**KATHOLIEKE UNIVERSITEIT LEUVEN**
FACULTEIT TOEGEPASTE WETENSCHAPPEN
DEPARTEMENT COMPUTERWETENSCHAPPEN
AFDELING NUMERIEKE ANALYSE EN
TOEGEPASTE WISKUNDE
Celestijnenlaan 200A – 3001 Heverlee

# Semiseparable matrices and the symmetric eigenvalue problem

Promotor:
Prof. Dr. ir. M. Van Barel

Proefschrift voorgedragen tot
het behalen van het doctoraat
in de toegepaste wetenschappen

door

**Raf VANDEBRIL**

mei 2004

**KATHOLIEKE UNIVERSITEIT LEUVEN**
FACULTEIT TOEGEPASTE WETENSCHAPPEN
DEPARTEMENT COMPUTERWETENSCHAPPEN
AFDELING NUMERIEKE ANALYSE EN
TOEGEPASTE WISKUNDE
Celestijnenlaan 200A – 3001 Heverlee

# Semiseparable matrices and the symmetric eigenvalue problem

Jury:
Prof. Dr. ir. J. Berlamont, voorzitter
Prof. Dr. ir. M. Van Barel, promotor
Prof. Dr. D. A. Bini
(Università di Pisa)
Prof. Dr. A. Bultheel
Prof. G. H. Golub
(Stanford University)
Prof. Dr. ir. A. Haegemans
Dr. N. Mastronardi
(IAC-CNR, Bari)
Prof. Dr. ir. S. Van Huffel

Proefschrift voorgedragen tot
het behalen van het doctoraat
in de toegepaste wetenschappen

door

**Raf VANDEBRIL**

U.D.C. 681.3*G13

mei 2004

The mathematician's patterns, like the painter's or the poet's must be beautiful; the ideas, like the colours or the words must fit together in a harmonious way. Beauty is the first test: there is no permanent place in this world for ugly mathematics.

A Mathematician's Apology (London 1941),
Godfrey Harold Hardy.

# Semiseparable matrices and the symmetric eigenvalue problem

Raf Vandebril

Abstract. In this thesis one of the basic linear algebra problems is considered, namely the symmetric eigenvalue problem. More precisely we translate the traditional method, based on tridiagonal matrices towards a tool based on semiseparable matrices. Three important parts are considered.

Firstly, the connection between the class of semiseparable matrices and tridiagonal matrices is thoroughly investigated. We define semiseparable matrices, such that the invertible ones have as inverse a tridiagonal matrix, and vice versa. It is shown that the symmetric semiseparable matrices can be represented by a Givens-vector representation having $2n-1$ parameters. Moreover, we show that this representation has nice numerical properties when solving eigenvalue problems.

In the second part of the algorithm, a method is proposed, for reducing, via orthogonal similarity transformations a matrix into a similar semiseparable one. The constructed method inherits some convergence properties, such as subspace iteration, and a type of Lanczos-convergence, which are fully investigated.

In the final part of the thesis, a detailed investigation is made of the $QR$-factorization of semiseparable matrices, unreduced semiseparable matrices, and an implicit $Q$-theorem for semiseparable matrices; leading to an implicit $QR$-algorithm for semiseparable matrices.

The combination of the results of Part 2 and 3 leads to a solver for the symmetric eigenvalue problem. Even more, an adaptation of the results presented in these parts is included, such that also the unsymmetric eigenvalue problem, and the singular value problem, can now be solved via semiseparable matrices.

Numerical experiments are included, and the corresponding software is made available.

D/2004/7515/18

ISBN 90-5682-480-5

# Contents

# Acknowledgements

Having the text of this thesis finished it is time to thank some people, without whom I never would have gotten this far.

First of all I'd like to express my gratitude towards my promoter Marc Van Barel who gave me the opportunity to start a Ph.D. at the department of computer science. All the results presented in this thesis are due to a close collaboration and fruitful discussions we had together. Moreover I'd like to thank him for always taking the necessary time whenever there were any kind of problems. I always enjoyed the nice moments we shared at conferences and at the department.

I thank all the people from the jury, especially Adhemar Bultheel, Ann Haegemans and Sabine Van Huffel for the careful reading of the text. Their corrections and suggestions greatly improved the readability of the thesis.

My colleagues with whom I share or shared the office also contributed in a positive and stimulating way to this thesis. I thank Steven Delvaux for providing some lemmas in Chapter 1. I thank Ellen Van Camp for the explanation of the $QR$-factorization of semiseparable matrices. I always enjoyed the company of Gianni Codevico and thank him for all his help he gave me on different subjects. I thank Nicola Mastronardi for his stimulating enthusiasm. Many of the results in this thesis are in close collaboration with him.

It has always been a pleasure for me to go to conferences. On many of these conferences I met interesting people with whom I had stimulating discussions. I thank Gene Golub, for the comments he gave on my talk about the definition of semiseparable matrices in Bari (2003). It was pleasant for me to get to know Klaus Diepold on the conference in Cadzand (2003). I also thank both of them for inviting me to visit them, which I will certainly do.

I thank Bernard Mourrain for giving me the opportunity to visit his group in Sophia Antipolis for a month. I enjoyed my stay there, and I thank Olivier Ruatta for creating a nice atmosphere during my visit.

Many other people contributed to this thesis. I thank Bart Verleye, with whom I changed the first definition of a representation to the current form. Thanks the people from the library in Leuven, especially Monique Kumps and Christophe Naessens, who located many of the papers presented in the third chapter. Peter Kravanja provided me his LaTeX version of his thesis. This saved me lots of time. I thank the system administrators and the secretaries for all their help. Sam Michiels helped me a lot with all the information he provided for writing a thesis.

The research which led to this thesis was partially supported by the Fund for Scientific Research–Flanders (FWO–V), project “SMA: Structured Matrices and their Applications”, grant #G.0078.01, and project “ANCILA: Asymptotic aNalysis of the Convergence behavior of Iterative methods in numerical Linear Algebra”, grant #G.0176.02, by the K.U.Leuven (Bijzonder Onderzoeksfonds), project “SLAP: Structured Linear Algebra Package”, grant #OT/00/16, by the Belgian Programme on Interuniversity Poles of Attraction, initiated by the Belgian State, Prime Minister’s Office for Science, Technology and Culture, and by MIUR, grant #2002014121.

Finally I thank all my friends for all the moments we shared. I sincerely thank my family and especially my parents for all the opportunities they gave me, and moreover for always being there whenever I needed them. I am also very grateful to Els for all her patience, love and comfort.

Raf,
3rd May 2004,
Leuven.

# Nederlandse samenvatting

In dit proefschrift wordt één van de problemen in de lineaire algebra bestudeerd, namelijk het symmetrisch eigenwaardeprobleem. Om dit probleem op te lossen, vertalen we het traditionele algoritme, dat gebruik maakt van tridiagonaalmatrices, naar de context van semiseparabele matrices.

In het eerste deel van de thesis bekijken we in detail wat semiseparabele matrices zijn en hoe we ze kunnen voorstellen. Verder worden nog enkele toepassingen van semiseparabele matrices bekeken, en ook een historisch overzicht wordt gegeven.

In hoofdstuk 1 wordt aandacht besteed aan de definitie van semiseparabele matrices. Gebruikmakend van het "nulliteitstheorema" wordt aangetoond dat inverteerbare semiseparabele matrices als inverse een tridiagonaalmatrix hebben. Voorts wordt met behulp van dit theorema de uitbreiding naar hogere klassen van semiseparabele matrices onderzocht. Op deze wijze worden dan $\{p,q\}$-semiseparabele matrices, en $\{p\}$-Hessenberggelijke matrices gedefinieerd. Voor de inverteerbare matrices in bovengenoemde klassen zijn hun inversen respectievelijk $\{p,q\}$-band matrices en $\{p\}$-Hessenberg matrices. Gebruikmakend van het "nulliteitstheorema" kunnen we op een eenvoudige manier ook decompositieëigenschappen van de semiseparabele matrices aantonen. Meer precies kunnen er dan voorspellingen gedaan worden over de structuur van de matrices $Q, R, L$ en $U$ die in de $QR$-decompositie en $LU$-decompositie van de semiseparabele matrices voorkomen. Het laatste belangrijke punt dat in dit hoofdstuk behandeld wordt is de vergelijking tussen de alternatieve definitie, voorgesteld in dit proefschrift, en de vaak gebruikte definitie met behulp van generatoren. Er wordt aangetoond dat de nieuw gedefinieerde klasse een uitbreiding is van de traditionele generator-voorstelbare klasse van semiseparabele matrices. Tevens wordt er ook aangetoond dat de nieuwe klasse gesloten is onder puntsgewijze convergentie.

Het $2^{\text{de}}$ hoofdstuk onderzoekt in detail hoe we de nieuwe, ruimere klasse van semiseparabele matrices op een efficiënte wijze kunnen voorstellen. Een nieuwe representatie wordt ontwikkeld. Gebruikmakend van een aantal Givens rotaties en een vector komen we tot $2n-1$ nodige parameters. Enkele algoritmen gebaseerd op de nieuwe representatie worden aangereikt: een snelle vermenigvuldiging van een semiseparabele matrix met een vector en een algoritme om de determinant te berekenen.

In het $3^{\text{de}}$ hoofdstuk worden enkele historische toepassingen van semiseparabele matrices besproken. De eerste toepassing komt uit de theorie van "oscillatiematrices", en behandelt kort hoe een semiseparabele matrix als zo een oscillatiematrix kan beschouwd worden. Ook het verband met matrices uit de statistiek, integraal vergelijkingen en orthogonale rationale functies wordt kort onderzocht. Bovendien bevat dit hoofdstuk een ruime literatuurstudie over semiseparabele matrices en verschillende toepassingen en algoritmen ervoor.

In het volgende deel worden algoritmen onderzocht die matrices omzetten naar de semiseparabele vorm. Hoofdstuk 4 geeft 3 constructieve methodes om, met behulp van Givens en Householder transformaties matrices om te zetten naar ofwel een symmetrisch semiseparabele matrix, ofwel een Hessenberggelijke matrix ofwel een bovendriehoekssemiseparabele matrix. De convergentieëigenschappen van de bovenvermelde reductie-algoritmen worden uitvoerig onderzocht in hoofdstuk 5. Twee types van convergentiegedragingen worden aangetoond. Eerst en vooral, analoog zoals in het tridiagonale geval, bevat het reeds gereduceerde deel van de matrices de Lanczos-Ritz waarden als eigenwaarden. Ten tweede wordt op het reeds gereduceerde deel van de matrices een soort deelruimte-iteratie uitgevoerd. De interactie tussen deze twee gedragingen wordt uitgebreid onderzocht. Numerieke experimenten die de theoretische behandeling staven worden in hoofdstuk 6 samen met details over de implementatie aangeboden.

Het 3$^{\text{de}}$ en tevens laatste deel van de thesis behandelt de eigenlijke $QR$-algoritmen om de eigenwaarden en/of singuliere waarden te berekenen. In hoofdstuk 7 worden de verschillende theoretische aspecten gerelateerd met zo'n algoritme onderzocht. De verschillende onderwerpen die hier aan bod komen zijn de volgende: de $QR$-factorisatie van semiseparabele-plus-diagonaalmatrices; onreduceerbare vorm voor semiseparabele matrices; het behoud van de semiseparabele structuur na het uitvoeren van een stap van het $QR$-algoritme; het terugbrengen van matrices tot de ongereduceerde vorm; en een impliciet $Q$-theorema voor deze matrices. Gebruikmakend van de angereikte materie, worden de echte impliciete $QR$-algoritmen geconstrueerd in hoofdstuk 8. Gebaseerd op zogenaamde "chase" technieken worden impliciete algoritmen voor zowel het symmetrisch als onsymmetrisch geval, als voor de singuliere waarden geconstrueerd. Als laatste hoofdstuk in dit deel worden enkele implementatiedetails evenals numerieke voorbeelden besproken.

Een combinatie van de resultaten uit de verschillende delen geeft ons een algoritme om het symmetrisch eigenwaardeprobleem op te lossen gebruikmakend van semiseparabele matrices. Bovendien zijn alle programma's in MATLAB versie beschikbaar.

# Deel 1

Het eerste deel van de thesis kan als een uitgebreide inleiding tot semiseparabele matrices beschouwd worden. Een beschouwing van de definitie, een onderzoek naar een efficiënte voorstelling, en een historisch overzicht vormen de hoofdzaken in het eerste deel.

## 1. Semiseparabele en gerelateerde matrices, definities en eigenschappen

### *De definitie*

In dit eerste hoofdstuk van de thesis wordt vooral aandacht besteed aan de klasse van gestructureerde rang matrices, waarvan semiseparabele matrices een deelklasse vormen.

DEFINITIE. Zij $A$ een $m \times n$ matrix en zij $M$ de verzameling getallen $\{1, 2, \ldots, m\}$ en $N$ de verzameling getallen $\{1, 2, \ldots, n\}$. Laat $\alpha$ en $\beta$ twee niet-lege deelverzamelingen van $M$ en $N$ zijn. Met $A(\alpha; \beta)$ noteren we dan de deelmatrix van $A$, met rij-indices in $\alpha$ en kolom indices in $\beta$.

Een structuur $\Sigma$ wordt gedefinieerd als zijnde een niet-leeg deel van $M \times N$. Gebruikmakend van een zekere structuur kunnen we een gestructureerde rang van een matrix $A$ definiëren als (met $\alpha \times \beta$ bedoelt men de productverzameling $\{(i,j)|i \in \alpha, j \in \beta\}$):

$$\mathrm{r}(\Sigma; A) = \max\{\mathrm{rank}\,(A(\alpha;\beta))\,|\alpha \times \beta \subseteq \Sigma\}.$$

Gebruikmakend van de volgende structuren, kunnen we eenvoudig semiseparabele matrices definiëren.

DEFINITIE. Zij $M = \{1, \ldots, m\}$ en $N = \{1, \ldots, n\}$. Dan beschouwen we de volgende structuren:

- De deelverzameling
$$\Sigma_l = \{(i,j)|i \geq j, i \in M, j \in N\}$$
wordt de lagerdriehoeksstructuur genoemd.
- De deelverzameling
$$\Sigma_l^{(p)} = \{(i,j)|i > j - p, i \in M, j \in N\}$$
wordt de p-lagerdriehoeksstructuur genoemd.

Op een analoge manier kunnen we zo ook de bovendriehoeksstructuren definiëren. De gestructureerde rang die correspondeert met bijvoorbeeld de lagerdriehoeksstructuur noemt men kortweg de lagerdriehoeksrang. Gebruikmakend van deze nieuwe definities kunnen we heel eenvoudig semiseparabele en Hessenberggelijke matrices definiëren.

DEFINITIE. Een vierkante matrix $S$ van grootte $n \times n$, wordt een $\{p, q\}$-semiseparabele matrix genoemd, als hij volgende structuureigenschappen bezit:

$$\begin{aligned} \mathrm{r}\left(\Sigma_l^{(p)}; S\right) &\leq p \\ \mathrm{r}\left(\Sigma_u^{(-q)}; S\right) &\leq q. \end{aligned}$$

Verder in de inleiding wordt met een semiseparabele matrix een $\{1,1\}$-semiseparabele matrix bedoelt, tenzij anders vermeld.

DEFINITIE. Een matrix $A$ wordt een boven $\{p\}$-Hessenberggelijke matrix genoemd als de $p$-lager triangulaire rang van $A$ kleiner of gelijk is aan $p$:

$$\mathrm{r}\left(\Sigma_l^{(p)}; A\right) \leq p.$$

Analoog als voor de semiseparabele matrices wordt met een Hessenberggelijke matrix een $\{1\}$-Hessenberggelijke bedoeld.

### *Het nulliteitstheorema en enkele toepassingen*

Het nulliteitstheorema relateert op eenvoudige wijze rang structuren van een matrix aan zijn inverse:

STELLING (Het nulliteitstheorema). *Veronderstel dat we een inverteerbare matrix $A \in \mathbb{R}^{n \times n}$ hebben die we als volgt verdelen*

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}$$

*waarbij de submatrix $A_{11}$ grootte $p \times q$ heeft. De inverse van de matrix $A$ namelijk $B$, wordt als volgt verdeeld*

$$B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

*met de matrix $B_{11}$ van grootte $q \times p$. Dan zijn de nulliteiten $\mathrm{n}(A_{11})$ en $\mathrm{n}(B_{22})$ aan elkaar gelijk.*

Deze stelling wordt in het proefschrift op twee verschillende wijzen bewezen. Gebruikmakend van deze stelling kan men op zeer eenvoudige wijze aantonen, dat de inverse van een tridiagonaalmatrix een semiseparabele matrix is, dat de inverse van een $\{p\}$-Hessenberg matrix een $\{p\}$-Hessenberggelijke is, dat de inverse van een $\{p, q\}$-band matrix een $\{p, q\}$-semiseparabele matrix is. Ook andere stellingen, voor meer algemene gestructureerde rang matrices worden geformuleerd.

### *Veralgemeningen van het nulliteitstheorema*

Het nulliteitstheorema kan op eenvoudige wijze vertaald worden zodat het in een iets gewijzigde vorm geldig blijft voor bijvoorbeeld de $LU$ en $QR$-decompositie van gestructureerde rang matrices. Dit gewijzigde theorema kan dan de structuur van deelmatrices van de matrices $L, U, Q$ en $R$ voorspellen aan de hand van de rangstructuur van de originele, gefactoriseerde matrix $A$.

### *De vaak gebruikte definitie van semiseparabele matrices onderzocht*

De laatste sectie in dit hoofdstuk onderzoekt de vaak gebruikte definitie van generator-voorstelbare semiseparabele matrices. Kortweg kan men zeggen dat matrices die aan deze definitie voldoen, een boven en een benedendriehoeksstuk hebben dat afkomstig is van 2 rang 1 matrices. De klasse van semiseparabele matrices die hieraan voldoen, kunnen dan eenvoudig gereconstrueerd worden, door het kennen van de voortbrengers van de rang 1 matrices.

Enkele vaak voorkomende misverstanden betreffende deze klasse van matrices worden onderzocht en duidelijke verschillen tussen de hoger gedefinieerde semiseparabele matrices en deze klasse worden aangetoond.

STELLING. *Een semiseparabele matrix $S$ kan altijd geschreven worden als een blok diagonaalmatrix, waarvan de blokken generator-voorstelbare semiseparabele matrices zijn.*

Een gevolg van de voorgaande stelling met verregaande numerieke gevolgen, is het feit, dat de klasse van generator-representeerbare semiseparabele matrices niet gesloten is voor de puntsgewijze convergentie. Dit betekent ondermeer, dat deze klasse van matrices, niet geschikt is om een $QR$-algoritme op toe te passen, aangezien de matrix na 1 stap van het $QR$-algoritme niet noodzakelijk tot dezelfde klasse meer behoort. Er wordt aangetoond dat de puntsgewijze sluiting van de klasse van generator-representeerbare semiseparabele matrices de klasse van semiseparabele matrices is zoals hoger gedefinieerd.

## 2. De representatie van semiseparabele matrices

Zoals hoger vermeld, is de klasse van semiseparabele matrices die in de thesis beschouwd wordt, iets ruimer dan de klasse van generator-representeerbare semiseparabele matrices. De voorstelling van deze laatste klasse is niet uitbreidbaar naar de nieuwere definitie. Er is dus nood aan een meer algemenere voorstelling voor de semiseparabele matrices.

Na, op wiskundige wijze een representatie voor een klasse van matrices geformuleerd te hebben, worden drie types representaties onderzocht. Eerst en vooral wordt de klassieke generatoren voorstelling bekeken. Het blijkt dat deze klasse, zoals reeds vermeld, niet de ganse klasse van semiseparabele matrices kan voorstellen. Een tweede type representatie maakt gebruik van de diagonaal en de subdiagonaal. Deze voorstelling blijkt echter ook niet toereikend. De laatst behandelde voorstelling dekt de ganse klasse van semiseparabele matrices. Ze bestaat uit een rij van Givens transformaties en een vector. Later zullen numerieke experimenten aantonen dat deze voorstelling numeriek stabiel is voor het oplossen van eigenwaardeproblemen, in tegenstelling tot de voorstelling met behulp van generatoren. Verder wordt nog kort melding gemaakt van een recent gebruikte voorstelling. Deze voorstelling gebruikt meer parameters dan de hier voorgestelde “Givens-vector” voorstelling en wordt daarom niet verder behandeld. Verschillende voorbeelden en een manier om de voorstelling op een numeriek stabiele wijze te berekenen worden aangereikt. Als laatste, worden om de bruikbaarheid van de definitie aan te tonen, twee algoritmen die gebruik maken van de nieuwe voorstelling gegeven. Allereerst wordt er een matrix-vector vermenigvuldiging voorgesteld, die $O(n)$ bewerkingen vraagt. Als tweede wordt ook een $O(n)$ algoritme voorgesteld om de determinant van een semiseparabele matrix te berekenen.

## 3. Een overzicht van semiseparabele matrices

Het derde hoofdstuk geeft een kort historisch overzicht over het gebruik van semiseparabele matrices. Ook een uitgebreide referentielijst is toegevoegd, om de geïnteresseerde lezer een uitgebreid gamma van algoritmen, theoretische beschouwingen en toepassingen aan te bieden.

### ***Trillingsmatrices***

Het boek van Gantmacher & Krein [**83**] wordt vaak als het eerste boek geciteerd, waarin de semiseparabele matrices geïntroduceerd werden. In het boek werden ze vernoemd als “single-pair” of “one-pair” matrices. In deze sectie van de doctoraatstekst worden verschillende aspecten toegelicht van deze matrices. Het kader van de trillings- of bewegingsmatrices, waarin deze matrices terug te vinden zijn wordt kort toegelicht en enkele voorbeelden worden gegeven, waaronder de "single-pair" matrices. Vervolgens wordt aangetoond dat de inverse van een "single-pair" matrix een onreduceerbare tridiagonaalmatrix is. Het bewijs maakt gebruik van de determinants-formules zoals die destijds vaak gebruikt werden. De fysische interpretatie wordt kort toegelicht door aan te tonen dat de "single-pair" matrices corresponderen met de zogenoemde "influentiematrices" komende van trillingen van een snaar. Als laatste wordt zeer kort aangegeven, wat de fysische interpretatie van de eigenwaarden en eigenvectoren van deze klasse van matrices is.

