > 0$, dan geldt:

$$\varepsilon(A\underline{v}, A\underline{v}) + 2(A\underline{v}, A\underline{x}^* - \bar{b}) \geq 0$$

dus:
$$(A\underline{v}, A\underline{x}^* - \bar{b}) \geq 0 \quad \text{voor elke } \underline{v}.$$

Kiezen we $\varepsilon < 0$, dan is

$$(A\underline{v}, A\underline{x}^* - \bar{b}) \leq 0 \quad \text{voor elke } \underline{v}.$$

Dus:
$$(A\underline{v}, A\underline{x}^* - \bar{b}) = 0 \quad \text{voor elke } \underline{v}$$

en wegens stelling 2.7 geldt $(\underline{v}, A^T A\underline{x}^* - A^T \bar{b}) = 0$ voor elke \underline{v} . Maar een vector die loodrecht staat op alle vectoren kan alleen maar de nulvector zijn.

Dus $A^T A\underline{x}^* = A^T \bar{b}$.

Hiermee is de stelling bewezen. ○

2.7.2. m-de graadspolynoom door n+1 steunpunten (n > m)

Een van de belangrijkste toepassingen van stelling 2.14 in de techniek is de volgende. Stel dat van een fysische grootte bekend is dat hij zich gedraagt als een polynoom van de graad m in een of andere onafhankelijke variabele. Deze fysische grootte wordt nu gemeten voor verschillende waarden van de onafhankelijk veranderlijke zó dat het aantal metingen (veel) groter is dan de graad van het polynoom. Gevraagd wordt nu coëfficiënten van het polynoom te bepalen zó dat dit zo goed mogelijk (in kleinste kwadraten zin) aansluit bij de meetpunten. Stel de fysische grootte y , de onafhankelijk variabele t . Er moet gelden dat het polynoom

$$y = p_0 t^m + p_1 t^{m-1} + \dots + p_{m-1} t + p_m \tag{2.44}$$

$$\left(\sum_{i=0}^n t_i^m\right)p_0 + \left(\sum_{i=0}^n t_i^{m-1}\right)p_1 + \dots + \left(\sum_{i=0}^n t_i\right)p_{m-1} + (n+1)p_m = \sum_{i=0}^n y_i.$$

2.7.3. Conditie

Stelsel (2.46) is over het algemeen slecht geconditioneerd voor grote m . Het verdient geen aanbeveling om deze polynoom fit te doen met polynomen van graad > 6 . Hoe lager de graad, des te beter is het resultaat. Het nut van deze kleinste kwadraten methode is dat mogelijke meetfouten in één meting worden ‘uitgesmeerd’ en minder doorwerken naarmate het aantal metingen groter is.

Voorbeeld 2.5

Van een eenparig versneld bewegend voorwerp wordt de afgelegde weg gemeten als functie van de tijd.

s	0	1.05	2.23	3.44	4.82	6.30
t	0	0.1	0.2	0.3	0.4	0.5

Bepaal een polynoom van de vorm $s = p_0 t^2 + p_1 t + p_2$ dat in kleinste kwadraten zin zo goed mogelijk bij deze punten aansluit.

We bepalen

$$\sum_{i=0}^5 t_i^m \quad m = 0, 1, 2, 3, 4,$$

alsmede $\sum_{i=0}^5 t_i^m s_i \quad m = 0, 1, 2.$

Dit geeft $\sum_{i=0}^5 t_i = 1.5$; $\sum_{i=0}^5 t_i^2 = 0.55$; $\sum_{i=0}^5 t_i^3 = 0.225$; $\sum_{i=0}^5 t_i^4 = 0.0979$.

$$\sum_{i=0}^5 s_i = 17.84; \quad \sum_{i=0}^5 t_i s_i = 6.661; \quad \sum_{i=0}^5 t_i^2 s_i = 2.7555.$$

Volgens (2.46) moeten we oplossen het stelsel vergelijkingen

$$\begin{aligned} 0.0979 p_0 + 0.225 p_1 + 0.55 p_2 &= 2.7555 \\ 0.225 p_0 + 0.55 p_1 + 1.5 p_2 &= 6.661 \\ 0.55 p_0 + 1.5 p_1 + 6 p_2 &= 17.84. \end{aligned}$$

We vinden (vier-cijfer nauwkeurigheid)

$$p_0 = 5.268; \quad p_1 = 9.943; \quad p_2 = 0.005.$$

Het polynoom dat het beste aansluit in kleinste kwadraten zin bij de meetwaarden is dus $s = 5.268t^2 + 9.943t + 0.005$.

Uit de fysica weten we dat voor de afgelegde weg geldt: $s = \frac{1}{2} at^2 + v_0t + s_0$, met a de versnelling, v_0 de initiële snelheid. Uit de metingen kunnen we dus benaderingen vinden voor versnelling en initiële snelheid door middel van de kleinste kwadraten methode. De conclusie dat $a \approx 10.54$ en $v_0 \approx 9.943$ is alleen gerechtvaardigd als het probleem goed geconditioneerd is, wat hier nog wel het geval is.

In het algemeen moet men echter bijzonder oppassen met het trekken van conclusies ten aanzien van fysische grootheden uit polynoomcoëfficiënten van een kleinste kwadratenbenadering.