#### *Covariantiematrices*

Enkele jaren na het boek van Gantmacher & Krein maakten de semiseparabele matrices ook hun intrede in een andere tak van de wiskunde, namelijk de statistiek. Verschillende covariantiematrices, gekoppeld aan verschillende kansverdelingen, zijn semiseparabele-plus-diagonaalmatrices. Voor deze matrices zijn vaak ook de eigenwaarden en de eigenvectoren van belang.

#### *Discretisatie van integraal vergelijkingen & Orthogonale rationale functies*

Semiseparabele matrices komen ook voor als discretisatiematrices van integraalvergelijkingen met een "Green's" kern. Evenals tridiagonaalmatrices een drie-terms recursie relatie bepalen, hebben ook semiseparabele plus diagonaalmatrices hun analoge in de theorie van orthogonale rationale functies.

#### *Historisch overzicht*

Als laatste sectie in dit hoofdstuk wordt een uitgebreid overzicht gegeven van artikels en boeken, op een of andere wijze gerelateerd aan semiseparabele matrices. Een korte beschrijving van de inhoud van de publicaties geeft aan, dat er een grote hoeveelheid aan algoritmen en theoretische beschouwingen voor semiseparabele matrices bestaan.

# Deel 2

Het tweede deel van de tekst behandelt verschillende algoritmen, om matrices om te zetten naar semiseparabele vorm. De algoritmen worden besproken, en ook een uitgebreid onderzoek naar het convergentiegedrag is bijgevoegd.

## 4. Verscheidene algoritmen om matrices om te zetten naar de semiseparabele vorm

#### *Een orthogonale gelijkvormigheidstransformatie van een symmetrische matrix naar een symmetrisch semiseparabele matrix*

De volgende stelling kan in het licht van het symmetrisch eigenwaardeprobleem als de belangrijkste stelling gezien worden van dit hoofdstuk.

STELLING. *Zij $A$ een symmetrische $n\times n$ matrix. Dan bestaat er een orthogonale matrix $U$, zondanig dat*

$$U^T AU = S,$$

*waarbij $S$ een semiseparabele matrix is.*

Het bijgevoegde bewijs is constructief, en werkt volgens eindige inductie. Voor stap 1 in het iteratief proces starten we met de matrix $A^{(0)}$. De matrix in het begin van stap $k$, namelijk $A^{(k)}$ heeft de volgende vorm:

$$A^{(k)} = \begin{pmatrix} A_k & R_k^T \\ R_k & S_k \end{pmatrix}$$

waarbij dan de matrix $R_k$ van grootte $k \times (n-k)$ is en van rang 1, en $S_k$ is een vierkante $k \times k$ matrix die al in semiseparabele vorm is. Elke stap van de inductie procedure groeit de dimensie van de matrix $S_k$, en worden 1 rij en 1 kolom toegevoegd aan de semiseparabele structuur. De matrix $S_k$ heeft dus dimensie $(k+1) \times (k+1)$.

***Een orthogonale gelijkvormigheidstransformatie naar een Hessenberggelijke matrix & Een orthogonale transformatie naar een bovendriehoekssemiseparabele matrix***

Op een analoge wijze als de vorige stelling worden dan in de volgende secties, volgende twee stellingen aangetoond. Eerst en vooral kan een gelijkaardig algoritme als hierboven gebruikt worden om niet-symmetrische matrices om te zetten naar gelijkvormige Hessenberggelijke matrices. En een iets gewijzigd procédé kan aangewend worden, om gebruikmakend van orthogonale transformaties een matrix om te zetten naar een bovendriehoekssemiseparabele matrix.

STELLING. *Zij $A$ een $n \times n$ matrix. Dan bestaat er een orthogonale matrix $U$ zodanig dat*

$$U^T AU = H,$$

*waarbij $H$ een Hessenberggelijke matrix voorstelt.*

STELLING. *Zij $A \in \mathbb{R}^{m\times n}$, $m \geq n$. Dan bestaan er twee orthogonale matrices $U \in \mathbb{R}^{m\times m}$ en $V \in \mathbb{R}^{n\times n}$ zodat*

$$U^T AV = \begin{pmatrix} S_u \\ 0 \end{pmatrix},$$

*waarbij $S_u$ een bovendriehoekssemiseparabele matrix voorstelt.*

## 5. Convergentieëigenschappen van de verschillende reductie-algoritmen

Vooraleer we twee verschillende convergentiegedragingen van de verschillende reductie-algoritmen gaan onderzoeken, wordt er een iets breder kader voor het Lanczos-convergentiegedrag geformuleerd. Gebruikmakend van dit kader kan het Lanczos-convergentiegedrag eenvoudig aangetoond worden voor verschillende reductie-algoritmen.

### *De Arnoldi-Ritz waarden in een orthogonale gelijkvormigheidstransformatie*

Nadat eerst de nodige voorwaarden afgeleid zijn waaraan een gelijkvormigheidstransformatie voldoet opdat het reeds gereduceerde deel in de matrix de Arnoldi-Ritz waarden bevat, kunnen we aantonen dat dit ook voldoende voorwaarden zijn.

Het type gelijkvormigheidstransformatie dat wij beschouwen is als volgt. Laat ons starten met een matrix $A^{(0)}$, welke via een orthogonale gelijkvormigheidstransformatie omgezet wordt in de matrix $A^{(1)} = Q_0^T A^{(0)} Q_0$. Merk op dat de matrix $Q_0$ niet uniek is en vaak gelijk aan de identiteitsmatrix genomen wordt. Een andere keuze wordt gebruikt om het convergentiegedrag te beïnvloeden. De andere orthogonale transformaties $Q_k$ komen voort uit het reductie-algoritme, en de matrix $A^{(k+1)}$ voldoet aan $A^{(k+1)}$,

$$A^{(k+1)} \quad = \quad Q_k^T A^{(k)} Q_k$$

De transformaties zijn zodanig dat de matrix $A^{(k+1)}$ van de volgende vorm is:

$$\left(\begin{array}{c|c} R_{k+1} & \times \\ \hline \times & A_{k+1} \end{array}\right)$$

met $R_{k+1}$ het $(k+1) \times (k+1)$ deel van de matrix dat al in de gepaste vorm is bv. tridiagonaal, semiseparabel, Hessenberg, etc.

Stelling. *Veronderstel dat we een gelijkvormigheidstransformatie op de matrix $A$ uitvoeren (van de vorm zoals hierboven beschreven, met $A^{(0)} = A$):*

$$Q_0 e_1 = \pm \frac{v}{\|v\|} \text{ and } Q_0^T A^{(0)} Q_0 = A^{(1)}.$$

*En veronderstel dat we in stap $k$ van het reductie-algoritme de volgende eigenschappen hebben*

- *De orthogonale matrix $Q_k^T$, heeft een nulblok van grootte $(n-k-1) \times k$ in het links beneden stuk;*
- *De matrix $\tilde{A}^{(k)} = Q_k^T A^{(k)}$ heeft een nulblok van grootte $(n-k-1) \times k$ in het links beneden stuk;*

*Dan volgt daaruit dat voor de matrix $A^{(k)}$ verdeeld als*

$$A^{(k)} = \left(\begin{array}{c|c} R_k & \times \\ \hline \times & A_k \end{array}\right),$$

*met $R_k$ van grootte $k \times k$, dat de eigenwaarden van $R_k$ de Arnoldi-Ritz waarden zijn t.o.v de Krylov deelruimte $\mathcal{K}_k(A, v) = \langle v, Av, \ldots, A^{k-1}v \rangle$.*

### *Convergentieëigenschappen van de orthogonale gelijkvormigheidstransformatie naar symmetrisch semiseparabele vorm*

Gebruikmakend van de stelling hierboven kan er zeer eenvoudig aangetoond worden dat het reeds gereduceerde deel van de matrix, dat wil zeggen, het deel van de matrix dat al symmetrisch semiseparabel is, de Lanczos-Ritz waarden als eigenwaarden heeft. Dit is het eerste convergentiegedrag dat het reductie-algoritme vertoont. Er is echter nog een tweede soort convergentiegedrag dat interageert met het Lanczos-convergentiegedrag.

Op het laatste blok van de matrix, namelijk het deel dat al in semiseparabele vorm is, wordt namelijk ook een deelruimte-iteratie uitgevoerd. Omdat het gereduceerde semiseparabele deel na elke iteratiestap groeit zal ook de deelruimte waar de iteratie op uitgevoerd wordt bij elke stap groter worden. Dit wordt een geneste deelruimte-iteratie genoemd.

De interactie tussen de twee gedragingen kan heel eenvoudig geïnterpreteerd worden. Wanneer men het reductie-algoritme meer gedetailleerd bekijkt, merkt men dat vooraleer de deelruimte-iteratie uitgevoerd wordt, de Lanczos-Ritz waarden al in dit blok zitten. Kortweg wil dat zeggen dat de deelruimte-iteratie uitgevoerd wordt op een semiseparabele matrix die de Lanczos-Ritz waarden als eigenwaarden bevat. De deelruimte-iteratie kan dus enkel werken als de Lanczos-Ritz waarden al op voldoende wijze de dominante eigenwaarden van de semiseparabele matrix benaderen.

***Convergentieëigenschappen van de reductie naar Hessenberggelijke vorm & Convergentieëigenschappen van de reductie naar bovendriehoekssemiseparabele vorm***

Analoog als voor de reductie naar een gelijkvormige semiseparabele matrix, kunnen hier dezelfde convergentiegedragingen afgeleid worden. Voor de reductie naar bovendriehoeksvorm moet men wel opletten dat de convergentieëigenschappen niet toepasbaar zijn op de eigenwaarden maar in plaats daarvan werken de gedragingen in op de singuliere waarden.

## 6. Implementaties en numerieke experimenten

In dit afsluitende hoofdstuk van deel 2, worden eerst en vooral details over de implementatie van de verschillende algoritmen gegeven. Vervolgens worden verschillende numerieke experimenten uitgevoerd met de hoger vermelde implementatie, die gebaseerd is op de Givens-vector representatie zoals die in hoofdstuk 2 beschreven werd. De eerste reeks van experimenten onderzoekt de interactie tussen het Lanczos-convergentiegedrag en de deelruimte-iteratie. De verschillende experimenten geven duidelijk aan hoe de deelruimte-iteratie beïnvloed wordt door het Lanczos-convergentiegedrag. In een tweede experiment wordt de nauwkeurigheid van het bovenstaande reductie-algoritme vergeleken met de reductie naar de tridiagonale vorm. Het blijkt experimenteel dat de aanpak via semiseparabele matrices meer nauwkeurige resultaten genereert. Het laatste experiment vergelijkt de deflatieëigenschappen, van de traditionele reductie naar tridiagonale vorm, met de nieuwe reductie. Hieruit blijkt duidelijk de belangrijke invloed van de deelruimte-iteratie, die ook verscheidene clusters van eigenwaarden weet te lokaliseren. Dit gedrag kan niet geobserveerd worden bij de reductie naar tridiagonaalmatrices.

# Deel 3

In het derde en laatste deel van de thesis worden de impliciete $QR$-algoritmen voor semiseparabele matrices ontwikkeld. Eerst worden verschillende theoretische resultaten die nodig zijn voor een $QR$-algoritme aangetoond, waarna de echte $QR$-algoritmen afgeleid worden. Ook details over de implementatie en numerieke experimenten worden gegeven.

## 7. Theoretische resultaten nodig voor het ontwikkelen van $QR$-algoritmen voor semiseparabele matrices

Dit hoofdstuk bevat al de basisingrediënten die nodig zijn voor het ontwikkelen van $QR$-algoritmen voor semiseparabele matrices. Om al de resultaten in dit hoofdstuk in een zo algemeen mogelijk kader te plaatsen worden de verschillende theoretische beschouwingen toegepast op de algemene klasse van Hessenberggelijke matrices.

***Een $QR$-factorizatie voor Hessenberggelijke-plus-diagonaalmatrices***

In deze sectie wordt uit de doeken gedaan hoe we door 2 reeksen van Givens transformaties toe te passen op een Hessenberggelijke-plus-diagonaalmatrix, deze matrix kunnen omzetten in een bovendriehoeksmatrix. Dit deel is gebaseerd op de paper van Mastronardi, Van Camp en Van Barel. Een eerste reeks van $n-1$ Givens transformaties werkt op de Hessenberggelijke matrix, van beneden naar boven en zet daarbij deze matrix om in een bovendriehoeksmatrix. Deze Givens transformaties zetten de diagonaalmatrix echter om in een boven Hessenberg matrix, wat ervoor zorgt dat we nog een tweede reeks van Givens transformaties nodig hebben om de Hessenberg matrix om te zetten in een bovendriehoeksmatrix.

***Ongereduceerde Hessenberggelijke matrices***
***&***
***De reductie naar ongereduceerde vorm***

In de tweede en vierde sectie van dit hoofdstuk worden bovenstaande onderwerpen behandeld. Het ongereduceerd zijn van een Hessenberggelijke matrix is essentieel voor de werking van een impliciet $QR$-algoritme, net als in het Hessenberg geval. Bij de Hessenberggelijke matrix, zijn er twee eisen die voldaan moeten zijn. Een eerste eis stemt overeen met het klassieke geval, namelijk dat de matrix niet in blok diagonaal vorm mag zijn. Een tweede eis is specifiek voor de klasse van matrices in semiseparabele vorm. Eenvoudigweg zegt deze eis, onder enkele extra condities, dat de semiseparabele structuur zich niet kan uitbreiden boven de diagonaal.

De transformatie van een Hessenberggelijke matrix naar onreduceerbare vorm wordt gedaan via een orthogonale gelijkvormigheidstransformatie, die gebaseerd is op de $QR$-factorisatie, zodanig dat alle nul rijen zich onderaan bevinden.

***Het behoud van de Hessenberggelijke structuur onder een $QR$-stap***

Het behoud van de structuur van de Hessenberggelijke matrix is evenzeer als de onreduceerbaarheid van cruciaal belang. Het behoud van de structuur maakt het mogelijk om herhaaldelijke stappen van $QR$ uit te voeren zodat we een rij van Hessenberggelijke matrices krijgen die convergeert naar een diagonaalmatrix. Het is tevens ook zeer belangrijk dat voor singuliere Hessenberggelijke matrices de structuur behouden blijft. Door het kiezen van een perfecte shift namelijk, wordt in feite een singulier Hessenberggelijke-plus-diagonaalmatrix gecreëerd, waarop we dan een stap van $QR$ zonder shift uitvoeren. De diagonaalmatrix is hier een shift matrix.

Opdat we uiteindelijk verder kunnen werken met deze matrix hebben we dus nodig dat na 1 stap van $QR$ op deze Hessenberggelijke-plus-diagonaalmatrix, we terug een Hessenberggelijke-plus-diagonaal krijgen, met als diagonaal dezelfde shift matrix als waar we van vertrokken. Deze stelling tonen we aan, niet enkele voor een shift matrix als diagonaal, maar er wordt bovendien bewezen dat een willekeurige diagonaal bewaard blijft onder een stap van het $QR$-algoritme. Om deze stelling aan te tonen, worden verschillende eenvoudigere stellingen geformuleerd en bewezen. Volgende twee stellingen zijn de hoofdstellingen in deze sectie.

STELLING. *Veronderstel dat we een Hessenberggelijke matrix $Z$ hebben. We passen een stap van het $QR$-algoritme met shift toe op deze matrix. (Waarbij we de $QR$-factorisatie gebruiken, zoals toegelicht in dit hoofdstuk.)*

$$\left\{ \begin{array}{rcl} Z - \kappa I & = & QR \\ \tilde{Z} & = & RQ + \kappa I. \end{array} \right.$$

*Dan zal de resulterende matrix $\tilde{Z}$ opnieuw een Hessenberggelijke matrix zijn.*

De volgende stelling is iets krachtiger en toont in feite aan dat een willekeurige diagonaalmatrix behouden blijft onder een stap van het $QR$-algoritme.

STELLING. *Veronderstel dat $Z$ een Hessenberggelijke matrix is en $D$ een diagonaalmatrix. Het toepassen van een stap van $QR$ met shift op de matrix $Z + D$ (Met de $QR$-factorisatie uit hoofdstuk 7), resulteert opnieuw in een Hessenberggelijke-plus-diagonaalmatrix, die geschreven kan worden als $\tilde{Z} + D$, $\tilde{Z}$ een Hessenberggelijke matrix, en $D$ dezelfde diagonaalmatrix als hierboven.*

### *Het impliciet $Q$-theorema voor de Hessenberggelijke matrix*

Het impliciet $Q$-theorema kan opgevat worden als het laatste belangrijke theorema voor de ontwikkelling van een impliciet $QR$-algoritme. Gebruikmakend van de onreduceerbare vorm van de Hessenberggelijke matrix wordt de volgende stelling naar analogie met het Hessenberggeval aangetoond.

Laat ons eerst het onreduceergetal associëren aan een Hessenberggelijke matrix.

DEFINITIE. Zij $Z$ een Hessenberggelijke matrix, het onreduceergetal $k$ van $Z$ is het kleinste geheel getal zodat aan minstens 1 van de volgende twee condities voldaan is:

(1) de deelmatrix $S(k + 1 : n; 1 : k) = 0$;
(2) het element $S(k+1, k+2)$ waarbij $k < n-1$ voldoet aan de semiseparable structuur van het benedendriehoeksdeel.

Wanneer de matrix onreduceerbaar is dan is $k = n$

STELLING (impliciet $Q$-theorema voor Hessenberggelijke matrices). *Veronderstel dat we de volgende vergelijkingen hebben,*

$$\begin{array}{rcl} Q_1^T A Q_1 & = & Z \\ Q_2^T A Q_2 & = & X \end{array}$$

*waarbij $Q_1 e_1 = Q_2 e_1$ en de matrices $Z$ en $X$ zijn twee Hessenberggelijke matrices met respectievelijke onreduceergetallen $k_1$ en $k_2$. Zij $k = \min(k_1, k_2)$. Dan hebben we* *dat de eerste $k$ kolommen van $Q_1$ en $Q_2$ tot op het teken na gelijk zijn. Bovendien bestaat er een matrix $V = \mathrm{diag}(1, \pm 1, \pm 1, \ldots, \pm 1)$, van grootte $k \times k$, zodat aan volgende vergelijkingen voldaan is:*

$$\begin{aligned} Q_1(1:k, 1:k) &= Q_2(1:k, 1:k)\, V \\ Z(1:k, 1:k)\, V &= V\, X(1:k, 1:k) \end{aligned}$$

## 8. Impliciete $QR$-algoritmen

In dit hoofdstuk worden voor de klasse van eigenwaardeproblemen, voor het symmetrisch en onsymmetrisch geval, en voor de klasse van singuliere waardeproblemen, verschillende impliciete $QR$-algoritmen ontwikkeld, gebruikmakend van de semiseparabele matrices, i.p.v. de traditionele tridiagonale, Hessenberg of bidiagonale vorm.

### *Een impliciet $QR$-algoritme voor de symmetrisch semiseparabele matrices*

Voor de klasse van symmetrisch semiseparabele matrices wordt er alleerst bestudeerd hoe deze matrices op een meer efficiëntere wijze omgezet kunnen worden naar de onreduceerbare vorm. Vervolgens wordt het impliciet $QR$-algoritme ontwikkeld, in analogie, met de $QR$-factorisatie van de Hessenberggelijke matrix. Laat ons de eerste rij van transformaties uit de factorisatie noteren met $Q_1$, en de tweede reeks van Givens transformaties met $Q_2$. Het impliciet toepassen van deze transformaties op de matrix $S$, bestaat er eerst en vooral in om gelijktijdig links en rechts van deze matrix de Givens transformaties uit te voeren.

Het uitvoeren van de transformatie $Q_1$, langs links en rechts kan op zeer efficiënte wijze gedaan worden. In feite komt het erop neer dat een $QR$-stap zonder shift op de matrix $S$ wordt uitgevoerd. We weten dat de resulterende matrix $Q_1^T S Q_1$ opnieuw symmetrisch semiseparabel is.

Om de tweede stap impliciet uit te voeren, moet enkel de eerste Givens transformatie $G$ van de matrix $Q_2$ bepaald worden. Zo krijgen we de matrix $G^T Q_1^T S Q_1 G$, die de semiseparabele structuur verloren heeft. Vervolgens wordt een reeks van transformaties op de matrix $G^T Q_1^T S Q_1 G$ uitgevoerd zonder de eerste kolom te verstoren , zodanig dat de resulterende matrix opnieuw een semiseparabele matrix is. Gebruikmakend van het impliciet $Q$-theorema kunnen we dan aantonen dat de resulterende matrix equivalent is met de matrix na een stap van het $QR$-algoritme.

### *Een impliciet $QR$-algoritme voor de Hessenberggelijke matrices*

Op een analoge wijze als voor het symmetrisch semiseparabele geval wordt er hier een impliciet $QR$-algoritme voor een Hessenberggelijke matrix geconstrueerd.

### *Een impliciet $QR$-algoritme voor het berekenen van de singuliere waarden via bovendriehoekssemiseparabele matrices*

De laatste sectie van dit hoofdstuk vertaalt het $QR$-algoritme voor het symmetrisch eigenwaardeprobleem naar een impliciet $QR$-algoritme voor het berekenen van de singuliere waarden van matrices. In het traditionele algoritme wordt er een stap van $QR$ uitgevoerd op een bidiagonaalmatrix, door een eerste Givens transformatie uit te voeren die de bidiagonale structuur verstoort. Een reeks van Givens transformaties herstelt dan de verstoorde structuur, met als resultaat een bidiagonaalmatrix, waar een impliciete stap van $QR$ is op uitgevoerd.

Dit algoritme vertaalt zich analoog in het semiseparabele geval. In plaats van op een bidiagonaalmatrix, werken we hier met een bovendriehoekssemiseparabele matrix. Het uitvoeren van een eerste Givens transformatie verstoort de semiseparabele structuur, waarna een reeks van Givens transformaties op de verstoorde matrix wordt uitgevoerd. Deze Givens transformaties herstellen opnieuw de semiseparabele structuur en gebruikmakend van het impliciete $QR$-algoritme, kunnen we dan aantonen dat er een impliciete stap van $QR$ op de bovendriehoekssemiseparabele matrix uitgevoerd is.

## 9. Implementatie van $QR$-gerelateerde algoritmen en numerieke experimenten

### *De implementatie van $QR$-factorisatie van de semiseparabele-plus-diagonaalmatrices*

De implementatie hier beschreven is voor een semiseparabele-plus-diagonaalmatrix, omdat deze implementatie iets moeilijker is dan voor het eenvoudigere geval van de Hessenberggelijke-plus-diagonaalmatrix. Gebruikmakend van de hier gegeven details is het mogelijk om het algoritme te herschrijven voor het Hessenberggelijke-plus-diagonaal geval. Voor het implementeren van de $QR$-factorisatie van de matrix, werd opnieuw gebruik gemaakt van de Givens-vector voorstelling. Het algoritme bestaat uit twee delen.

Eerst wordt de eerste rij van Givens transformaties toegepast die een Hessenberg matrix construeert. Bovendien zorgt de structuur van de toegepaste orthogonale transformatie ervoor, dat het bovendriehoeksdeel van de resulterende Hessenberg matrix semiseparabel is en van semiseparabiliteits rang 1. Het is dus belangrijk dit deel op een efficiënte wijze voor te stellen. Dit gebeurt met twee maal een Givens-vector voorstelling.

In het tweede deel van het algoritme wordt opnieuw een reeks van Givens transformaties uitgevoerd op de Hessenberg matrix om deze te transformeren naar een bovendriehoeksmatrix. De resulterende bovendriehoeksmatrix kan gezien worden als een strict bovendriehoeksmatrix van semiseparabele vorm plus een diagonaal.

Gebruikmakend van de voorstelling van de bovendriehoeksmatrix is het zeer eenvoudig om achterwaartse substitutie in orde $n$ operaties toe te passen. Op deze manier kunnen we in orde $n$ bewerkingen een stelsel van vergelijkingen met als coëfficiënten matrix een semiseparabele-plus-diagonaalmatrix oplossen.

### *De reductie naar de ongereduceerde vorm & Deflatie na een stap van het $QR$-algoritme*

Heel belangrijk bij de reductie naar ongereduceerde vorm en bij deflatie zijn volgende twee vragen: wanneer is een blok numeriek van rang 1 en wanneer is een getal numeriek nul. Voor het deflatiecriterium wordt het aggressieve en normale deflatiecriterium gehanteerd. Het numeriek van rang 1 zijn, wordt herleid tot een test die binnen in het algoritme voor het terugbrengen naar onreduceerbare vorm wordt uitgevoerd.

### *De implementatie van een stap van $QR$ met en zonder shift*

Zoals reeds vermeld, bestaat een stap van het $QR$-algoritme erin door eerst een stap van het $QR$-algoritme zonder shift uit te voeren, en daarna de “chasing” procedure te starten.

Het implementeren van een $QR$-stap zonder shift is bijna triviaal gebruikmakend van de Givens-vector representatie. Wanneer er namelijk geen shift betrokken is, zijn de Givens transformaties dezelfde transformaties als deze die nodig zijn voor het bovendriehoeks maken van de semiseparabele matrix. Deze stap kan zeer eenvoudig in $O(n)$ bewerkingen uitgevoerd worden.

Het tweede deel is de zogenaamde “chase” procedure. Eerst wordt de semiseparabele matrix verstoord door een Givens transformatie. In feite wordt daardoor de matrix opgedeeld in twee verschillende semiseparabele delen. Een bovenste deel van grootte $2 \times 2$ en en onderste deel van grootte $(n-1) \times (n-1)$. Gebruikmakend van de bijzondere Givens transformaties die in het vorige hoofdstuk aangereikt werden kunnen we nu het bovenste semiseparabele deel laten groeien, en het onderste deel laten afnemen in grootte. Telkens worden de 2 verschillende semiseparabele delen voorgesteld door de Givens-vector representatie.

### *Het berekenen van de eigenvectoren*

Voor het berekenen van de eigenvectoren moet er een onderscheid gemaakt worden tussen twee verschillende problemen. Eerst en vooral het probleem waarbij men alle eigenvectoren wil berekenen en ten tweede waarbij slechts enkele eigenvectoren gevraagd zijn. Het berekenen van alle eigenvectoren kan men best doen door alle uitgevoerde orthogonale transformaties bij te houden en/of uit te voeren op de identiteitsmatrix. Het berekenen van slechts enkele specifieke eigenvectoren wordt met behulp van inverse iteratie gedaan. Voor de inverse iteratie, is het oplossen nodig van een semiseparabel-plus-diagonaalstelsel, wat eerder in dit hoofdstuk aan bod kwam.

### *Numerieke experimenten*

In dit hoofdstuk worden verschillende experimenten uitgevoerd, enerzijds om het algoritme via de semiseparabele matrices te vergelijken met het klassieke algoritme, en anderzijds om de nauwkeurigheid en snelheid van het algoritme te onderzoeken.

In een eerste experiment vergelijkt men het traditionele eigenwaardealgoritme, met het nieuwe algoritme om de eigenwaarden van een symmetrische matrix te berekenen. De geconstrueerde matrix is bijna in de blokdiagonaalvorm wat mogelijk problemen oplevert voor het impliciete $QR$-algoritme. Een vergelijking tussen het maximaal aantal stappen nodig vooraleer een eigenwaarde voldoende nauwkeurig benaderd is, wordt gemaakt.

In een tweede experiment worden de diagonaalelementen na de constructie van de semiseparabele matrix vergeleken met de echte eigenwaarden. Deze echte eigenwaarden zijn in een bijzondere trapvorm gekozen. Het blijkt dat de diagonaalelementen deze eigenwaarden al bijzonder goed benaderen.

Het derde voorbeeld maakt gebruik van de volledige eigenwaardesolver voor de semiseparabele matrices voor het oplossen van enkele probleemmatrices zoals die al in het 6$^{\text{de}}$ hoofdstuk aan bod kwamen.

In een vierde test worden het aggressieve en normale deflatiecriterium voor de symmetrische matrices vergeleken. Het blijkt dat beide criteria even accurate resultaten opleveren.

De laatste reeks van testen vergelijkt de nauwkeurigheid en de snelheid van het nieuwe algoritme om de singuliere waarden te berekenen met de klassieke aanpak via de bidiagonaal vorm.

## 10. Het corresponderende softwarepakket

Samen met deze thesis wordt via het internet een softwarepakket aangeboden voor de programmeeromgeving Matlab. In dit hoofdstuk worden de aangeboden routines beknopt toegelicht. Deze zijn gebaseerd op de implementatiedetails gegeven in deze thesis.

# Outline of the thesis

In this thesis we study one of the basic problems in linear algebra, namely the symmetric eigenvalue problem. In this text we translate the traditional eigenvalue solver, based on tridiagonal matrices, towards the context of semiseparable matrices.

In the first part of the thesis, we consider in detail the class of semiseparable matrices and how they can be represented in an efficient way. Furthermore some traditional applications are considered and an historical overview of semiseparable matrices is given.

In Chapter 1 attention is paid to the definition of semiseparable matrices. Using the "nullity theorem" we prove that the inverse of invertible tridiagonal matrices are semiseparable matrices and vice versa. Furthermore we use this theorem to consider also band matrices and higher rank semiseparable matrices. In this way we define the classes of $\{p\}$-Hessenberg-like and $\{p, q\}$-semiseparable matrices. The inverses of these classes of matrices are respectively $\{p\}$-Hessenberg matrices and $\{p, q\}$-band matrices. Two extensions of the nullity theorem are given to predict the structure of the $Q, R, L$ and $U$ factors in the decomposition of structured rank matrices, e.g., semiseparable matrices. The last important topic in this chapter is a comparison between the class of semiseparable matrices as we define them, and the traditional definition of generator representable semiseparable matrices. Moreover we prove that the new class of semiseparable matrices is closed under the pointwise limit.

The $2^{\text{nd}}$ chapter investigates in detail, how we can represent the new, larger class of semiseparable matrices. A new representation is developed, using a sequence of Givens transformations and a vector. This gives us a representation for a symmetric semiseparable matrix using only $2n - 1$ parameters. Some algorithms based on the new representation are provided: a fast multiplication of a semiseparable matrix and a vector and an algorithm to compute the determinant.

In the $3^{\text{th}}$ chapter some historical applications of semiseparable matrices are considered. The first application arises from the theory of "oscillation matrices" and considers briefly how a semiseparable matrix can be seen as such an oscillation matrix. Also the connection between matrices arising from statistical applications, integral equations and orthogonal rational functions are briefly investigated. Also included in this chapter is a broad study of the literature related to semiseparable matrices. This study contains references of publications on algorithms, theoretical results and applications.

In the second part of the thesis, algorithms transforming matrices to semiseparable form using orthogonal transformations are considered. Chapter 4 presents three constructive methods to transform, via Givens and Householder transformations matrices to either, semiseparable, Hessenberg-like and upper triangular semiseparable form. The convergence properties of the these reduction algorithms are investigated thoroughly in Chapter 5. Two types of convergence behavior are observed. Similar as in the tridiagonal case, we prove that in the reduction to semiseparable form the part of the matrix already in semiseparable form contains the Lanczos-Ritz values. Secondly it is proved that on this already reduced part some kind of nested subspace iteration works. The interaction between these two types of convergence behavior, and the corresponding results are investigated. Numerical experiments showing this interaction and details about this implementation are included in Chapter 6.

The $3^{\text{th}}$ and last part of the thesis considers the actual $QR$-algorithms for computing the eigen- and/or singular values. In Chapter 7, the different theoretical aspects related with this type of algorithms are investigated. The subjects which are considered are the following: the $QR$-factorization of semiseparable plus diagonal matrices; the unreduced form for semiseparable matrices; the preservation of the semiseparable structure under a step of the $QR$-algorithm; the reduction of matrices to the unreduced form; and an implicit $Q$-theorem for Hessenberg-like matrices. Using these theoretical results the real implicit $QR$-algorithms are derived in Chapter 8. Based on the so-called "chase"techniques, implicit $Q$-theorems are derived for the eigenvalues in the symmetric and the unsymmetric case, as well as for singular values. Chapter 9 reveals some details about the implementations, and also some numerical results are included.

Combining the results in the previous parts gives us an algorithm for solving the symmetric eigenvalue problem using semiseparable matrices instead of tridiagonal matrices. Moreover all related routines are implemented in MATLAB and available on the web.

More details on the (symmetric) eigenvalue problem solved using intermediate (tridiagonal) Hessenberg matrices can be found in [**45, 91, 140, 162, 169, 189**] and the references therein. For the solution of the singular value problem using intermediate bidiagonal matrices, we refer the interested reader to [**45, 91, 162, 169**] and the references therein.

# Part 1

# Introduction to semiseparable matrices

CHAPTER 1

# Semiseparable and related matrices, definitions and properties

In this first chapter of the thesis we will pay special attention to the definition of semiseparable matrices. Semiseparable matrices are commonly known as the inverses of tridiagonal matrices. Using techniques from the second and third section we will provide a proof of this statement. Consecutively we will provide decomposition theorems, making it possible to predict the structure of the factors in the $LU$-decomposition and the $QR$-decomposition of semiseparable matrices. Finally we investigate in detail the widespread definition of semiseparable matrices in terms of generators.

In the first section semiseparable matrices, $\{p, q\}$-semiseparable matrices and $\{p\}$-Hessenberg-like matrices will be defined. These matrices are structured matrices, with respect to the rank. This means that certain subblocks within these matrices satisfy certain rank properties, therefore these matrices are often called structured rank matrices. It is clear, that also other types of matrices can be structured rank matrices. In this section we will precisely define what the concepts structure and structured rank of a matrix mean. These two concepts will make it possible to define semiseparable and related matrices in a beautiful way.

In the second section we will prove the nullity theorem in two different ways. The nullity theorem states, that the nullity of a subblock of a certain invertible matrix $A$ has the same nullity as the complementary subblock in the matrix $A^{-1}$. First we prove the theorem directly as Fiedler and Markham did it in [**73**], by multiplying the partitioned matrix $A$ by its correspondingly partitioned inverse, thereby revealing properties of the subblocks, connected to the partitioning. Afterwards we enhance the ideas from Barrett and Feinsilver [**9**] to provide the reader with another proof of the theorem, based on the determinants of matrices. Because the formulation of the theorem is quite abstract, two small corollaries will be formulated, which will give more immediate insight into the structured rank of matrices and their inverses.

The third section is merely based on the papers [**71, 72, 73, 77**]. General theorems describe the behavior of structured ranks of matrices under inversion. The theorems presented, provide tools applicable to different types of matrices, as shown in the various examples. Detailed theorems will be formulated revealing the connection between tridiagonal and semiseparable matrices. These connections can be generalized immediately to the class of $\{p, q\}$-semiseparable, $\{p, q\}$-band, $\{p\}$-Hessenberg and $\{p\}$-Hessenberg-like matrices.

Section 4 provides generalizations of the nullity theorem. Two theorems are formulated, one in terms of the $LU$-decomposition and another in terms of the $QR$-decomposition of structured rank matrices. A first theorem reveals the connection between the structured rank of the $L$ and $U$ factors of the $LU$-decomposition of a matrix $A$ and the structured rank of the original matrix $A$. This theorem is applied to a semiseparable matrix, to predict the structure of the $L$ and $U$ factor. Also another, more direct approach, is mentioned. Via the $LU$-decomposition of the inverse of the semiseparable matrix $S$ one can directly derive the structure of the $L$ and $U$ factors. A second theorem reveals the connection between the structured rank of $Q$ of the $QR$ factorization of a matrix $A$, and the structured rank of the matrix $A$. The theorem is applied to a semiseparable matrix, and again we prove structural properties of the $Q$ and the $R$ factors in two different ways.

In the last section of this chapter, we investigate the commonly used definition of semiseparable matrices in terms of generators, which we will refer to as generator representable semiseparable matrices. First some widespread misunderstandings connected with the generator representable matrices are mentioned, and illustrated with examples. Secondly we will show that the class of generator representable matrices has a theoretical problem, namely diagonal matrices do not belong to this class of matrices, which will lead to numerical instabilities. This theoretical problem is due to the fact that the class of generator representable matrices is not closed for the pointwise convergence, i.e. the limit of a sequence of generator representable matrices is not necessarily a generator representable matrix anymore. As a third topic in this section we will prove that the pointwise closure of the class of generator representable semiseparable is the class of semiseparable matrices as defined in the first section. Moreover, we prove that every semiseparable matrix can be written as a block diagonal matrix, for which all the blocks are generator representable matrices.

## 1. Definitions for semiseparable and related matrices

Semiseparable matrices are structured rank matrices, i.e. all submatrices corresponding to a structure satisfy certain rank properties. Structure and structured ranks are defined as follows.

DEFINITION 1. Let $A$ be an $m \times n$ matrix. Denote with $M$ the set of numbers $\{1, 2, \ldots, m\}$ and with $N$ the set of numbers $\{1, 2, \ldots, n\}$. Let $\alpha$ and $\beta$ be nonempty subset of $M$ and $N$, respectively. Then, we denote with the matrix $A(\alpha; \beta)$ the submatrix of $A$ with row indices in $\alpha$ and column indices in $\beta$. A structure $\Sigma$ is defined as a nonempty subset of $M \times N$. Based on a structure, the structured rank $\mathrm{r}(\Sigma; A)$ is defined as (where $\alpha \times \beta$ denotes the set $\{(i,j)|i \in \alpha, j \in \beta\}$):

$$\mathrm{r}(\Sigma; A) = \max\{\mathrm{rank}\,(A(\alpha; \beta))\,|\alpha \times \beta \subseteq \Sigma\}.$$

Before giving the definition of a semiseparable matrix we have to specify the corresponding structure. In Section 3 some more structures will be constructed and investigated.

DEFINITION 2. For $M = \{1, \ldots, m\}$ and $N = \{1, \ldots, n\}$ we define the following structures:

- The subset
$$\Sigma_l = \{(i,j) | i \geq j, i \in M, j \in N\}$$
is called the lower triangular structure; in fact the elements of the structure correspond to the indices from the lower triangular part of the matrix.
- The subset
$$\Sigma_{wl} = \{(i,j) | i > j, i \in M, j \in N\}$$
is called the weakly lower triangular structure.
- The subset
$$\Sigma_l^{(p)} = \{(i,j) | i > j - p, i \in M, j \in N\}$$
is called the $p$-lower triangular structure and corresponds with all the indices of the matrix A, below the $p$th diagonal. The 0th diagonal corresponds to the main diagonal, while the $p$th diagonal refers to the $p$th superdiagonal (for $p > 0$) and the $-p$th diagonal refers to the $p$th subdiagonal (for $p > 0$).

Note that $\Sigma_l^{(1)} = \Sigma_l$, $\Sigma_l^{(0)} = \Sigma_{wl}$ and $\Sigma_{wl} \subsetneq \Sigma_l$. Note that the structure $\Sigma_l^{(p)}$ for $p > 1$ contains all the indices from the lower triangular part, but also contains some superdiagonals of the strictly upper triangular part of the matrix. The weakly lower triangular structure is sometimes also called the strictly lower triangular structure or the subdiagonal structure. For the upper triangular part of the matrix, the structures $\Sigma_u, \Sigma_{wu}$ and $\Sigma_u^{(p)}$ are defined similarly, and are called the upper triangular structure, the weakly upper triangular structure and the $p$-upper triangular structure, respectively. The structured rank connected to the lower triangular structure, is called the lower triangular rank. Similar definitions are assumed for the other structures.

With the above defined structures we can define semiseparable and closely related matrices.

Definition 3. An $n \times n$ matrix $S$ is called a $\{p,q\}$-semiseparable matrix, with $p \geq 0$ and $q \geq 0$, if the following two properties are satisfied:
$$\begin{aligned} \mathrm{r}\left(\Sigma_l^{(p)}; S\right) &\leq p \\ \mathrm{r}\left(\Sigma_u^{(-q)}; S\right) &\leq q. \end{aligned}$$
This means that the $p$-lower triangular rank is less than or equal to $p$ and the $q$-upper triangular rank is less than or equal to $q$.

The above definition says that the maximum rank of all subblocks which one can take out of the matrix below the $p$th superdiagonal is less than or equal to $p$ and the maximum rank of all subblocks which one can take above the $q$th subdiagonal is less than or equal to $q$. When speaking about a $\{p\}$-semiseparable matrix or a semiseparable matrix of semiseparability rank $p$, we mean a $\{p,p\}$-semiseparable matrix. When briefly speaking about a semiseparable matrix, we refer to a semiseparable matrix of semiseparability rank 1.

Because in this thesis, we will mainly work with semiseparable matrices of semiseparability rank 1, we will take a closer look at them. For a semiseparable matrix $S$ we have that

$$\begin{aligned}\mathrm{r}\left(\Sigma_l; A\right) &\leq 1\\ \mathrm{r}\left(\Sigma_u; A\right) &\leq 1.\end{aligned}$$

This means that the rank of all subblocks which one can take out of the lower (resp. upper) triangular part have rank at most equal to 1. One can easily verify that this class of matrices contains the diagonal matrices and also the zero matrix.

It is not necessary for a structured rank matrix to take the structure from the upper as well as from the lower triangular part of the matrix.

Definition 4. A matrix $Z$ is called an upper $\{p\}$-Hessenberg-like matrix if the $p$-lower triangular rank of $Z$ is less than or equal to $p$:

$$\mathrm{r}\left(\Sigma_l^{(p)}; Z\right) \leq p.$$

A lower $\{q\}$-Hessenberg-like matrix is defined in a similar way.

Like in the semiseparable case, when speaking about a Hessenberg-like, a $\{1\}$-Hessenberg-like matrix is meant. When it is clear from the context, we omit the notation "upper". In the literature also the class of semiseparable plus diagonal matrices is often investigated. The name already defines these matrices: a semiseparable plus diagonal is defined as the sum of a semiseparable matrix and a diagonal matrix. In the same way one can define a $\{p, q\}$-semiseparable matrix plus a diagonal or even a block diagonal, and so on.

In the next two sections we will prove that the inverse of an invertible tridiagonal matrix is an invertible semiseparable one and vice versa. We will prove even more, namely that the inverse of an invertible $\{p, q\}$-semiseparable matrix is a $\{p, q\}$-band matrix (this is a matrix with $p$ subdiagonals, and $q$-superdiagonals). Using the following definition, we will prove that the inverse of an invertible $\{p\}$-generalized Hessenberg matrix is an invertible $\{p\}$-Hessenberg-like matrix, and vice versa.

Definition 5. A matrix $H$ is defined as a $\{p\}$-generalized Hessenberg matrix if and only if all the elements below the $p$th subdiagonal are equal to zero.

In order to prove all the above mentioned properties of semiseparable and Hessenberg-like matrices, in Section 2 we will first prove a very powerful theorem, namely the nullity theorem. Based on this theorem, we will deduce properties of different classes of structured rank matrices in Section 3. After investigating simple properties connected to easy structures, stronger properties will be given, more and more related to semiseparable matrices.

## 2. The nullity theorem

In this section we will prove the nullity theorem in two different ways. Although this theorem is not so widely spread, it can easily be used to derive several interesting results about structured rank matrices and their inverses. It was formulated for the first time by Gustafson [**102**] for matrices over principal ideal domains. In [**73**], Fiedler and Markham translated this abstract formulation to matrices over a field.

Barrett and Feinsilver formulated theorems close to the nullity theorem in [**7, 9**]. Based on their observations we will provide an alternative proof of this theorem. The theorem will be followed by some small corollaries. In the following section we will apply these corollaries, to classes of matrices closely related to semiseparable matrices.

DEFINITION 6. Suppose a matrix $A \in \mathbb{R}^{m\times n}$ is given. The nullity $\mathrm{n}(A)$ is defined as the dimension of the right null space of $A$.

THEOREM 7 (The nullity theorem). *Suppose we have the following invertible matrix $A \in \mathbb{R}^{n\times n}$ partitioned as*

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}$$

*with $A_{11}$ of size $p \times q$. The inverse $B$ of $A$ is partitioned as*

$$B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

*with $B_{11}$ of size $q \times p$. Then the nullities $\mathrm{n}(A_{11})$ and $\mathrm{n}(B_{22})$ are equal.*

PROOF (From [**73**]). Suppose $\mathrm{n}(A_{11}) \leq \mathrm{n}(B_{22})$. If this is not true, we can prove the theorem for the matrices

$$\begin{pmatrix} A_{22} & A_{21} \\ A_{12} & A_{11} \end{pmatrix} \quad \begin{pmatrix} B_{22} & B_{21} \\ B_{12} & B_{11} \end{pmatrix}$$

which are also each others inverse. Suppose $\mathrm{n}(B_{22}) > 0$ otherwise $\mathrm{n}(A_{11}) = 0$ and the theorem is proved.

When $\mathrm{n}(B_{22}) = c > 0$, then there exists a matrix $F$ with $c$ linearly independent columns, such that $B_{22}F = 0$. Hence, multiplying the following equation to the right by $F$

$$A_{11}B_{12} + A_{12}B_{22} = 0,$$

we get

$$A_{11}B_{12}F = 0. \tag{1}$$

Applying the same operation to the relation:

$$A_{21}B_{12} + A_{22}B_{22} = I$$

it follows that $A_{21}B_{12}F = F$, and therefore $\operatorname{rank}(B_{12}F) \geq c$. Using this last statement together with equation (1), we derive

$$\mathrm{n}(A_{11}) \geq \operatorname{rank}(B_{12}F) \geq c = \mathrm{n}(B_{12}).$$

Together with our assumption $\mathrm{n}(A_{11}) \leq \mathrm{n}(B_{22})$, this proves the theorem. $\square$

This provides us the first proof of the theorem. The alternative proof is based on some lemmas, and makes use of determinantal formulas. Let us denote with $|\alpha|$ the cardinality of the corresponding set $\alpha$.

LEMMA 8 ([**83**, p. 13]). *Suppose $A$ is an $n \times n$ invertible matrix and $\alpha$ and $\beta$ two nonempty sets of indices in $N = \{1, 2, \ldots, n\}$, such that $|\alpha| = |\beta| < n$. Then, the determinant of any square submatrix of the inverse matrix $B = A^{-1}$ satisfies the following equation*

$$|\det B(\alpha;\beta)| = \frac{1}{|\det(A)|} |\det A(N\backslash\beta; N\backslash\alpha)|.$$

*With $N\backslash\beta$ the difference between the sets $N$ and $\beta$ is meant ($N$ minus $\beta$).*

The theorem can be seen as an extension of the standard formula for calculating the inverse of a matrix, for which each element is determined by a minor in the original matrix. This lemma already implies the nullity theorem for square subblocks and for nullities equal to 1, since this case is equivalent with the vanishing of a determinant. The following lemma shows that we can extend this argument also to the general case, i.e. every rank condition can be expressed in terms of the vanishing of certain determinants.

LEMMA 9. *Suppose $A \in \mathbb{R}^{n\times n}$ is a nonsingular matrix and $n \geq |\alpha| \geq |\beta|$. The following three statements are equivalent:*

(1) $\mathrm{n}\,(A(\alpha;\beta)) \geq d$.
(2) $\det A(\alpha';\beta') = 0$ *for all* $\alpha' \subseteq \alpha$ *and* $\beta' \subseteq \beta$ *and* $|\alpha'| = |\beta'| = |\beta| - d + 1$.
(3) $\det A(\alpha';\beta') = 0$ *for all* $\alpha \subseteq \alpha'$ *and* $\beta \subseteq \beta'$ *and* $|\alpha'| = |\beta'| = |\alpha| + d - 1$.

PROOF. The arrows (1) $\Leftrightarrow$ (2) and (1) $\Rightarrow$ (3) are straightforward. The arrow (3) $\Rightarrow$ (1) makes use of the nonsingularity of the matrix $A$. Suppose the nullity of $A(\alpha;\beta)$ to be less than $d$. This would mean that there exist $|\beta| - d + 1$ linearly independent columns in the block $A(\alpha;\beta)$. Therefore $A(\alpha;N)$ has rank less then $|\alpha|$, implying the singularity of the matrix $A$. $\square$

An alternative proof of the nullity theorem can be derived easily combining the previous two lemmas. In [**164**], Strang proves a related result and comments on different ways to prove the nullity theorem.

The following corollary is a straightforward consequence of the nullity theorem.

COROLLARY 10 (Corollary 3 in [**73**]). *Suppose $A \in \mathbb{R}^{n\times n}$ is a nonsingular matrix, and $\alpha, \beta$ to be nonempty subsets of $N$ with $|\alpha| < n$ and $|\beta| < n$. Then*

$$\operatorname{rank}\left(A^{-1}(\alpha;\beta)\right) = \operatorname{rank}\left(A(N\backslash\beta; N\backslash\alpha)\right) + |\alpha| + |\beta| - n.$$

PROOF. By permuting the rows and columns of the matrix $A$, we can always move the submatrix $A(N\backslash\beta; N\backslash\alpha)$ into the upper left part $A_{11}$. Correspondingly, the submatrix $B(\alpha;\beta)$ of the matrix $B = A^{-1}$ moves into the lower right part $B_{22}$. We have

$$\begin{aligned} \mathrm{n}\,(A_{11}) &= n - |\alpha| - \operatorname{rank}(A_{11}) \\ \mathrm{n}\,(B_{22}) &= |\beta| - \operatorname{rank}(B_{22}) \end{aligned}$$

and because $\mathrm{n}\,(A_{11}) = \mathrm{n}\,(B_{22})$, this proves the corollary. $\square$

When choosing $\alpha = N\backslash\beta$, we get

Corollary 11. *For a nonsingular matrix $A \in \mathbb{R}^{n\times n}$ and $\alpha \subseteq N$, we have:*

$$\operatorname{rank}\left(A^{-1}(\alpha; N\backslash\alpha)\right) = \operatorname{rank}\left(A(\alpha; N\backslash\alpha)\right).$$

In the next section we will use the previously obtained results about the ranks of complementary blocks of a matrix and its inverse to prove the rank properties of the inverse for some classes of structured rank matrices.

## 3. Inverse of structured rank matrices

In the papers [**3, 7, 9, 61, 71, 72, 73, 74, 77, 164**] the connection between the structured rank of a matrix and its inverse is investigated for different classes of matrices. In some papers the nullity theorem is used to obtain these connections while in others they are based on determinantal formulas. In this section, we reformulate the most important results and illustrate them by different examples.

Using Corollaries 10 and 11, one can prove the results of the following examples, by choosing the correct submatrices. Note that in this section only statements considering the lower triangular part are made. Similar results hold for the upper triangular part.

Example 1.

- Suppose the nonsingular matrix $A \in \mathbb{R}^{n\times n}$ is lower triangular. Then its inverse $A^{-1}$, will also be lower triangular. This can be seen rather easily, because for every $k \in \{1,\dots,n\}$, with $\alpha = \{1,\dots,k\}$, applying Corollary 10, we have that:

  $$\operatorname{rank}\left(A^{-1}(\alpha; N\backslash\alpha)\right) = \operatorname{rank}\left(A(\alpha; N\backslash\alpha)\right) = 0.$$

  Hence,

  $$\mathrm{r}(\Sigma_{wu}; A) = \mathrm{r}(\Sigma_{wu}; A^{-1}) = 0.$$

- Suppose we have a matrix for which the (first) subdiagonal is different from zero, and all the other entries in the strictly lower triangular part are zero. Then we know by Corollary 10 that the blocks coming from the strictly lower triangular part of the inverse of this matrix have maximum rank 1. Hence, the weakly lower triangular rank is maintained.

Before we can deduce theorems connected to other structured rank matrices, we will define some more structures. The structure

$$\Sigma_\sigma = (N \times N)\backslash\{(1,1),(2,2),\dots,(n,n)\}$$

is called the off-diagonal structure. Connected with this off-diagonal structure we have the following theorem. (When the theorems are straightforward extensions of the results presented in Section 2, the proofs are not included.)

Theorem 12 (Theorem 2.2 in [**74**]). *Assume $A$ is a nonsingular matrix. Then the off-diagonal rank of $A$ equals the off-diagonal rank of $A^{-1}$.*

$$\mathrm{r}\left(\Sigma_\sigma; A\right) = \mathrm{r}\left(\Sigma_\sigma; A^{-1}\right)$$

Example 2. Using Theorem 12, we can easily prove that the inverse of an invertible rank $k$ matrix plus a diagonal is again a rank $k$ matrix plus a diagonal.

A partition of a set $N$ is a decomposition of $N = N_1 \dot{\cup} N_2 \dot{\cup} \cdots \dot{\cup} N_p$, where $\dot{\cup}$ denotes the disjunct union: this means that $N_i \cap N_j$ is the empty set, $\forall i, j$ with $i \neq j$. A generalization of Theorem 12, from diagonal to block diagonal is as follows

THEOREM 13 (Theorem 2 in [**71**]). *Let $N = N_1 \dot{\cup} N_2 \dot{\cup} \cdots \dot{\cup} N_p$ a partition of $N$ with $N = \{1, 2, \ldots, n\}$. Let*

$$\Sigma_{\sigma b} = (N \times N) \setminus \bigcup_{i=1}^{p} (N_i \times N_i) .$$

*Then, for every nonsingular $n \times n$ matrix $A$ we have:*

$$\mathrm{r}\left(\Sigma_{\sigma b}; A^{-1}\right) = \mathrm{r}\left(\Sigma_{\sigma b}; A\right) .$$

With this theorem one can generalize the previous example.

EXAMPLE 3. The inverse of a rank $k$ matrix plus a block diagonal matrix is again a rank $k$ matrix plus a block diagonal matrix for which the sizes of the blocks of the first and the latter diagonal are the same.

The main disadvantage of both of Theorems 12 and 13 is the fact that the theorems only deal with matrices from which a diagonal, or a block diagonal matrix is subtracted. Nothing is mentioned about matrices for which the upper triangular part of rank $k$ comes from another matrix than the lower triangular part of rank $k$. This is generalized in the following theorem. Note that this theorem is already immediately in the block version, similar to Theorem 13.

THEOREM 14 (Theorem 3 in [**71**]). *Let $N = N_1 \dot{\cup} N_2 \dot{\cup} \cdots \dot{\cup} N_p$ a partition of $N$ with $N = \{1, 2, \ldots, n\}$. Let*

$$\Sigma = \bigcup_{(i,j), i > j} (N_i \times N_j) .$$

*Then, for every nonsingular $n \times n$ matrix $A$ we have:*

$$\mathrm{r}\left(\Sigma; A^{-1}\right) = \mathrm{r}\left(\Sigma; A\right) .$$

Note first of all that the entries of, e.g., $N_k$ do not necessarily need to be smaller than the ones of $N_{k+1}$. It can be seen however that the structure $\Sigma_{wl}$ fits in this theorem, we can generalize this structure towards a weakly lower triangular block structure:

$$\Sigma_{wlb} = \bigcup_{1 \leq j < i \leq n} (N_i \times N_j)$$

for which all indices in $N_k$ are smaller than all indices in $N_{k+1}$ with $1 \leq k \leq p - 1$.

EXAMPLE 4. The inverse of a matrix plus a block diagonal matrix, for which the weakly block upper triangular rank is $k$ and the weakly block lower triangular rank is $l$, is again such a matrix with the same weakly block upper and weakly block lower triangular rank. This means also that the inverse of $\{p, q\}$-semiseparable plus (block) diagonal matrix is again a $\{p, q\}$-semiseparable plus (block) diagonal matrix. Also the structure of a $\{p\}$-Hessenberg-like plus a (block) diagonal is maintained under inversion.

The theorems above already provide a lot of information connected to the class of semiseparable matrices. However, nothing is mentioned yet about the connection with the diagonal. In all the theorems the diagonal elements are excluded from the structure. We know for example that a semiseparable plus diagonal matrix has as inverse again a semiseparable plus diagonal matrix. However, the definition of the semiseparable matrices includes the diagonal, or even exceeds the diagonal. This is the point of interest in the following theorems.

Admitting the diagonal entries as well in the structure $\Sigma_{wl}$, which is the structure $\Sigma_l$, gives rise to the following theorem.

THEOREM 15. *The lower triangular ranks of $A$ and $A^{-1}$ differ at most by one:*

$$\left|\mathrm{r}(\Sigma_l; A) - \mathrm{r}\left(\Sigma_l; A^{-1}\right)\right| \leq 1$$

PROOF (From [**71**]). Choosing the correct blocks and applying Corollary 10 gives us immediately the result. We have

$$\mathrm{r}(\Sigma_l; A) = \max_{k=1,\ldots,n} \operatorname{rank} A[\{k, k+1, \ldots, n\}; \{1, \ldots, k\}].$$

Also the following relation holds:

$$\begin{aligned} &\operatorname{rank} A[\{k, k+1, \ldots, n\}; \{1, \ldots, k\}] \\ &\quad = \quad \operatorname{rank} A^{-1}[\{k+1, \ldots, n\}; \{1, \ldots, k-1\}] \end{aligned}$$

(which is just 1 on the right-hand side if $k = 1$ or $k = n$). Combining the previous relations gives us

$$\begin{aligned} &\operatorname{rank} A^{-1}[\{k, k+1, \ldots, n\}; \{1, \ldots, k\}] \\ &\quad \leq \quad \operatorname{rank} A^{-1}[\{k+1, \ldots, n\}; \{1, \ldots, k-1\}] + 2 \\ &\quad = \quad \operatorname{rank} A[\{k, \ldots, n\}; \{1, \ldots, k\}] + 1. \end{aligned}$$

Therefore the maxima also differ at most by 1. □

How this theorem fits into the theory of semiseparable matrices is shown in the following example.

EXAMPLE 5. Suppose we have the following invertible tridiagonal matrix:

$$A = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix},$$

which has lower triangular rank 2. Its inverse however is a semiseparable matrix

$$A^{-1} = \begin{pmatrix} 0 & 1 & -1 \\ 1 & -1 & 1 \\ -1 & 1 & 0 \end{pmatrix},$$

with lower triangular rank 1.

The previous example shows that for our purpose further investigation is needed to understand more clearly the connection between the lower triangular ranks of matrices and their inverses.

From now on we will investigate the connection between the class of tridiagonal matrices and the class of semiseparable matrices with respect to taking the inverse.

This means that we will restrict ourselves to the matrices having weakly upper and lower triangular rank 1. Note that in this class of matrices the tridiagonal matrices as well as the semiseparable matrices are included. Nevertheless these theorems can be generalized towards higher rank semiseparable and band matrices. For example the following behavior needs investigation:

EXAMPLE 6. Suppose we have a tridiagonal matrix $T$, with its inverse $S$:

$$T = \frac{1}{5}\begin{pmatrix} 2 & 1 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 0 & 1 & 2 & 1 \\ 0 & 0 & 1 & 2 \end{pmatrix} \qquad S = T^{-1} = \begin{pmatrix} 4 & -3 & 2 & -1 \\ -3 & 6 & -4 & 2 \\ 2 & -4 & 6 & -3 \\ -1 & 2 & -3 & 4 \end{pmatrix}.$$

It can clearly be seen that $T^{-1}$ has weakly upper and weakly lower triangular rank equal to one. However, the inverse of a matrix satisfying these conditions is not necessarily a tridiagonal matrix. Taking the matrix $A$ and its inverse $A^{-1}$

$$A = \begin{pmatrix} 1 & 2 & 2 & 2 & 2 \\ 1 & 1 & 2 & 2 & 2 \\ 1 & 1 & 1 & 2 & 2 \\ 1 & 1 & 1 & 1 & 2 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix} \quad A^{-1} = \begin{pmatrix} -1 & 0 & 0 & 0 & 2 \\ 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 1 & -1 \end{pmatrix}$$

the matrix $A$ as well as its inverse have weakly upper and weakly lower triangular rank equal to 1. However, $A^{-1}$ is not at all a tridiagonal matrix.

Investigations concerning these questions can be found in a more general framework in [**72, 77**]. Interesting to know is the fact that the framework in the latter papers has very strong connections with oscillation matrices and totally nonnegative matrices [**64, 75**]. First we will give a theorem explaining the structure of the matrices in the previous example.

THEOREM 16 (Lemma 2.5 in [**77**]). *Suppose $A$ is a nonsingular $n \times n$ matrix with weakly lower triangular rank one. For $k$ with $1 < k < n$, the following two statements are equivalent.*

- *The weakly lower triangular rank of $A$ remains one even if we extend the weakly lower triangular structure by the diagonal position $(k, k)$.*
- *In $A^{-1}$ there is a block of zeros $A^{-1}(N\backslash N_k; N_{k-1})$ with $N_k = \{1, \ldots, k\}$.*

PROOF. Straightforward using Corollary 10. □

We can immediately construct some very interesting examples connected with this theorem.

EXAMPLE 7.

- The inverse of a lower bidiagonal matrix is a lower triangular semiseparable matrix and vice versa.
- We have the following matrix $A$ and its inverse:

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \end{pmatrix}, \quad A^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 1 & -1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{pmatrix}.$$

What one can see very easily looking at the two matrices, is the fact, that in the first matrix the diagonal positions $(2, 2)$ and $(4, 4)$ can be added to the weakly lower triangular structure, without changing the rank. This means that in its inverse there will be two zero blocks which can be identified easily. It can also be seen that the first matrix has a zero block in the lower left position. Hence, its inverse has now a diagonal element in the position $(3, 3)$ which can be included in the structure without changing the rank.

- Another example showing this relation between zero submatrices and extended rank properties is the following. Take, e.g., the next matrix and its inverse:

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 \end{pmatrix}, \quad A^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{pmatrix}.$$

  It can be seen that for the extended diagonal positions $(3, 3)$ and $(4, 4)$, the corresponding zero blocks appear in the matrix.
- Let us look again at matrix $A$ from Example 6. It is important to note that the inverse of a matrix whose lower triangular rank is 1, and whose weakly upper triangular rank is 1 is not necessarily a tridiagonal matrix. The inverse matrix will only have the subdiagonal different from zero in the lower triangular part, and the weakly upper triangular rank will be 1, as the weakly upper triangular rank is maintained under inversion.

Using Theorem 16 the following theorem can be derived immediately.

THEOREM 17. *The inverse of an invertible*

- *tridiagonal matrix is a semiseparable matrix;*
- *$\{p, q\}$-band matrix is a $\{p, q\}$-semiseparable matrix;*
- *$\{p\}$-generalized Hessenberg matrix is a $\{p\}$-Hessenberg-like matrix.*

We already deduced very useful properties from the nullity theorem. Nevertheless, we can adapt the theorem a little bit and obtain immediate information about decompositions of structured rank matrices.

## 4. Generalizations of the nullity theorem

Based on the proof of the nullity theorem by Fiedler, it is very easy to generalize the nullity theorem and apply it to decompositions of structured rank matrices.

THEOREM 18. *Suppose we have an invertible matrix $A$, with an $LU$ factorization of the following form:*

$$A = LU.$$

*Suppose $A$ to be partitioned in the following form*

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}$$

*with $A_{11}$ of dimension $p \times q$. The inverse $B$ of $U$ is partitioned as*

$$U^{-1} = B = \begin{pmatrix} B_{11} & B_{12} \\ 0 & B_{22} \end{pmatrix}$$

*with $B_{11}$ of dimension $q \times p$. Then the nullities $\mathrm{n}(A_{12})$ and $\mathrm{n}(B_{12})$ are equal.*

It is enough to have the structure in terms of ranks of the matrix $U^{-1}$, because using the nullity theorem one can easily deduce the structured rank of the matrix $U$. The proof is very similar to the one of the nullity theorem.

PROOF. First, we will prove that $\mathrm{n}\,(A_{12}) \geq \mathrm{n}\,(B_{12})$, by using the relation $AU^{-1} = L$. Suppose that the nullity of $B_{12}$ equals $c$. Then there exists a matrix $F$ with $c$ linearly independent columns such that $B_{12}F = 0$. Partitioning $L$ in the following way

$$L = \begin{pmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{pmatrix}$$

with $L_{11}$ of dimension $p \times p$, we can write down the following equations:

$$\begin{aligned} A_{11}B_{12} + A_{12}B_{22} &= 0 \\ A_{12}B_{22}F &= 0 \end{aligned}$$

and

$$\begin{aligned} A_{21}B_{12} + A_{22}B_{22} &= L_{22} \\ A_{22}B_{22}F &= L_{22}F. \end{aligned}$$

Therefore $\mathrm{rank}\,(B_{22}F) \geq c$, because $L_{22}$ is of full rank. This leads us to the result:

$$\mathrm{n}\,(A_{12}) \geq \mathrm{rank}\,(B_{22}F) \geq c = \mathrm{n}\,(B_{12})\,.$$

This proves already one direction of the proof. For the other direction, $\mathrm{n}\,(A_{12}) \leq \mathrm{n}\,(B_{12})$, we use a partitioning for the inverse of $A$ and the matrix $U$, such that the upper left block of $C = A^{-1}$, denoted as $C_{11}$ has size $q \times p$ and the upper left block of $U$ denoted as $U_{11}$ has size $p \times q$. Using the equation $UA^{-1} = L^{-1}$ we can prove in a similar way as above that

$$\mathrm{n}\,(U_{12}) \geq \mathrm{n}\,(C_{12})\,.$$

Using the nullity theorem gives us

$$\mathrm{n}\,(B_{12}) = \mathrm{n}\,(U_{12}) \geq \mathrm{n}\,(C_{12}) = \mathrm{n}\,(A_{12})\,.$$

This proves the theorem. □

An analogous theorem can be formulated for the lower triangular matrix $L$. This theorem is very useful because the structured rank of both of the factors $U$ and $L$ can be determined now in terms of the structured rank of the original matrix $A$. We will give as an example here the $LU$ factorization of semiseparable matrices. We derive the structure of $L$ and $U$ in two different ways: based on the generalized nullity Theorem 18 and based on the structure of the $LU$ factorization of the inverse.

EXAMPLE 8. We will prove here that the inverse $U^{-1}$ of the matrix $U$ in the $LU$-decomposition of an invertible semiseparable matrix $S$ is an upper bidiagonal matrix. Hence, the factor $U$ is an upper triangular semiseparable matrix. (One can deduce similar properties for the lower triangular matrix $L$.) Suppose our semiseparable matrix $S$ is of size $n \times n$. Use the set of indices :

$$\begin{array}{lcl} \alpha = \{1, \ldots, k\} & \text{and} & N\backslash\alpha = \{k+1, \ldots, n\} \\ \beta = \{k, \ldots, n\} & \text{and} & N\backslash\beta = \{1, \ldots, k-1\} \end{array}$$

We have

$$\begin{aligned} \mathrm{n}\left(S(\alpha;\beta)\right) &= n-k+1-\mathrm{rank}\left(S(\alpha;\beta)\right) \\ \mathrm{n}\left(U^{-1}(N\backslash\beta; N\backslash\alpha)\right) &= n-k-\mathrm{rank}\left(U^{-1}(N\backslash\beta; N\backslash\alpha)\right). \end{aligned}$$

Rewriting these equations and using Theorem 18, we get:

$$\mathrm{rank}\left(S(\alpha;\beta)\right) = \mathrm{rank}\left(U^{-1}(N\backslash\beta; N\backslash\alpha)\right) + 1.$$

This means that for a semiseparable matrix of semiseparability rank 1 all elements above the super diagonal in the matrix $U^{-1}$ have to be zero. Therefore the inverse $U$ will be an upper triangular semiseparable matrix.

An alternative approach is to look at the inverse of the matrix $S$, namely $T$, which is tridiagonal. Let $T = U_T L_T$ be a $UL$ decomposition of the tridiagonal matrix $T$, where $L_T$ is a lower bidiagonal matrix, and $U_T$ is an upper bidiagonal matrix. This means that the $LU$ decomposition of $S$ has the following form:

$$\begin{aligned} S &= T^{-1} \\ &= L_T^{-1} U_T^{-1} \\ &= LU \end{aligned}$$

for which $L$ is lower triangular semiseparable and $U$ is upper triangular semiseparable.

Note that for the more general $LU$ factorization

$$PA = LU$$

with $P$ a nontrivial permutation matrix the two factors $L$ and $U$ are not necessarily semiseparable. Take for example the matrix

$$A = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix}.$$

Then, $PA = LU$ with

$$P = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \qquad L = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}, \qquad U = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}.$$

Hence, it is clear that $L$ is not semiseparable.

Similar theorems as Theorem 18 can be deduced for other types of decompositions. We prove a similar theorem for the $QR$-decomposition, which will give us information about the structured rank of the factor $Q$.

THEOREM 19. *Suppose we have an invertible matrix $A$, with a $QR$-factorization $A = QR$. Suppose $A$ to be partitioned in the following form*

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}$$

*with $A_{11}$ of dimension $p \times q$. The inverse of $Q$, the matrix $B$, is partitioned as*

$$Q^{-1} = B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

*with $B_{11}$ of dimension $q \times p$. Then the nullities $\mathrm{n}(A_{21})$ and $\mathrm{n}(B_{21})$ are equal.*

PROOF. Similar as the one from Theorem 18. □

Using this theorem one can deduce the structure of the lower triangular part of the orthogonal matrix $Q$. Even more, one can deduce the structure of the $Q$ factor of decompositions of for example rank $k$ matrices plus a diagonal or of $\{p, q\}$-semiseparable matrices plus a diagonal. As an example we will investigate as in the previous example the $QR$-decomposition of semiseparable matrices. A more elaborate study of the $QR$-factorization of semiseparable matrices can be found in for example [**178**]. Because of the importance of $QR$-decomposition in the remaining part of the thesis we will summarize the result in the following theorem.

THEOREM 20. *Suppose $S$ is an invertible semiseparable matrix. Suppose $S = QR$ is a $QR$-decomposition of the semiseparable matrix $S$. Then we have that $R$ has upper triangular rank 2. Moreover, $Q$ is a lower Hessenberg matrix for which the lower triangular rank is 1.*

PROOF. The structured rank properties of the orthogonal matrix $Q$ can also be derived from Theorem 19. However, here we will work with the inverse matrix $T = S^{-1}$. Denote with $T = R_T Q_T$ an $RQ$ decomposition of the matrix $T$. $R_T$ denotes an upper triangular matrix and $Q_T$ an orthogonal matrix. Because $T$ is a tridiagonal matrix, we know that $R_T$ is an upper triangular matrix with only the diagonal and the next two superdiagonals different from zero. Moreover the orthogonal matrix $Q_T$ is an upper Hessenberg matrix, for which the upper triangular rank is 1.

Translating the above equations towards $S$ we get:

$$\begin{aligned} S &= T^{-1} \\ &= Q_T^{-1} R_T^{-1} \\ &= QR. \end{aligned}$$

Because of Theorem 17 we know that $R$ has upper triangular rank 2. The matrix $Q$ is lower Hessenberg matrix for which the lower triangular rank is 1. □

Another type of decomposition is the Cholesky decomposition which is a special case of the $LU$ decomposition. When $S = R^T R$ is the Cholesky decomposition, of a positive definite semiseparable matrix $S$, then the upper triangular matrix $R$ has upper triangular rank 1.

The definition of semiseparable matrices proposed in Section 1 is justified by stating that the inverse of a tridiagonal is a semiseparable matrix. However in the literature, authors tend to use another definition where they define semiseparable matrices in terms of generators claiming that this class also contains the inverses of tridiagonal matrices. We will investigate this widespread definition, and related misunderstandings and problems.

## 5. The frequently used definition of semiseparable matrices investigated

Most of the definitions of semiseparable matrices are made by using generators, e.g., the papers [**35, 66, 129, 178**]. After giving the definition we will investigate the properties of this class of matrices.

DEFINITION 21. A matrix $S$ is called a $\{p,q\}$-generator representable semiseparable matrix if there exist two matrices $R_1$ and $R_2$ of rank $p$, $q$ respectively, such that

$$S = \operatorname{triu}(R_1, 1) + \operatorname{tril}(R_2)$$

where $\operatorname{triu}(R_1, 1)$ and $\operatorname{tril}(R_2)$ denote respectively the strictly upper triangular part of the matrix $R_1$ and the lower triangular part of the matrix $R_2$.

The operators triu and tril are chosen in correspondence with MATLAB[1]. The terminology of $\{p\}$-generator representable semiseparable matrix and generator semiseparable matrix is completely similar as in Definition 3.

Suppose $p = 1$. Because $R_1$ and $R_2$ are now two rank one matrices, they can both be written as the outer product of two vectors, respectively $u$ and $v$ for $R_1$ and $s$ and $t$ for $R_2$. These vectors are called the generators of the semiseparable matrix $S$. The generator representable semiseparable matrices defined in this way can be reconstructed by keeping $O(n)$ information. For example a symmetric generator representable semiseparable matrix looks like:

$$\left(\begin{array}{ccccc} u_1.v_1 & u_2.v_1 & u_3.v_1 & \dots & u_n.v_1 \\ u_2.v_1 & u_2.v_2 & u_3.v_2 & \dots & u_n.v_2 \\ u_3.v_1 & u_3.v_2 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & & \\ u_n.v_1 & u_n.v_2 & \dots & & u_n.v_n \end{array}\right). \tag{2}$$

In the rest of this thesis when speaking about either semiseparable or generator representable semiseparable matrices, we mean symmetric matrices of semiseparability rank 1, unless otherwise stated. We will also denote generator representable semiseparable matrices from now on with $S(u,v)$, i.e., a semiseparable matrix representable with two generators $u$ and $v$. When the matrix is not necessarily representable by generators, we denote it as $S$.

**5.1. Common misunderstandings about generator representable semiseparable matrices.** The following examples are included to illustrate some common misunderstandings about generator representable semiseparable matrices. In several papers these statements appear which are not completely true. The following

[1] MATLAB is a registered trademark of the Mathworks inc.

examples show that the inverse of a tridiagonal is not always a generator representable semiseparable matrix or a generator representable semiseparable matrix plus a diagonal, as is sometimes stated.

EXAMPLE 9. Several papers state that the inverse of a tridiagonal matrix is a generator representable semiseparable matrix. However, consider the following matrix:

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

This is clearly a nonsingular symmetric tridiagonal matrix. According to the statement above, its inverse should be a semiseparable matrix representable with two generators $u$ and $v$. Matrix $A$ is its own inverse, and one cannot represent this matrix with two generators $u$ and $v$. When we expand this class to the class of generator representable semiseparable plus diagonal matrices, we can represent the matrix $A$ in this way, but this is not the case for all the inverses of tridiagonal matrices.

For example consider the following matrix $A$, which is a block combination of the matrix from above:

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

The reader can verify that the inverse of this nonsingular symmetric tridiagonal matrix cannot be represented by two generators $u$ and $v$ nor by two generators $u, v$ and a diagonal.

Moreover the inverse of a matrix satisfying Definition 21 is not always a band matrix in general anymore, as shown in the following example:

EXAMPLE 10. The inverse of the following matrix:

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

is the matrix

$$A^{-1} = \begin{pmatrix} -1 & 0 & 2 \\ 1 & -1 & 0 \\ 0 & 1 & -1 \end{pmatrix}$$

which is clearly not tridiagonal, even though the lower triangular part comes from a rank 1 matrix, and also the strictly upper triangular part comes from a rank 1 matrix.

It was already shown in Section 3 that inclusion of the diagonal in the structure is the key to solving this problem. This means that to have a tridiagonal inverse matrix also the upper triangular part has to come from a rank 1 matrix, not only the strictly upper triangular part. This is why the inverse of the previous matrix $A$ is not tridiagonal.

**5.2. A theoretical problem, with numerical consequences.** The following example deals with the numerical instability of the representation with generators. After the example we will show that this numerical problem is a consequence of a problem in the definition.

EXAMPLE 11. Suppose a symmetric $5 \times 5$ matrix $A$ is given with the following eigenvalues: $(1, 2, 3, 100, 10^4)$. Reducing the matrix $A$ to a similar symmetric semiseparable matrix (the procedure for doing that is explained in Part 3 of the thesis) generates the following matrix $S$ (using 16 decimal digits of precision in MATLAB):

$$\begin{pmatrix} 1.6254 & -5.0812 \cdot 10^{-1} & -8.0163 \cdot 10^{-2} & -6.3532 \cdot 10^{-5} & 5.2384 \cdot 10^{-11} \\ -5.0812 \cdot 10^{-1} & 1.4259 & 2.2496 \cdot 10^{-1} & 1.7829 \cdot 10^{-4} & -1.4597 \cdot 10^{-10} \\ -8.0163 \cdot 10^{-2} & 2.2496 \cdot 10^{-1} & 2.9485 & 2.3368 \cdot 10^{-3} & -1.9152 \cdot 10^{-9} \\ -6.3532 \cdot 10^{-5} & 1.7829 \cdot 10^{-4} & 2.3368 \cdot 10^{-3} & 9.9999 \cdot 10^{1} & -8.1957 \cdot 10^{-5} \\ 5.2384 \cdot 10^{-11} & -1.4597 \cdot 10^{-10} & -1.9152 \cdot 10^{-9} & -8.1957 \cdot 10^{-5} & 1.0000 \cdot 10^{4} \end{pmatrix}.$$

Although this matrix is semiseparable it can clearly be seen that the last entry of the diagonal already approximates the largest eigenvalue. Representing this matrix with the traditional generators $u$ and $v$ gives us the following vectors for $u$ and $v$ respectively:

$$\begin{pmatrix} 3.1029 \cdot 10^{10} & -9.6998 \cdot 10^{9} & -1.5302 \cdot 10^{9} & -1.2128 \cdot 10^{6} & 1.0000 \end{pmatrix}^T$$

and

$$\begin{pmatrix} 5.2384 \cdot 10^{-11} & -1.4597 \cdot 10^{-10} & -1.9152 \cdot 10^{-9} & -8.1957 \cdot 10^{-5} & 1.0000 \cdot 10^{4} \end{pmatrix}^T.$$

Because the second element of $v$ is of the order $10^{-10}$ and is constructed by summations of elements of order 1, we can expect that this element has a precision of only 6 significant decimal digits left. Trying to reconstruct the matrix $S$ with the given generators $u$ and $v$ will therefore create large relative errors in the matrix (e.g., up to $\approx 10^{-2}$). This means that this representation looses approximately 14 decimal digits. The digit loss in this example is unacceptable.

This problem can however be explained rather easily, and it is inherent to the representation connected with the definition. Because the diagonal matrices do not belong to the class of matrices representable by the generators $u$ and $v$, the representation of the matrix above can never be very well. Very large and very small numbers can be seen in the vectors $u$ and $v$ to try to compensate the fact that the matrix is almost block diagonal.

To investigate more in detail this numerical problem, we will first look at the class of tridiagonal matrices. A first important property of symmetric tridiagonal matrices, is that they are defined by the diagonal and subdiagonal elements. In fact storing $2n - 1$ elements is enough to reconstruct the tridiagonal matrix. The following property is important in several applications. Suppose we have a symmetric tridiagonal matrix $T$ and apply the $QR$ algorithm, to compute the eigenvalues of this matrix. Doing so we get a sequence of tridiagonal matrices

$$T^{(0)} \rightarrow T^{(1)} \rightarrow \cdots \rightarrow T^{(n)} \rightarrow \cdots$$

converging towards a (block) diagonal matrix. This diagonal matrix also belongs to the class of tridiagonal matrices. To formulate this more precisely, we need the definition of pointwise convergence:

DEFINITION 22. The pointwise limit of a collection of matrices $A_\epsilon \in \mathbb{R}^{n\times n}$ (if it exists) for $\epsilon \longrightarrow \epsilon_0$, with $\epsilon, \epsilon_0 \in \mathbb{R}$ and with the matrices $A_\epsilon$ as

$$A_\epsilon = \begin{pmatrix} (a_{1,1})_\epsilon & \cdots & (a_{1,n})_\epsilon \\ \vdots & \ddots & \vdots \\ (a_{n,1})_\epsilon & \cdots & (a_{n,n})_\epsilon \end{pmatrix}$$

is defined as:

$$\lim_{\epsilon\longrightarrow\epsilon_0} A_\epsilon = \begin{pmatrix} \lim_{\epsilon\rightarrow\epsilon_0}(a_{1,1})_\epsilon & \cdots & \lim_{\epsilon\rightarrow\epsilon_0}(a_{1,n})_\epsilon \\ \vdots & \ddots & \vdots \\ \lim_{\epsilon\rightarrow\epsilon_0}(a_{n,1})_\epsilon & \cdots & \lim_{\epsilon\rightarrow\epsilon_0}(a_{n,n})_\epsilon \end{pmatrix}.$$

Let us summarize the properties of tridiagonal matrices mentioned above in a corollary.

COROLLARY 23. *For the class of symmetric tridiagonal matrices $T$ we have the following properties:*

- *They can be represented by order $O(n)$ information.*
- *The class of tridiagonal matrices is closed for pointwise convergence.*

If we look again at the class of generator representable semiseparable matrices, we can clearly see that they can be represented by $O(n)$ information. This is the main reason why authors chose this definition. Nevertheless, as we already mentioned, the set of diagonal matrices is not included in the class of generator representable semiseparable matrices, and we can easily construct a sequence of generator representable semiseparable matrices (e.g., via the $QR$-algorithm) converging to a diagonal. This means that the class of generator representable semiseparable matrices is not closed under pointwise convergence. One might correctly wonder now, what is the connection between semiseparable and generator representable semiseparable matrices with respect to the pointwise limit.

**5.3. Connection between the generator representable semiseparable and semiseparable matrices.** We will prove in this section that the class of semiseparable matrices is closed under the pointwise convergence, and even more that the closure of the generator representable semiseparable matrices is indeed the class of semiseparable matrices. First a theorem is needed. In this thesis, the submatrix of the matrix $A$ consisting of the rows $i, i+1, \ldots, j-1, j$ and the columns $k, k+1, \ldots, l-1, l$ is denoted using the MATLAB-style notation $A(i:j, k:l)$, the same notation-style is used for the elements $i, \ldots, j$ in a vector $u$: $u(i:j)$ and the $(i,j)$th element in a matrix $A$: $A(i,j)$.

The next theorem shows how the class of generator representable semiseparable matrices can be embedded in the class of semiseparable matrices.

PROPOSITION 24. *Suppose a symmetric semiseparable matrix $S$ is given, which cannot be represented by two generators. Then this matrix can be written as a block diagonal matrix, for which all the blocks are semiseparable matrices representable with two generators.*

PROOF. It can be seen that a matrix $S$ cannot be represented by two generators (e.g. $u$ and $v$), if and only if

$$\begin{array}{rcl} \exists k : 1 \leq k \leq n, \exists l : 1 \leq l \leq k & \text{such that} & S(k,l) = 0 \\ \exists i : l \leq i < k & \text{such that} & S(i,l) \neq 0 \\ \exists j : l < j \leq k & \text{such that} & S(k,j) \neq 0. \end{array}$$

$$\begin{array}{cc} & \begin{array}{ccccc} & & l & j & k \\ & & \downarrow & \downarrow & \downarrow \end{array} \\ \begin{array}{c} \\ \\ l \rightarrow \\ i \rightarrow \\ k \rightarrow \\ \\ \end{array} & \left( \begin{array}{ccccc} \ddots & & \vdots & & \vdots \\ & \ddots & \vdots & & \vdots \\ \cdots & \cdots & \ddots & & \vdots \\ & & \times & \ddots & \vdots \\ \cdots & \cdots & 0 & \times & \ddots \\ & & \vdots & & \end{array} \right) \end{array}.$$

Suppose now, that the element $S(\hat{i}, l) \neq 0$, with $l \leq \hat{i} < k$ and all $S(i,l) = 0$ for $\hat{i} < i < k$. The rank one assumption on the blocks implies that $S(i,j) = 0$, for all $\hat{i} < i \leq n$ and $1 \leq j < \hat{i} + 1$. This means that our matrix can be divided into two diagonal blocks. This procedure can be repeated until all the diagonal blocks are representable by two generators. □

NOTE. We will investigate this connection in more detail in Theorem 28 of Chapter 2, when investigating the generator representation in more detail.

For example when considering a diagonal matrix, the previous property states, that we should divide the matrix completely into 1 by 1 blocks, which are naturally representable with two generators.

We are now ready to formulate the most important proof of this section. Clearly seen in the following proof, is the situation when problems arise with the definition of semiseparable matrices in terms of the generators.

THEOREM 25. *The pointwise closure of the class of semiseparable matrices representable by two generators is the class of semiseparable matrices.*

PROOF. $\Rightarrow$ Suppose a family of semiseparable matrices representable with two generators is given:

$$S(u(\epsilon), v(\epsilon)) \in \mathbb{R}^{n \times n} \text{ for } \epsilon \in \mathbb{R} \text{ and } \epsilon \rightarrow \epsilon_0,$$

such that the pointwise limit exists:

$$\lim_{\epsilon \rightarrow \epsilon_0} S(u(\epsilon), v(\epsilon)) = S \in \mathbb{R}^{n \times n}.$$

It will be shown that this matrix belongs to the class of semiseparable matrices. It is known that $\lim_{\epsilon \rightarrow \epsilon_0} (u_i(\epsilon) v_j(\epsilon)) \in \mathbb{R}$. (Note that this does not imply that $\lim_{\epsilon \rightarrow \epsilon_0} u_i(\epsilon), \lim_{\epsilon \rightarrow \epsilon_0} v_j(\epsilon) \in \mathbb{R}$, which can lead to numerical unsound problems when representing these semiseparable matrices with two generators $u, v$.) It remains to prove that, $\forall i \in \{2, \ldots, n\}$:

$$\operatorname{rank}\left(\lim_{\epsilon \to \epsilon_0} (S(u(\epsilon), v(\epsilon))(i:n, 1:i))\right) = \operatorname{rank}(S(i:n,1:i)) \leq 1.$$

We have ($\forall i \in \{2, \ldots, n\}$):

$$\begin{aligned} \operatorname{rank}(S(i:n,1:i)) &= \max\{\operatorname{rank}(M) \,|\, M \text{ is a nonempty} \\ &\qquad \text{square submatrix of } S(i:n,1:i)\}. \end{aligned}$$

Let us take a certain nonempty $2 \times 2$ square submatrix $M$ of $S(i:n,1:i)$, denote the corresponding square submatrix of $S(u(\epsilon), v(\epsilon))$ with $M(\epsilon)$. We know that $\forall \epsilon : \operatorname{rank}(M(\epsilon)) \leq 1$, i.e. $\det(M(\epsilon)) = 0$. The determinant is continuous and therefore we have:

$$\begin{aligned} \det(M) &= \det\left(\lim_{\epsilon \to \epsilon_0} M(\epsilon)\right) \\ &= \lim_{\epsilon \to \epsilon_0} \det(M(\epsilon)) \\ &= 0. \end{aligned}$$

This means that $\operatorname{rank}(M) \leq 1$, which leads directly to

$$\operatorname{rank}(S(i:n,1:i)) \leq 1,$$

which proves one direction of the theorem.

$\Leftarrow$ Suppose a semiseparable matrix $S$ is given such that it cannot be represented by two generators. Then there exists a family $S(u(\epsilon), v(\epsilon))$ with $\epsilon \to \epsilon_0$ such that

$$\lim_{\epsilon \to \epsilon_0} S(u(\epsilon), v(\epsilon)) = S.$$

According to Proposition 24 the matrix can be written as a block diagonal matrix, consisting of 2 diagonal blocks (more diagonal blocks can be dealt with in an analogous way), which can both be represented by two generators, i.e., $S$ has the following structure:

$$S = \begin{pmatrix} S(u,v) & 0 \\ 0 & S(s,t) \end{pmatrix}.$$

In a straightforward way we can define the generators $u(\epsilon), v(\epsilon)$:

$$\begin{aligned} u(\epsilon) &= \left[\frac{u_1}{\epsilon}, \ldots, \frac{u_k}{\epsilon}, s_1, \ldots, s_l\right] \\ v(\epsilon) &= [\epsilon v_1, \ldots, \epsilon v_k, t_1, \ldots, t_n]. \end{aligned}$$

It is clearly seen that the following limit converges (it becomes even more clear when looking at the matrix (2)) :

$$\lim_{\epsilon \to 0} S(u(\epsilon), v(\epsilon)) = S.$$

This proves the theorem.

$\square$

The proof shows that the limit

$$\lim_{\epsilon \to 0} S(u(\epsilon), v(\epsilon)) = S$$

exists, but the limits of the generating vectors

$$\begin{aligned} u(\epsilon) &= \left[\frac{u_1}{\epsilon}, \ldots, \frac{u_k}{\epsilon}, s_1, \ldots, s_l\right] \\ v(\epsilon) &= [\epsilon v_1, \ldots, \epsilon v_k, t_1, \ldots, t_n] \end{aligned}$$

do not necessarily exist. In fact for $\epsilon \to 0$ some elements of $u(\epsilon)$ can become extremely large, while some elements of $v(\epsilon)$ can become extremely small. This is the behavior observed in Example 11.

We proved in this last theorem that the class of semiseparable matrices is in fact an extension of the class of generator representable semiseparable matrices. Moreover we proved that the class of semiseparable matrices is closed under pointwise convergence. Looking back at Corollary 23, one might wonder now if it is also possible to represent the class of semiseparable matrices in order $O(n)$.

## Conclusions

A thorough study of the literature, led to the observation that often, the class of semiseparable matrices and the class of generator representable semiseparable matrices are mistakenly assumed to be identical. In Section 5 we formulated some of these misunderstandings and proved an essential difference between the two classes. We observed a numerical problem, connected with the generator definition, while applying the $QR$-algorithm to this type of matrices (more about the $QR$-algorithms can be found in Part 3). This led us immediately to the nonclosedness for pointwise convergence. With the knowledge of Sections 1-3 and Section 5, it is clear that we choose the rank definition of semiseparable matrices in the sequel of the thesis.

Sections 2 and 3 are merely a résumé of the results presented by Fiedler, Markham, Barrett and Feinsilver, except for the alternative proof of the nullity theorem, which we formulated. In Section 4 we proved two generalizations of the nullity theorem applied to decompositions of structured rank matrices.

CHAPTER 2

# The representation of semiseparable matrices

In the previous chapter it was already shown that, when one wants to solve the eigenvalue problem by means of the $QR$-algorithm, the definition of semiseparable matrices with generators has some disadvantages. Therefore we proposed the more elaborate definition, in terms of the structured rank. This class of semiseparable matrices is closed under the pointwise limit. However, this new class of matrices can only be used efficiently if we have also an efficient representation as indicated by Corollary 23. This representation is the subject of this chapter.

Different types of representations will be investigated, e.g., the generator representation, the representation with a diagonal and a subdiagonal and the representation with a sequence of Givens transformations and a vector. Currently the choice for a suitable representation is the subject of investigation in several publications. Authors try to fit the class of semiseparable matrices into the larger class of recursively semiseparable matrices to obtain a stable representation, or, as we will also describe in one of the following sections, sometimes they insert parameters in order to obtain a more stable representation of the class. Some examples will be shown, revealing the possibilities and limitations of all the above mentioned representations.

In the first section we define in a mathematical way what is meant with a representation of a class of matrices. In the second and third section we study two different types of representations. The first representation is directly derived from the definition of generator representable semiseparable matrices. The second type of representation is based on papers of Fiedler [**72, 77**], where the diagonal and subdiagonal of the semiseparable matrix are taken as a representation.

In the fourth section a new type of representation is presented. For this representation we store a sequence of Givens transformations, which will denote the dependencies, between the different rows, and also a vector is stored, from which, using the dependencies we can recover all the elements within the matrix. Using this representation we are able to represent all the semiseparable matrices in a stable way. As shown in the section, this representation is a special case of a fourth more general type of representation used in the paper [**19**]. Several examples of representations, using the Givens-vector representation, of semiseparable matrices are included in Section 6. Also one example is shown, indicating that it is numerically unsound to retrieve the dependencies between the rows, based on only two elements in the same column of each of the rows. This numerical problem leads us immediately to the following section in which a numerically stable algorithm is derived to retrieve the Givens-vector representation of a semiseparable matrix.

In the last section three algorithms are presented, based on the Givens-vector representation. These algorithms show that it is possible to obtain fast algorithms when using this type of representation. The first algorithm swaps in fact the representation of the semiseparable matrix. In Section 4 the Givens transformations keep the dependencies between the rows of the matrices. Here we will change the transformations such that the Givens transformations will keep the dependencies between the columns. This will be very useful, later on, from an implementational point of view. The second algorithm will perform the multiplication of a semiseparable matrix of order $n$ and a vector in $O(n)$ operations. The semiseparable matrix does not necessarily need to be symmetric. The last algorithm will calculate the determinant of a semiseparable matrix in $O(n)$ operations. To do this we use the fact that the Givens transformations stored in the representation keep in fact enough information to calculate immediately the $QR$-factorization of the corresponding semiseparable matrix. In this way we only need to calculate the diagonal elements of the upper triangular matrix $R$.

## 1. The definition of a representation

Before investigating different possible representations, it is necessary to define what is exactly meant by a representation of a class of matrices. Although this is intuitively clear, we need a precise definition. Based on this definition we will take a close look at all the possible representations proposed.

We define a representation based on a map as follows:

Definition 26. We say that an element $v \in \mathcal{V}$ represents an element $u \in \mathcal{U}$ if there is a map $r$

$$r : \mathcal{V} \to \mathcal{U} \subseteq \mathcal{W}$$

with the following properties ($\mathcal{V}, \mathcal{W}$ are vector spaces and $\mathcal{U}$ is a set):

- $\dim(\mathcal{V}) \leq \dim(\mathcal{W})$
- $r(\mathcal{V}) = \mathcal{U}$, i.e. the map is surjective
- $\exists$ a map $s : \mathcal{U} \to \mathcal{V}$ such that $r|_{s(\mathcal{U})}$ is bijective and $r(s(u)) = u$, $\forall u \in \mathcal{U}$,

such that $r(v) = u$.

According to the definition it is possible that $\dim(\mathcal{V}) = \dim(\mathcal{W})$. It is however obvious that we want the dimension of $\mathcal{V}$ to be as small as possible, as we will use the representation $v$ instead of the element $u$ for implementing algorithms connected to the set $\mathcal{U}$. In fact the map $s : \mathcal{U} \to \mathcal{V}$ always exists, but it is included in the definition, to show the importance of a map, which will in fact, return the representation for a given matrix. Depending on the application one is working with, different types of maps will be considered, in order to obtain the most stable map. The map $r$ is called a representation map of the set $\mathcal{U}$. The element $v \in s(\mathcal{U}) \subseteq \mathcal{V}$ for which $r(v) = u$ with $u \in \mathcal{U}$ is called a representation of $u$.

To check if this definition suits our needs, we investigate the following map, when studying tridiagonal matrices. First we denote the class of symmetric tridiagonal matrices with:

$$\mathcal{T} = \{A \in \mathbb{R}^{n \times n} | A \text{ is a symmetric tridiagonal matrix}\}.$$

We have the following map:

$$
\begin{array}{rcl}
r_{\mathcal{T}} : \mathbb{R}^{n-1} \times \mathbb{R}^{n} & \rightarrow & \mathcal{T} \\
(d^{(s)}, d) & \mapsto & T = \begin{pmatrix} d_1 & d_1^{(s)} & 0 & & \\ d_1^{(s)} & d_2 & d_2^{(s)} & \ddots & \\ 0 & d_2^{(s)} & \ddots & \ddots & 0 \\ & \ddots & \ddots & & d_{n-1}^{(s)} \\ & & 0 & d_{n-1}^{(s)} & d_n \end{pmatrix}.
\end{array}
$$

It can clearly be seen that all the properties from Definition 26 are satisfied. The map $s_{\mathcal{T}}$ can be defined very easily as $s_{\mathcal{T}} = r_{\mathcal{T}}^{-1}$. This states the fact that we can use the diagonal and subdiagonal of a tridiagonal matrix to represent it.

## 2. The representation of semiseparable matrices with generators

Let us look now for a representation for symmetric $\{p, q\}$-semiseparable matrices with $p = q = 1$ as defined in Definition 3. The set of all these matrices is denoted by $\mathcal{S}$:

$$\mathcal{S} \quad = \quad \{A \in \mathbb{R}^{n \times n} | A \text{ is a symmetric semiseparable matrix}\}.$$

First of all we investigate if the map corresponding to the definition of generator representable semiseparable matrices (Definition 21) satisfies the properties of a representation map for $\mathcal{S}$. This map $r_{\mathcal{S}_1}$ is defined in the following natural way:

$$
\begin{array}{rcl}
r_{\mathcal{S}_1} : \mathbb{R}^{n} \times \mathbb{R}^{n} & \rightarrow & \mathcal{S} \\
(u, v) & \mapsto & \operatorname{tril}(uv^T) + \operatorname{triu}(vu^T, 1).
\end{array}
$$

The first condition on the map: $\dim(\mathbb{R}^n \times \mathbb{R}^n) \leq \dim(\mathbb{R}^{n \times n})$ is satisfied in a natural way, moreover the dimension of $\mathbb{R}^n \times \mathbb{R}^n$ is much smaller than the one of $\mathbb{R}^{n \times n}$. The surjectivity condition however leads to problems, e.g., the matrix

$$S_1 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

belongs to the class of semiseparable matrices $\mathcal{S}$ but $S_1 \notin r_{\mathcal{S}_1}(\mathbb{R}^n \times \mathbb{R}^n)$. Therefore this representation can never be used to represent the complete class of semiseparable matrices, as we already knew from the previous chapter, so the target set needs to be adapted. Denote

$$\mathcal{S}_{uv} = r_{\mathcal{S}_1}(\mathbb{R}^n \times \mathbb{R}^n) = \{A \in \mathcal{S} | A \text{ can be represented by } u \text{ and } v\}$$

and let us restrict the map to this subclass of semiseparable matrices. Redefine $r_{\mathcal{S}_1}$ as

$$
\begin{array}{rcl}
r_{\mathcal{S}_1} : \mathbb{R}^{n} \times \mathbb{R}^{n} & \rightarrow & \mathcal{S}_{uv} \\
(u, v) & \mapsto & \operatorname{tril}(uv^T) + \operatorname{triu}(vu^T, 1).
\end{array}
$$

This restriction makes $r_{\mathcal{S}_1}$ a representation map for the class $\mathcal{S}_{uv}$ because the surjectivity condition is clearly satisfied. We will now search for the map $s_{\mathcal{S}_1}$. The inverse of $r_{\mathcal{S}_1}$ can not be chosen as $s_{\mathcal{S}_1}$ because it is no map anymore as shown in the following example:

EXAMPLE 12. Taking the following two vectors: $u_1 = [1,2,3]^T$ and $v_1 = [2,2,2]^T$, the matrix $r_{\mathcal{S}_1}(u_1, v_1)$ is the following:

$$r_{\mathcal{S}_1}(u_1, v_1) = \begin{pmatrix} 2 & 4 & 6 \\ 4 & 4 & 6 \\ 6 & 6 & 6 \end{pmatrix}.$$

Constructing $r_{\mathcal{S}_1}(u_2, v_2)$ with $u_2 = [2,4,6]^T$ and $v_2 = [1,1,1]^T$ gives the following result:

$$r_{\mathcal{S}_1}(u_2, v_2) = \begin{pmatrix} 2 & 4 & 6 \\ 4 & 4 & 6 \\ 6 & 6 & 6 \end{pmatrix}.$$

This means that the map

$$\begin{array}{rcl} r_{\mathcal{S}_1} : r_{\mathcal{S}_1}^{-1}(\mathcal{S}_{uv}) = \mathbb{R}^n \times \mathbb{R}^n & \rightarrow & \mathcal{S}_{uv} \\ (u, v) & \mapsto & \operatorname{tril}(uv^T) + \operatorname{triu}(vu^T, 1) \end{array}$$

is not bijective.

Some kind of normalization is needed such that every matrix will have a unique set of generators. The condition $u_1 = 1$ will often lead to a unique representation, but not always.

EXAMPLE 13. Suppose the following matrix $S$ is given:

$$S = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 2 & 3 \\ 0 & 3 & 6 \end{pmatrix}.$$

The following generators both have the first element of the vector $u$ equal to 1 but are not the same: $u_1 = [1,2,3]^T, v_1 = [0,1,2]^T$ and the couple $u_2 = [1,1,3/2]^T, v_2 = [0,2,4]^T$.

Let us define a map which satisfies the conditions to be a possible choice for the map $s_{\mathcal{S}_1}$. Suppose we have a nonzero (symmetric) matrix $S \in \mathcal{S}_{uv}$, $S = [K_1, K_2, \ldots, K_n]$, where $K_i$ denotes the $i$th column of the matrix $S$. Suppose $1 \leq l \leq k \leq n$ such that $K_l$ is the first column in $S$ different from zero, and $K_k$ is the last column in $S$ different from zero. The element $S(k,l)$ clearly has to be different from zero. Otherwise either the column $K_k$ or $K_l$ has to be zero as is illustrated in the following figure:

$$\begin{array}{cc} \begin{array}{ccccc} & l & & k & \\ & \downarrow & & \downarrow & \end{array} & \\ \left(\begin{array}{ccccc} 0 & \vdots & 0 & \vdots & 0 \\ \cdots & \cdot & \cdots & S(k,l) & \cdots \\ 0 & \vdots & & \vdots & 0 \\ \cdots & S(k,l) & \cdots & \cdot & \cdots \\ 0 & \vdots & 0 & \vdots & 0 \end{array}\right) & \begin{array}{c} \\ \leftarrow \; l \\ \\ \leftarrow \; k \\ \\ \end{array} \end{array}$$

Define then $u = K_l/S(k,l)$ and $v = K_k$. Then we get a unique set $(u,v)$ for every semiseparable matrix in $\mathcal{S}_{uv}$. Note that this construction is only suitable for symmetric semiseparable matrices. This allows us to define the following map as a possible choice for the function $s_{\mathcal{S}_1}$.

DEFINITION 27. Suppose for each nonzero matrix $S \in \mathcal{S}_{uv}$ of dimension $n$, $K_l$ is the first column of $S$ different from zero and $K_k$ is the last column of $S$ different from zero. Then we define the map $s_{\mathcal{S}_1}$ in the following way:

$$\begin{array}{rcl} s_{\mathcal{S}_1} : \mathcal{S}_{uv} & \rightarrow & \mathbb{R}^n \times \mathbb{R}^n \\ S & \mapsto & \left( \dfrac{K_l}{S(k,l)}, K_k \right). \end{array}$$

Because this defines the projection into the vectors $u$ and $v$ in a unique way, we have that $r_{\mathcal{S}_1}|_{s(\mathcal{S}_{uv})}$ is bijective. We have now a unique representation for each element of the set $\mathcal{S}_{uv}$ but, as expected, we do not yet have a suitable representation for the complete class $\mathcal{S}$ of semiseparable matrices. Before investigating another type of representation, we include a small theorem, stating whether or not a semiseparable matrix can be represented with the generator representation.

THEOREM 28. *Suppose $S$ is a symmetric semiseparable matrix of size $n$. Then, $S$ is not a generator representable semiseparable matrix if and only if there exist indices $i,j$ with $1 \leq j \leq i \leq n$ such that $S(i,j) = 0$, $S(i, 1:i) \neq 0$ and $S(j:n, j) \neq 0$.*

PROOF. Suppose we have a symmetric semiseparable matrix $S$ such that the element $S(i,j) = 0$, $S(i, 1:i) \neq 0$ and $S(j:n,j) \neq 0$ with $1 \leq j \leq i \leq n$. If $S$ would be representable with two generators $u$ and $v$, this means that $S(i,j) = u_i v_j$. Hence, $u_i = 0$ or $v_j = 0$. If $u_i = 0$ this implies that

$$S(i, 1:i) = 0$$

because

$$\begin{array}{rcl} S(i,1:i) & = & u_i[v_1, \ldots, v_i] \\ & = & 0[v_1, \ldots, v_i] \\ & = & 0. \end{array}$$

If $u_i$ is different from zero this means that $v_j$ equals zero, implying

$$\begin{array}{rcl} S(j:n,j) & = & v_j[u_j, \ldots, u_n]^T \\ & = & 0[u_j, \ldots, u_n]^T \\ & = & 0. \end{array}$$

This leads to a contradiction. The other direction of the proof can be given in a similar way. □

This theorem is illustrated by the examples of Section 6.

## 3. The representation based on the diagonal and subdiagonal

The next map we will investigate is based on the diagonal and subdiagonal of semiseparable matrices. Let $d^{(s)}$ be the subdiagonal of a semiseparable matrix and $d$ its diagonal: which subclass can be represented by these two vectors? Because the rows of a semiseparable matrix are dependent up to the diagonal, the diagonal and the subdiagonal elements contain almost all information about the semiseparable matrix. More details about the information which is stored in the diagonal and subdiagonal can be found in for example the papers [**72, 77**]. In the last paragraphs of this section we will briefly mention the most important properties connected with this representation. When all the $d_i^{(s)}$ and $d_i$ are different from zero, we can construct a corresponding semiseparable matrix in the following way. Already known is the first 2 by 2 block. Note that because of symmetry we only construct the lower triangular semiseparable part. The upper left 2 by 2 block looks like

$$S = \begin{pmatrix} d_1 & \\ d_1^{(s)} & d_2 \end{pmatrix}.$$

The upper 3 by 3 block is as follows

$$S = \begin{pmatrix} d_1 & & \\ d_1^{(s)} & d_2 & \\ S(2,1) & d_2^{(s)} & d_3 \end{pmatrix}.$$

The issue now is how to compute the unknown element $S(2,1)$. By definition, all submatrices in the lower triangular part of $S$ have maximum rank 1. Hence, the following equation has to be satisfied:

$$\frac{S(2,1)}{d_1^{(s)}} = \frac{d_2^{(s)}}{d_2}. \tag{3}$$

The equation is easily solved for $S(2,1)$. Continuing this procedure gives us the following 4 by 4 matrix. Note however, that it is essential that all elements are different from zero.

$$S = \begin{pmatrix} d_1 & & & \\ d_1^{(s)} & d_2 & & \\ S(2,1) & d_2^{(s)} & d_3 & \\ S(3,1) & S(3,2) & d_3^{(s)} & d_4 \end{pmatrix}. \tag{4}$$

Using the equations:

$$\frac{S(3,1)}{S(2,1)} = \frac{S(3,2)}{d_2^{(s)}} = \frac{d_3^{(s)}}{d_3},$$

the unknown elements $S(3,1)$ and $S(3,2)$ can be found. This process can be repeated to complete the lower triangular part of the semiseparable matrix $S$. It is clear that the resulting semiseparable matrix is unique. The assumption however that all the diagonal and subdiagonal elements have to be nonzero is quite strong, and cannot be guaranteed in general. When zeros occur on the diagonal and/or the subdiagonal it is possible that there are different semiseparable matrices, having the same diagonal and subdiagonal. Therefore, we have to make choices, such that our map will point to only one semiseparable matrix. We distinguish different cases.

- $\mathbf{d_i^{(s)}} = \mathbf{0}$. Then all the elements $S(i,j)$ with $1 \leq j \leq i$ can be chosen equal to zero such that the assumptions about the rank one blocks are still satisfied. (Later on we will show that there are also other possibilities.)
- $\mathbf{d_i} = \mathbf{0}$ and $\mathbf{d_i^{(s)}} \neq \mathbf{0}$. Because of this special situation one can check that the element $d_{i-1}^{(s)}$ has to be zero, and therefore, because of the assumption above, the complete row $i$ equals zero. Because all the elements in the row $i+1$ (except for the subdiagonal and diagonal), can take various values now, and still satisfy the semiseparable structure, we assume that all the elements in this row (except the subdiagonal and the diagonal) are zero.

Using the construction explained above, the following map can be defined:

$$\begin{array}{rcl} r_{\mathcal{S}_2} : \mathbb{R}^{n-1} \times \mathbb{R}^n & \rightarrow & \mathcal{S} \\ (d^{(s)}, d) & \mapsto & S \end{array}$$

with $S$ the matrix as constructed above. We will now investigate if this map is a representation map. For the surjectivity of this map, one can expect problems, because in the preceding lines we had already to make distinctions between the different types of matrices. This can also be illustrated by the following example:

EXAMPLE 14. For each of the following two symmetric semiseparable matrices there does not exist $d^{(s)}$ and $d$ such that $r_{\mathcal{S}_2}(d^{(s)}, d)$ equals the given matrix:

$$S_1 = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix}, \qquad S_2 = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}.$$

The matrices with the same subdiagonal and diagonal which are constructed by applying $r_{\mathcal{S}_2}(d^{(s)}, d)$ are as follows:

$$S_1 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \qquad S_2 = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}.$$

Because of this nonsurjective behavior of the map, an adaptation of the target set is needed, just like in the previous case. Denote:

$$\mathcal{S}_{d,d^{(s)}} = r_{\mathcal{S}_2}(\mathbb{R}^{n-1} \times \mathbb{R}^n) = \{A \in \mathcal{S} | A \text{ can be represented by } d \text{ and } d^{(s)}\}$$

Let us now redefine the map in the following sense:

$$\begin{array}{rcl} r_{\mathcal{S}_2} : \mathbb{R}^{n-1} \times \mathbb{R}^n & \rightarrow & \mathcal{S}_{d,d^{(s)}} \\ (d^{(s)}, d) & \mapsto & S \end{array}$$

This map is a representation map for the new class, because surjectivity is now by construction satisfied, and the so-called inverse $s_{\mathcal{S}_2}$ is defined in a natural way, by projecting the diagonal and the subdiagonal of the matrix $S$. It can clearly be seen that this map satisfies the wanted properties. One important question remains uninvestigated. Is the set $\mathcal{S}_{d,d^{(s)}}$ closed for the pointwise convergence? Unfortunately this is not the case: consider for example the matrix

$$\left(\begin{array}{cccc} 1 & 1 & \epsilon & 1 \\ 1 & 1 & \epsilon & 1 \\ \epsilon & \epsilon & \epsilon & 1 \\ 1 & 1 & 1 & 1 \end{array}\right).$$

This matrix can clearly be represented by the diagonal subdiagonal representation, but the limit of this matrix:

$$\lim_{\epsilon \to 0} \left(\begin{array}{cccc} 1 & 1 & \epsilon & 1 \\ 1 & 1 & \epsilon & 1 \\ \epsilon & \epsilon & \epsilon & 1 \\ 1 & 1 & 1 & 1 \end{array}\right) = \left(\begin{array}{cccc} 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{array}\right)$$

cannot be represented by the diagonal and subdiagonal representation.

Some might find this choice of representation somewhat arbitrary. The next theorems provide however theoretical results indicating that this class of diagonal subdiagonal representable matrices has very interesting properties. The main reason for including this type of representation in the thesis, is the fact that this class has been the subject of a lot of research, just like in the case of the class of generator representable semiseparable matrices. The following theorems are based on the papers [**72, 77**]. As we are interested in the structure of the inverse, and in the structure of the $L$ and $U$ factors of the $LU$-decomposition of these matrices we provide theorems connected to these subjects.

The representation of the matrices presented below, is in fact broader than the diagonal subdiagonal approach. We will not store the diagonal and subdiagonal, but the subdiagonal and the second subdiagonal. In this way, we can represent a larger class of matrices such as semiseparable, semiseparable plus diagonal but also tridiagonal matrices. We will only derive theorems for the strictly lower triangular part, and we assume, that the subdiagonal elements are different from zero.

THEOREM 29. *Suppose $A$ is a matrix, with weakly lower triangular rank equal to* 1 *and all the subdiagonal elements different from zero. The strictly lower triangular part of $A$ is uniquely determined by the first two subdiagonals. More explicitely, if $k - i > 2$, then*

$$a_{ki} = a_{k,k-2}\, a_{k-1,k-2}^{-1}\, a_{k-1,k-3}\, a_{k-2,k-3}^{-1}\, \cdots\, a_{i+2,i}. \tag{5}$$

*Conversely, if the previous statement is true for a certain matrix $A$ with the subdiagonal elements different from zero, then this matrix has weakly lower triangular rank equal to* 1.

Important to remark is the fact when recalculating an element in fact a zigzag path is followed through the matrix as illustrated in the following example:

EXAMPLE 15. Suppose we have a $6 \times 6$ matrix who has subdiagonal rank 1. Then we have the following equation for the element:

$$\begin{array}{rcl} a_{51} & = & a_{53}\, a_{43}^{-1}\, a_{42}\, a_{32}^{-1}\, a_{31} \\ a_{62} & = & a_{64}\, a_{54}^{-1}\, a_{53}\, a_{43}^{-1}\, a_{42}. \end{array}$$

A zigzag path is followed through the matrix, starting at the lower right $\otimes$ and arriving at the upper left $\otimes$ as can be seen in Figure 2.1.

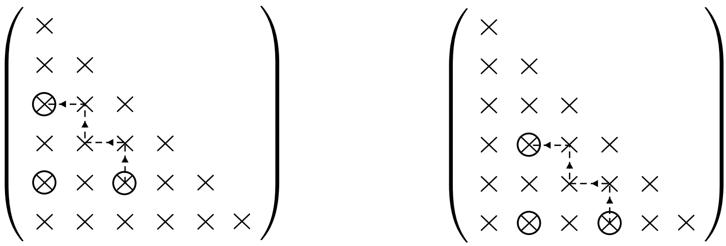

FIGURE 2.1: Calculating elements in a semiseparable part of a matrix

It is clear from the previous example that for a certain element from the second subdiagonal $a_{k+1,k-1}$ equal to zero, the complete submatrix $A(N \backslash N_k; N_{k-1})$ will be zero for $N = \{1, 2, \dots, n\}$ and $N_k = \{1, 2, \dots, k\}$.

The next theorem gives an explicit inversion formula for lower triangular matrices having weakly lower triangular rank 1. This theorem can be very useful for designing inversion algorithms for this type of matrices. Moreover when factorizing, for example a semiseparable matrix into the *LU* decomposition, both of the factors will be lower or upper triangular matrices having weakly upper or lower triangular rank 1.

THEOREM 30 (Theorem 2.7 in [**77**]). *Suppose $A$ is an $n \times n$ nonsingular matrix with weakly lower triangular rank 1 and subdiagonal elements different from zero. Then its inverse $B$ will also have weakly lower triangular rank 1 with the subdiagonal entries different from zero*

$$\begin{array}{rcll} b_{ii} & = & a_{ii}^{-1}, & i = 1, \dots, n \\ b_{i+1,i} & = & -a_{i+1,i+1}^{-1}\, a_{i+1,i}\, a_{ii}^{-1} & i = 1, \dots, n-1 \\ b_{i+1,i-1} & = & a_{i+1,i+1}^{-1} \left( a_{i+1,i}\, a_{ii}^{-1}\, a_{i,i-1} - a_{i+1,i-1} \right) a_{i-1,i-1}^{-1} & i = 2, \dots, n-2 \end{array} .$$

*The other elements can be calculated by using Theorem 29.*

It is clear that the factor $\left( a_{i+1,i}\, a_{ii}^{-1}\, a_{i,i-1} - a_{i+1,i-1} \right)$ determines the zero in the position $b_{i+1,i-1}$ depending on the rank of the block $A(i : i+1, i-1 : i)$.

THEOREM 31 (Corollary 2.2 in [**72**]). *Suppose $A$ is an $n \times n$ nonsingular matrix with weakly lower triangular rank 1 and all the subdiagonal elements different from zero. If $A = LU$ is the LU-decomposition of $A$. The $L$ is also a matrix with weakly lower triangular rank 1 and the subdiagonal elements different from zero.*

Theorem 31 can easily be generalized to the weakly upper triangular rank 1 case, such that also the $U$ factor has weakly upper triangular rank 1 and nonzero sup-diagonal entries.

One can clearly see that a combination of Theorems 31 and 30 already presents a method for inverting a large class of matrices, for example semiseparable, semiseparable plus diagonal and tridiagonal matrices.

Even though useful decompositions and theorems exist for the above class, it does not yet satisfy our needs. Still not all semiseparable matrices can be represented with the diagonal subdiagonal representation.

## 4. A new representation for symmetric semiseparable matrices

We consider a new type of representation. For a symmetric semiseparable matrix of dimension $n$, this representation consists of $n-1$ Givens transformations and a vector of length $n$. The Givens transformations are denoted as $G = [G_1, \ldots, G_{n-1}]$ and the vector as $d = [d_1, \ldots, d_n]$. It is clear that this representation needs $2n-1$ parameters to reconstruct the complete semiseparable matrix, as a Givens transformation can be stored in one parameter as described in [**91**, Section 5.1.11]. Here however, we chose to store the cosine and the sine of the Givens transformation, as shown later on.

The following figures denote how the semiseparable matrix can be reconstructed, using this information. How to retrieve the representation given a matrix, is the subject of Section 7. The elements denoted by $\boxtimes$ make up the semiseparable part of the matrix. Initially one starts on the first 2 rows of the matrix. The element $d_1$ is placed in the upper left position, then a Givens transformation is applied, and finally to complete the first step, element $d_2$ is added in position $(2,1)$. Only the first two columns and rows are shown here.

$$\left(\begin{array}{cc} d_1 & 0 \\ 0 & 0 \end{array}\right) \rightarrow G_1 \left(\begin{array}{cc} d_1 & 0 \\ 0 & 0 \end{array}\right) + \left(\begin{array}{cc} 0 & 0 \\ 0 & d_2 \end{array}\right) \rightarrow \left(\begin{array}{cc} \boxtimes & 0 \\ \boxtimes & d_2 \end{array}\right).$$

The second step consists of applying the Givens transformation $G_2$ on the second and the third row, furthermore $d_3$ is added in position $(3,3)$. Here only the first three columns are shown and the second and third row. This leads to:

$$\left(\begin{array}{ccc} \boxtimes & d_2 & 0 \\ 0 & 0 & 0 \end{array}\right) \rightarrow G_2 \left(\begin{array}{ccc} \boxtimes & d_2 & 0 \\ 0 & 0 & 0 \end{array}\right) + \left(\begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & d_3 \end{array}\right) \rightarrow \left(\begin{array}{ccc} \boxtimes & \boxtimes & 0 \\ \boxtimes & \boxtimes & d_3 \end{array}\right).$$

This process can be repeated by applying the Givens transformation $G_3$ on the third and the fourth row of the matrix, and afterwards adding the diagonal element $d_4$. After applying all the Givens transformations and adding all the diagonal elements, the lower triangular part of a symmetric semiseparable matrix is constructed. Because of the symmetry also the upper triangular part is known.

Suppose the Givens-vector representation of a semiseparable matrix $S$ is known. When denoting the Givens transformations as:

$$G_l = \left(\begin{array}{cc} c_l & -s_l \\ s_l & c_l \end{array}\right),$$

the elements $S(i,j)$ with $n > i \geq j$ are calculated in the following way: $S(i,j) = c_i s_{i-1} s_{i-2} \cdots s_j d_j$. When $n = i$ we have $S(i,j) = s_{n-1} s_{n-2} \cdots s_j d_j$. When $n \geq j > i$, $S(i,j)$ can be calculated in a similar way, because of the symmetry. The elements of the semiseparable matrix can therefore be calculated in a stable way based on the Givens-vector representation. This means that we have constructed the following map $r_{\mathcal{S}_3}$

$$\begin{array}{rcl} r_{\mathcal{S}_3} : \mathbb{R}^{2\times(n-1)} \times \mathbb{R}^n & \rightarrow & \mathbb{R}^{n\times n} \\ \left[\left(\begin{array}{ccc} c_1 & \ldots & c_{n-1} \\ s_1 & \ldots & s_{n-1} \end{array}\right), (d_1, \ldots, d_n)\right] & \mapsto & \left(\begin{array}{cccc} c_1 d_1 & & & \\ c_2 s_1 d_1 & c_2 d_2 & & \\ c_3 s_2 s_1 d_1 & c_3 s_2 d_2 & c_3 d_3 & \\ \vdots & \vdots & & \ddots \end{array}\right). \end{array}$$

The storage costs $3n-2$, which is of the same order as the generator representation. We store the cosine and sine separately because of numerical accuracy. Theoretically, only storing the cosine (or sine) would be enough leading to a storage cost of $2n-1$.

Example 16 (Example 11 continued). The Givens-vector representation for the matrix of Example 11 is the following: (In the first row of $G$ the elements $c_1, \ldots, c_4$ are stored and in the second row the elements $s_1, \ldots, s_4$.)

$$G = \left(\begin{array}{rrrr} 0.9534 & 0.9878 & 1.0000 & 1.0000 \\ -0.3017 & 0.1558 & 0.0008 & 0.0000 \end{array}\right)$$

and

$$d = \left(\begin{array}{ccccc} 1.7049 & 1.4435 & 2.9485 & 9.9999 \cdot 10 & 1.0000 \cdot 10^4 \end{array}\right).$$

All the elements of the semiseparable matrix can be reconstructed now with high relative precision if the corresponding elements of $G$ and $d$ are known with high relative precision. In this example the maximum absolute error between the original semiseparable matrix and the semiseparable matrix represented with the Givens-vector representation is of the order $10^{-14}$. For the generator representation we obtained elements which had only 2 significant digits left.

In this section the construction of a symmetric semiseparable matrix given a sequence of Givens transformations and a vector was presented. Before developing a method for computing the representation given a symmetric semiseparable matrix, we will give another type of representation, recently presented in [**19**] followed by a section with some illustrating examples.

## 5. Other types of representations

In this subsection we will investigate a special class of matrices, which, as we will prove, is the same class of matrices as the semiseparable matrices $\mathcal{S}$. Let us define the matrices in this class:

Definition 32. Suppose we have a class of matrices $\mathcal{S}'$ such that $S \in \mathcal{S}'$ has the following structure:

$$S(i,j) = \begin{cases} u_i t_{i-1} t_{i-2} \ldots t_j v_j & 1 \leq j < i \leq n \\ u_i v_i & 1 \leq i = j \leq n \\ u_j t_{j-1} t_{j-2} \ldots t_i v_i & 1 \leq i < j \leq n \end{cases}$$

with $u_i, v_i \in \mathbb{R}$ for $1 \leq i \leq n$ and $t_i \in \mathbb{R}$ for $1 \leq i \leq n-1$.

This class of matrices is a special case of sequentially semiseparable matrices as defined in [**34**] and of quasiseparable matrices as defined in [**56**]. See also [**48, 49, 58**]. Note however that in the references mentioned above the diagonal is not incorporated into the structure while this is the case in Definition 32.

First we will prove that this class of matrices is equal to the class of semiseparable matrices.

THEOREM 33. *The class of matrices $\mathcal{S}'$ satisfying Definition 32 and the class of semiseparable matrices $\mathcal{S}$ are equal.*

PROOF. The proof is divided in two parts.

$\mathcal{S}' \subset \mathcal{S}$: Let $S \in \mathcal{S}'$. We have to prove that for all $1 \leq i \leq n$

$$\operatorname{rank}\left(S(i:n, 1:i)\right) \leq 1.$$

This corresponds to the demand that the determinant of every $2 \times 2$ submatrix of the lower triangular part of $S$ equals 0. Let us consider the following $2 \times 2$ submatrix (with $i \geq k, l > i$ and $k > j$.)

$$\begin{aligned} &\left(\begin{array}{cc} S(i,j) & S(i,k) \\ S(l,j) & S(l,k) \end{array}\right) \\ &= \left(\begin{array}{ll} u_i t_{i-1} \ldots t_k \ldots t_j v_j & u_i t_{i-1} \ldots t_k v_k \\ u_l t_{l-1} \ldots t_{i-1} \ldots t_k \ldots t_j v_j & u_l t_{l-1} \ldots t_{i-1} \ldots t_k v_k \end{array}\right). \end{aligned}$$

This matrix can be written as the product of a diagonal and a rank 1 matrix:

$$\left(\begin{array}{cc} u_i t_{i-1} \ldots t_k & 0 \\ 0 & u_l t_{l-1} \ldots t_k \end{array}\right)\left(\begin{array}{cc} t_{k+1} \ldots t_j v_j & v_k \\ t_{k+1} \ldots t_j v_j & v_k \end{array}\right).$$

Therefore, we have that the determinant of every $2 \times 2$ matrix of the lower triangular part of $S$ equals zero and $S \in \mathcal{S}$ which proves the first part of the theorem.

$\mathcal{S} \subset \mathcal{S}'$: Let $S \in \mathcal{S}$, then we know that $S$ is a block diagonal matrix for which all the blocks are generator representable semiseparable matrices. Without loss of generality we assume there are only 2 blocks on the diagonal. Let us denote the generators of the first block with $\tilde{u}$ and $\tilde{v}$, both of length $n_1$, and denote the generators of the second block with $\hat{u}$ and $\hat{v}$ both of length $n_2$. If we define now the following vectors:

$$u = \left(\begin{array}{c} \tilde{u} \\ \hat{u} \end{array}\right) \quad v = \left(\begin{array}{c} \tilde{v} \\ \hat{v} \end{array}\right)$$

and $t_1 = t_2 = \ldots = t_{n_1-1} = 1$, $t_{n_1} = 0$ and $t_{n_1+1} = t_{n_1+n_2-1} = 1$, then we have that the matrix constructed from Definition 32 with $u, v$ and $t$ is equal to the matrix $S$. This means that the matrix $S \in \mathcal{S}'$, which proves the other part of the theorem. $\square$

The theorem above proves in fact that the map $r_{\mathcal{S}'}$:

$$\begin{aligned} r_{\mathcal{S}'} : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^{n-1} &\rightarrow \mathcal{S} \\ (u, v, t) &\mapsto S \end{aligned}$$

for which $S$ is defined as in Definition 32 is surjective.

The construction of the map $s_{\mathcal{S}'}$ is however not so straightforward. The elements $t$ are not defined uniquely and one should choose the elements $t$ in such a way that the representation is stable, this choice is however not straightforward, see for example [**19**]. In this reference the lower triangular part of a matrix is represented using Definition 32. Moreover, the representation of the semiseparable matrix as presented here needs $3n-1$ parameters.

The representations presented here are not the only ones. More types of representations can be found in Chapter 3, Section 5. For example the paper [**127**] provides a representation for nonsymmetric semiseparable matrices which requires $3n$ storage capacity.

## 6. Some examples

The reader can try to decide whether the following matrices are representable with the generator representation, the diagonal subdiagonal representation or the Givens-vector representation. With diag$(v)$ we denote the diagonal matrix with as diagonal elements the elements coming from the vector $v$.

$$D_1 = \operatorname{diag}([1,1,1,1,1]) \tag{6}$$

$$D_2 = \operatorname{diag}([0,1,1,1,1]) \tag{7}$$

$$S_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \end{pmatrix} \tag{8}$$

$$S_2 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{9}$$

$$S_3 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{10}$$

$$S_4 = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix} \tag{11}$$

$$S_5 = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \tag{12}$$

The results can be found in the next table.

| matrix | singular? | representation $u$ and $v$ | representation $d$ and $d^{(s)}$ | representation $d$ and $G$ |
|---|---|---|---|---|
| matrix 6 | no | no | yes | yes |
| matrix 7 | yes | no | yes | yes |
| matrix 8 | no | no | yes | yes |
| matrix 9 | yes | no | yes | yes |
| matrix 10 | no | no | yes | yes |
| matrix 11 | no | yes | no | yes |
| matrix 12 | yes | yes | no | yes |

One can clearly see that the Givens-vector representation is the most general one. Moreover, this representation is still an $O(n)$ representation.

In the following examples the Givens-vector representation of several different semiseparable matrices is given. The construction of this representation is the subject of the next section.

$$\left(\begin{array}{ccc} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{array}\right) \leftrightarrow \begin{array}{rcl} G & = & \left(\begin{array}{cc} \sqrt{3}/3 & \sqrt{2}/2 \\ \sqrt{6}/3 & \sqrt{2}/2 \end{array}\right) \\ d & = & \left(\sqrt{3}, \sqrt{2}, 1\right) \end{array}$$

$$\left(\begin{array}{ccc} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{array}\right) \leftrightarrow \begin{array}{rcl} G & = & \left(\begin{array}{cc} 1 & 1 \\ 0 & 0 \end{array}\right) \\ d & = & (1, 2, 3) \end{array}$$

$$\left(\begin{array}{ccc} 1 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{array}\right) \leftrightarrow \begin{array}{rcl} G & = & \left(\begin{array}{cc} \sqrt{2}/2 & 0 \\ \sqrt{2}/2 & 1 \end{array}\right) \\ d & = & \left(\sqrt{2}, 1, 1\right) \end{array}$$

$$\left(\begin{array}{ccc} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \end{array}\right) \leftrightarrow \begin{array}{rcl} G & = & \left(\begin{array}{cc} \sqrt{3}/3 & \sqrt{2}/2 \\ \sqrt{6}/3 & \sqrt{2}/2 \end{array}\right) \\ d & = & \left(\sqrt{3}, 0, \sqrt{2}\right) \end{array}$$

$$\left(\begin{array}{ccc} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{array}\right) \leftrightarrow \begin{array}{rcl} G & = & \left(\begin{array}{cc} \sqrt{2}/2 & 0 \\ \sqrt{2}/2 & 1 \end{array}\right) \\ d & = & \left(\sqrt{2}, 0, 1\right) \end{array}$$

$$\left(\begin{array}{ccc} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{array}\right) \leftrightarrow \begin{array}{rcl} G & = & \left(\begin{array}{cc} 0 & 1 \\ 1 & 0 \end{array}\right) \\ d & = & (1, 0, 1) \end{array}$$

$$\left(\begin{array}{ccccc} 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \end{array}\right) \leftrightarrow \begin{array}{rcl} G & = & \left(\begin{array}{cccc} 1/2 & \sqrt{3}/3 & 0 & \sqrt{2}/2 \\ \sqrt{3}/2 & \sqrt{6}/3 & 1 & \sqrt{2}/2 \end{array}\right) \\ d & = & \left(2, 0, \sqrt{2}, \sqrt{2}, 1\right) \end{array}$$

In the following example, we construct the Givens-vector representation of a semiseparable matrix based on the diagonal and subdiagonal elements. In fact we already know that this naive procedure cannot work in all cases, because the diagonal subdiagonal representable matrices do not cover the complete set of semiseparable matrices. The following example shows that the procedure, just explained to determine the representation is not stable.

EXAMPLE 17. Suppose we have a given semiseparable matrix,

$$S = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

and we add random noise of the size $10^{-16}$ to this matrix. We will then construct the Givens-vector representation of this matrix, based on the diagonal and subdiagonal elements, and build up again the semiseparable matrix given the Givens-vector representation. We see that we get large errors even though there was only a small change of the size $10^{-16}$. The disturbed matrix $A$ has the following form:

$$\begin{pmatrix} 1.0000 & 1.0000 & 4.0570 \cdot 10^{-16} \\ 1.0000 & 7.3820 \cdot 10^{-16} & 9.3546 \cdot 10^{-16} \\ 7.9193 \cdot 10^{-16} & 1.7626 \cdot 10^{-16} & 1.0000 \end{pmatrix}.$$

Calculating the Givens-vector representation in the way presented above, and then recalculating the semiseparable matrix gives the following result:

$$\begin{pmatrix} 1.0000 & 1.0000 & 2.3877 \cdot 10^{-1} \\ 1.0000 & 7.3820 \cdot 10^{-16} & 1.7626 \cdot 10^{-16} \\ 2.3877 \cdot 10^{-1} & 1.7626 \cdot 10^{-16} & 1.0000 \end{pmatrix}.$$

This matrix has the element in position $(3, 1)$ recovered in a very inaccurate way. In the next section a more robust and stable algorithm will be presented, and we will reconsider this example.

## 7. Retrieving the representation from a semiseparable matrix

Here, a method is proposed to retrieve the Givens-vector representation of a semiseparable matrix in a stable way.

In fact we search for the map:

$$\begin{array}{rcl} s_{\mathcal{S}_2} : \mathcal{S} & \rightarrow & \mathbb{R}^{2\times(n-1)} \times \mathbb{R}^n \\ S & \mapsto & (G, d). \end{array}$$

Suppose we have a semiseparable matrix as in (6). The vector elements $d_i$ can be retrieved rather easily from the matrix. In fact:

$$\begin{array}{l} \|S(i:n,i)\|_2 \\ \quad = \sqrt{(c_i d_i)^2 + (c_{i+1} s_i d_i)^2 + \cdots + (c_{n-1} s_{n-2} \ldots s_i d_i)^2 + (s_{n-1} s_{n-2} \ldots s_i d_i)^2} \\ \quad = \sqrt{(c_i d_i)^2 + (c_{i+1} s_i d_i)^2 + \cdots + (s_{n-1}^2 + c_{n-1}^2)(s_{n-2} \ldots s_i d_i)^2} \\ \quad = \sqrt{(c_i^2 + s_i^2) d_i^2} \\ \quad = |d_i|. \end{array}$$

This means that the absolute value of $d_i$ can be calculated by calculating the norm $\|S(i:n,i)\|_2$. All diagonal elements are determined now, except their signs, this is done at the same time as the calculation of the Givens transformations. To calculate the corresponding Givens transformations connected with the semiseparable matrix, we first map the matrix $S$ towards another semiseparable matrix. This procedure is quite expensive, but results in a stable way to compute the Givens-vector representation. The matrix $S$ is mapped onto the following matrix of norms. Note that the choice of the norm does not play a role:

$$\hat{S} = \begin{pmatrix} \|S(1,1)\| & & \\ \|S(2,1)\| & \|S(2,1:2)\| & \\ \vdots & & \ddots \\ \|S(n,1)\| & \|S(n,1:2)\| & \ldots \|S(n,1:n)\| \end{pmatrix}. \tag{13}$$

So in fact a new matrix $\hat{S}$ is created with elements $\hat{S}_{i,j} = \|S(i,1:j)\|$. Note that this matrix has the same dependencies between the rows as the matrix $S$ (except for the signs). We start calculating the last Givens transformation $G_{n-1}$ such that

$$G_{n-1}\left(r_{n-1}\ ,\ 0\right)^T = \left(\hat{S}_{n-1,n-1}\ ,\ \hat{S}_{n,n-1}\right)^T.$$

Before calculating the next Givens transformation we have to update the matrix $\hat{S}$, by applying the Givens transformation $G_{n-1}$ to the rows $n-1$ and $n$. Denoting this new matrix as $\hat{S}^{(n-1)}$, the next Givens transformation $G_{n-2}$ is calculated such that

$$G_{n-2}\left(r_{n-2}\ ,\ 0\right)^T = \left(\hat{S}_{n-2,n-2}\ ,\ \hat{S}^{(n-1)}_{n-1,n-2}\right)^T = \left(\hat{S}^{(n-1)}_{n-2,n-2}\ ,\ \hat{S}^{(n-1)}_{n-1,n-2}\right)^T.$$

Updating again the matrix $\hat{S}^{(n-1)}$ by applying the Givens transformation $G^T_{n-2}$ to the rows $n-2$ and $n-1$ we get the matrix $\hat{S}^{(n-2)}$ and we can calculate $G_{n-3}$. Consecutively, all the Givens transformations can be calculated, satisfying:

$$G_i\left(r_i\ ,\ 0\right)^T = \left(\hat{S}_{i,i}\ ,\ \hat{S}^{(i+1)}_{i+1,i}\right)^T = \left(\hat{S}^{(i+1)}_{i,i}\ ,\ \hat{S}^{(i+1)}_{i+1,i}\right)^T.$$

This procedure gives us the Givens-vector representation, except for the signs of the diagonal elements, but the sign is determined rather easily, by looking at the signs of the elements in the original matrix. The Givens transformations are uniquely determined, because we take $c_i$ always positive, and when a Givens transformation of the following form has to be determined: $G\left(0\ ,\ 0\right)^T = (0\ ,\ 0)$ we take $G$ equal to the identity matrix. We illustrate the numerical stability with respect to the algorithm used in Example 17:

Example 18 (Example 17 continued). The same experiment is performed as in Example 17, but now with the newly designed algorithm: the output matrix is the following one:

$$\begin{pmatrix} 1.0000 & 1.0000 & 8.1131 \cdot 10^{-16} \\ 1.0000 & 7.5895 \cdot 10^{-16} & 6.1575 \cdot 10^{-31} \\ 8.1131 \cdot 10^{-16} & 6.1575 \cdot 10^{-31} & 1.0000 \end{pmatrix}$$

This matrix is much closer to the original one compared to the result in Example 17.

This construction of the Givens-vector representation is of course quite expensive (faster, even $O(n)$ algorithms, can be constructed, for arbitrary semiseparable matrices, but they are numerically unstable in general). In practical applications however, faster ($O(n)$) and numerically stable methods can be designed to retrieve the representation, e.g., if the given semiseparable matrix is represented with two generators $u$ and $v$, one can derive the Givens-vector representation in $O(n)$ computational time. These methods however, are highly dependent of the application or problem one is trying to solve. Therefore we do not further investigate algorithms to retrieve the Givens-vector representation.

In the reduction algorithm from symmetric to semiseparable, the output of the algorithm is already in the appropriate form (see Chapter 4 and Chapter 6). Also the $QR$-algorithm which can be found in Chapter 7 uses as input this representation and gives as output the Givens-vector representation of the new semiseparable matrix. In both of these algorithms there is no need to apply this expensive procedure for calculating the Givens-vector representation from the elements of the semiseparable matrix.

Moreover, this representation reveals already the $QR$-factorization of the semiseparable matrix $S$: the Givens transformations appearing in the representation of the matrix are exactly the same as the Givens transformations appearing in the $Q$ factor of the $QR$-factorization. (More information can be found in Chapter 7 and Chapter 9 .)

## 8. Some algorithms connected to the representation

**8.1. Swapping the representation.** Taking a closer look at the construction of a semiseparable matrix using the Givens-vector representation, we see that we keep in fact the dependency between the rows and construct the semiseparable matrix from top to bottom. In fact we can do completely the same by keeping the dependency between the columns and starting at the right side, and building the matrix from the right towards the left.

These two possibilities are in fact very important, because as we will show later on in the implicit $QR$-algorithm, the choice between these two can make the implementation much more simple and cheap in terms of operation cost. Therefore we present in this section an order $O(n)$ algorithm to swap the representation from top to bottom into a representation from right to left.

We will show this construction by an example of order $n = 4$. We have in fact the following matrix:

$$\begin{pmatrix} c_1 d_1 & & & \\ c_2 s_1 d_1 & c_2 d_2 & & \\ c_3 s_2 s_1 d_1 & c_3 s_2 d_2 & c_3 d_3 & \\ s_3 s_2 s_1 d_1 & s_3 s_2 d_2 & s_3 d_3 & d_4 \end{pmatrix}.$$

A naive way to do it, is to calculate the diagonal and subdiagonal elements, and to reconstruct the representation in the other direction by using this information. But as already mentioned before not all semiseparable matrices can be represented with the diagonal subdiagonal approach. We will start reconstructing the representation from right to left, this means that for $\hat{G}$ and $\hat{d}$, we first calculate $\hat{d}_4$ and $\hat{G}_3$. One can calculate $\hat{d}_4$ immediately as $\hat{d}_4 = c_1 d_1$. The dependency between the first and second column has to be calculated from $[s_1 d_1, d_2]$, because it is possible that $c_2$ equals zero. Let us use now the following notation: $[s_1 d_1, d_2]\hat{G}_3^T = [0, r_2]$, where $\hat{G}_3$ is the Givens transformation annihilating $s_1 d_1$ and $r_2$ is the norm of $[s_1 d_1, d_2]$. Now we calculate $\hat{G}_2$ as the Givens transformation such that $[s_2 r_2, d_3]\hat{G}_2^T = [0, r_3]$, and one has to continue now with $[s_3 r_3, d_4]\hat{G}_1^T = [0, r_4]$. The reader can verify, that with this construction we omit all the problems which can occur with zero structures. The new Givens transformations are determined by the Givens transformations $\hat{G}_i$ and the new diagonal elements are determined by the $r_i$.

From an implementational point of view this will be a very powerful tool. If one performs for example a row transformation on a semiseparable matrix presented with the Givens-vector representation from top to bottom, one can see that the Givens transformations do not contain the correct dependencies between the rows anymore, because the transformation changed them. If however the matrix was presented with the Givens-vector representation from right to left, performing a row transformation on the matrix, would not change the dependencies between the columns, these dependencies are stored in the Givens transformations and are therefore not altered by this transformation.

**8.2. A fast matrix vector multiplication.** In this section an example of an implementation with the Givens-vector representation is given. In later chapters, other algorithms based on this representation can be found.

Because for generator representable semiseparable matrices the multiplication with a vector clearly can be performed with $O(n)$ operations, we also want an $O(n)$ implementation for the Givens-vector representation. The formulas will be given for nonsymmetric semiseparable matrices, that are represented by two sequences of Givens transformations and two vectors. Let us denote the first sequence as:

$$\begin{aligned} G &= \begin{pmatrix} c_1 & c_2 & \dots & c_{n-1} \\ s_1 & s_2 & \dots & s_{n-1} \end{pmatrix} \\ d &= \begin{pmatrix} d_1 & d_2 & \dots & d_n \end{pmatrix} \end{aligned}$$

and the second as:

$$\begin{aligned} H &= \begin{pmatrix} r_1 & r_2 & \dots & r_{n-2} \\ t_1 & t_2 & \dots & t_{n-2} \end{pmatrix} \\ e &= \begin{pmatrix} e_1 & e_2 & \dots & e_{n-1} \end{pmatrix}. \end{aligned}$$

We want to calculate the multiplication between $S$ and $v$ where

$$S = \begin{pmatrix} c_1 d_1 & r_1 e_1 & r_2 t_1 e_1 & \dots & r_{n-2} t_{n-3} \cdots t_1 e_1 & t_{n-2} t_{n-3} \cdots t_1 e_1 \\ c_2 s_1 d_1 & c_2 d_2 & r_2 e_2 & & \vdots & \vdots \\ c_3 s_2 s_1 d_1 & c_3 s_2 d_2 & c_3 d_3 & \ddots & & \\ \vdots & & \ddots & \ddots & r_{n-2} e_{n-2} & t_{n-2} e_{n-2} \\ c_{n-1} s_{n-2} \cdots s_1 d_1 & \dots & & & c_{n-1} d_{n-1} & e_{n-1} \\ s_{n-1} s_{n-2} \cdots s_1 d_1 & \dots & & & s_{n-1} d_{n-1} & d_n \end{pmatrix}$$

and

$$v = \begin{pmatrix} v_1 & v_2 & \dots & v_n \end{pmatrix}^T.$$

To deduce the algorithm, we have to decompose the matrix $S$ into a strictly upper triangular, and a lower triangular part. Denote $S_1 = \mathrm{tril}(S)$ and $S_2 = \mathrm{triu}(S, 1)$. We will now compute:

$$\begin{aligned} y &= S_1 v \\ z &= S_2 v \end{aligned}$$

such that $x = y + z = Sv$ is the result. To calculate $y = \begin{pmatrix} y_1 & y_2 & \ldots & y_n \end{pmatrix}$ in a fast way, we rewrite the following formulas (only the first 4 components of $y$ are denoted):

$$\begin{aligned} y_1 &= c_1 d_1 v_1 \\ y_2 &= c_2 s_1 d_1 v_1 + c_2 d_2 v_2 \\ y_3 &= c_3 s_2 s_1 d_1 v_1 + c_3 s_2 d_2 v_2 + c_3 d_3 v_3 \\ y_4 &= c_4 s_3 s_2 s_1 d_1 v_1 + c_4 s_3 s_2 d_2 v_2 + c_4 s_3 d_3 v_3 + c_4 d_4 v_4. \end{aligned}$$

We use some temporary variables called $a_i$. Rewriting the formulas reveals the order $O(n)$ algorithm for the multiplication.

$$\begin{aligned} y_1 &= c_1 (d_1 v_1) \\ &= c_1 a_1 \\ y_2 &= c_2 (s_1 d_1 v_1 + d_2 v_2) \\ &= c_2 (s_1 a_1 + d_2 v_2) \\ &= c_2 a_2 \\ y_3 &= c_3 (s_2 (s_1 d_1 v_1 + d_2 v_2) + d_3 v_3) \\ &= c_3 (s_2 a_2 + d_3 v_3) \\ &= c_3 a_3 \\ y_4 &= c_4 (s_3 (s_2 (s_1 d_1 v_1 + d_2 v_2) + d_3 v_3) v_3 + d_4 v_4) \\ &= c_4 (s_3 a_3 + d_4 v_4) \\ &= c_4 a_4. \end{aligned}$$

Combining the last 2 equalities of all the $y_i$'s one can derive an order $n$ algorithm to perform the multiplication of $S_1$ and $v$. The multiplication of $S_2$ and $v$ can be derived in a completely analogous way.

**8.3. The determinant of a semiseparable matrix in order $O(n)$ flops.** In this section, we will design an order $O(n)$ algorithm for calculating the determinant of a semiseparable matrix represented with the Givens-vector representation.

We will use the fact that the Givens transformations for representing the matrix in fact contain all the information needed for the $QR$-factorization of the corresponding matrix. Using this information, we can very easily calculate the diagonal elements of the $R$ factor of the semiseparable matrix. Multiplying these diagonal elements will give us the wanted determinant of the semiseparable matrix.

We have, because of the special structure of the representation, the Givens transformations $G_1, \ldots, G_{n-1}$ such that applying $G_{n-1}^T$ on the last two rows will annihilate all the elements except for the diagonal element, applying $G_{n-2}^T$ on the third last and second last row, will annihilate all the elements in the second last row, except the last two elements (note that the Givens transformations by construction have the determinant equal to 1). In this fashion we can continue very easily to annihilate all the elements in the strictly lower triangular part. In fact we are only interested in the diagonal elements. Performing these transformations will change the diagonal elements in the following way. Denote with $d_i$ the diagonal elements of the semiseparable matrix, and with $d_i^{(s)}$ the super diagonal elements. The diagonal elements $i = 2, \ldots, n$ change by performing:

$$G_i^T \left( \begin{array}{c} d_{i-1}^{(s)} \\ d_i \end{array} \right) = \left( \begin{array}{c} \hat{d}_{i-1}^{(s)} \\ \hat{d}_i \end{array} \right)$$

where the elements $\hat{d}_i$ denote the diagonal elements of the upper triangular matrix $R$. Using this information one can easily deduce an order $n$ algorithm for calculating the determinant.

More information about the $QR$-factorization, from a computational point of view, can be found in Chapter 9 where the more general class of semiseparable plus diagonal matrices is considered. More information about the structure of the $Q$ factor and the $R$ factor, when calculating the $QR$-factorization of a semiseparable plus diagonal matrix can be found in Chapter 1.

## Conclusions

In this chapter we first presented a definition for a representation of a class of matrices. After investigating two types of existing representations for semiseparable matrices, i.e. the diagonal subdiagonal representation and the generator definition, we designed a new representation based on Givens transformations and a vector. We showed that this representation is the most general one. Examples and an algorithm to compute the Givens-vector representation were included. In a final section of this chapter we constructed some algorithms, working directly with the Givens-vector representation.

CHAPTER 3

# An overview of semiseparable matrices.

In this chapter some applications and early appearances of semiseparable matrices are investigated. This chapter is not essential for a full understanding of the forthcoming chapters. It is merely intended to present some interesting applications in which semiseparable matrices appear. For several of these applications the computation of the eigenvalues and eigenvectors plays an important role. This motivates the search for efficient and accurate algorithms to compute the eigendecomposition of semiseparable matrices. Also an historical overview of publications closely related to this subject is included.

In the first section we examine the properties of oscillation matrices. After the definition and some properties of this type of matrices are given, some examples are considered. It is shown that Jacobi (i.e. irreducible tridiagonal matrices) and symmetric generator representable semiseparable matrices are oscillation matrices. Moreover, to our knowledge, the most early proof that the inverse of an irreducible symmetric tridiagonal matrix is a generator representable semiseparable matrix is included. The proof is constructive, i.e., the inverse of the Jacobi matrix is explicitely calculated via determinantal formulas. After this proof the physical interpretation of such a generator representable matrix will be given. It is shown that the vibrational properties of a string are related to this matrix. To conclude the chapter it is shown that the properties of eigenvalues and eigenvectors of oscillation matrices are directly linked with the physical interpretation of such an oscillation system.

Semiseparable matrices do not only appear in physical applications, but also in the field of statistics. In Section 2 we investigate some of these matrices. We calculate the covariance matrix for a multinomial distribution. It will be shown that this matrix is a diagonal plus semiseparable matrix. Other multivariate distributions for which the covariance matrices have the semiseparable plus diagonal form, are included. In some of the statistical applications the eigenvalues and/or eigenvectors of these covariance matrices are desired.

In Section 3 an integral equation with a so-called Green's kernel is discretized via the trapezoidal rule. The resulting matrix is of semiseparable plus diagonal form.

In Section 4 the connection between orthogonal rational functions and semiseparable plus diagonal matrices is briefly investigated.

Section 5 is completely dedicated to an historical overview of the literature related to the semiseparable subject. It is interesting to see that the same results are sometimes obtained independently in different research fields. Finally some paragraphs are included about some applications and/or new research fields related to semiseparable matrices.

## 1. Oscillation matrices

These matrices appeared for the first time in the book of Gantmacher and Krein [**83**], which was published in Russia in 1950. The book was written in Russian. Several translations were made of it, first into the German tongue (1960) and afterwards (1961) also into English. The complete book as it appeared in 2002 is based on these three references. Because of the many translations, the original Russian name the authors gave to generator representable semiseparable matrices was translated in several papers and books into one-pair matrices, while other authors translated it into single-pair matrices. In our survey of this book we will use the name one-pair matrix. In fact these matrices are a special sort of semiseparable matrices, and as we will prove they have strong connections with "Jacobi" matrices. Jacobi matrices are nowadays more commonly known as irreducible tridiagonal matrices. Here, the origin of oscillation matrices in physics will be briefly mentioned. It is also shown that a symmetric generator representable semiseparable matrix, under some extra conditions, can be considered as an oscillation matrix. Also a proof based on properties of determinants is included, to show that the inverse of a one-pair matrix is a Jacobi matrix.

**1.1. Introduction.** Let us consider small transverse oscillations of a linear elastic continuum (e.g., a string or a rod), spread along the x-axis ranging from $x = a$ to $x = b$. The natural harmonic oscillation of the continuum is given by:

$$y(x,t) = \varphi(x)\sin(pt + \alpha).$$

The function $y(x,t)$ denotes the deflection at point $x$ at time $t$, $\varphi(x)$ stands for the amplitude at point $x$, $\alpha$ for the phase and $p$ for the frequency of the oscillation. A so-called segmental continuum has the following main oscillation properties (from [**83**, Introduction]):

(1) All the frequencies $p$ are simple. (This means that the amplitude function of a given frequency is uniquely determined up to a constant factor.)
(2) At frequency $p_i$ the oscillation has exactly $i$ nodes. (Suppose $x$ to be a node, this means that $\varphi(x) = 0$.)
(3) The nodes of two successive overtones alternate.
(4) When superposing natural oscillations with frequencies $p_k < p_l < \ldots < p_m$, the number of sign changes of the deflection fluctuates with time within the limits from $k$ to $m$.

Now we will construct a so-called oscillation matrix, and in [**83**] it is shown that all the properties mentioned above are strictly connected to the properties of the oscillation matrix. Suppose our oscillation system consists of $n$ masses $m_1, m_2, \ldots, m_n$, located at points $s_1, s_2, \ldots, s_n$. Note that the masses are not fixed but movable in the direction of the $y$-axis. With $K(x,s)$ we denote the so-called influence function. It denotes the deflection at point $x$ under the influence of a unit force at point $s$. Denote $a_{ik} = K(x_i, s_k)$. Assume $y_1, y_2, \ldots, y_n$ to be the deflection of these masses, under the influence of a force:

$$-m_k \frac{d^2 y_k}{dt^2} \quad (k = 1, 2, \ldots, n).$$

We have that the deflection at point $x$ at time $t$ can be written as

$$y(x,t) = -\sum_{k=1}^{n} K(x, s_k) m_k \frac{d^2 y_k}{dt^2}.$$

For $x$ equal to $s_1, s_2, \ldots, s_n$ we have

$$y_i = -\sum_{k=1}^{n} a_{ik} m_k \frac{d^2 y_k}{dt^2}.$$

When denoting the amplitude of the deflection as $u_i = \varphi(s_i)$ in the harmonic oscillation equation we get

$$y_i = u_i \sin(pt + \alpha)$$

which leads after differentiation of $y_i$ to the following system of equations:

$$u_i = p^2 \sum_{k=1}^{n} a_{ik} m_k u_k \quad (k = 1, 2, \ldots, n)$$

which can be rewritten, to obtain the following system of equations:

$$\begin{cases} (1 - p^2 a_{11} m_1) u_1 - p^2 a_{12} m_2 u_2 - \cdots - p^2 a_{1n} m_n u_n & = & 0 \\ -p^2 a_{21} m_1 u_1 + (1 - p^2 a_{22} m_2) u_2 - \cdots - p^2 a_{2n} m_n u_n & = & 0 \\ \vdots & & \vdots \\ -p^2 a_{n1} m_1 u_1 - p^2 a_{n2} m_2 u_2 - \cdots + (1 - p^2 a_{nn} m_n) u_n & = & 0. \end{cases}$$

Which has a solution if:

$$\det \begin{pmatrix} 1 - p^2 a_{11} m_1 & -p^2 a_{12} m_2 & \cdots & -p^2 a_{1n} m_n \\ -p^2 a_{21} m_1 & 1 - p^2 a_{22} m_2 & \cdots & p^2 a_{2n} m_n \\ \vdots & & \ddots & \vdots \\ -p^2 a_{n1} m_1 & -p^2 a_{n2} m_2 & \cdots & 1 - p^2 a_{nn} m_n \end{pmatrix} = 0$$

revealing thereby the possible frequencies of the oscillation.

As already mentioned, in [**124**] Krein observed that the oscillation properties, given above, are closely related to the influence coefficients $a_{ik}$. Even more, the oscillation properties are related to the fact that all minors (a minor is the determinant of a submatrix of the given matrix) of the matrix $(a_{ik})$ (of all orders) have to be nonnegative. The theory connected to these matrices is called the theory of oscillation matrices.

Definition 34. An oscillation matrix $A$ is a matrix such that all minors of all orders of this matrix (principal and nonprincipal) are nonnegative. A minor of $A$ is the determinant of a square submatrix of $A$.

Important properties of these oscillation matrices, and their connections with the oscillation properties where investigated in the papers [**80, 81, 82**]. Also other authors show their interest in this field of matrices [**64, 75, 76**], especially the article [**64**] contains several references related to positive and nonnegative matrices. The choice of the term *oscillation matrix* by the authors of the book [**83**] comes from the following circumstance: (citation from [**83**], Introduction, page 3)

> As soon as for a finite system of points, the matrix of coefficients of influence of a given linear elastic continuum is an oscillation matrix (as it always is in the case for a string or a rod supported at the endpoints in the usual manner), this automatically implies the oscillation properties of the vibration of the continuum, for any distribution of masses at these points.

The theory of oscillation matrices (sometimes also other matrix theories), has an analogue in the theory of integral equations. For the oscillation matrices this corresponds to the theory of the following integral equation:

$$\varphi(x) = \lambda \int_a^b K(x,s)\varphi(s)d\sigma(s),$$

with an oscillation kernel $K(x,s)$.

Definition 35. A kernel $K(x,s)$ is called an oscillation kernel if for every choice of $x_1 < x_2 < \ldots < x_n$ in the interval $[a,b]$, the matrix $K(x_i,x_k)$, with $(i,k \in \{1,2,\ldots,n\})$ is an oscillation matrix.

It follows from the definition that an oscillation kernel is characterized by the following inequalities:

- $\det(K(x_i,s_k)) \geq 0$ for every choice of points $a < x_1 < x_2 < \ldots < x_n < b$ and $a < s_1 < s_2 < \ldots < s_n < b$ where the equality sign should be omitted when $x_i = s_i$;
- $K(x,s) > 0$ for $a < x < b$ and $a < s < b$.

This type of integral equation for $d\sigma = ds$ was investigated in [**118, 119**]. The more general case for a nonsymmetric kernel and an increasing function $\sigma(s)$ is studied for example in [**79**]. In Section 3 of this chapter we will show that discretization of this integral equation also results in solving a system of equations with a semiseparable matrix as coefficient matrix.

For the purpose of our thesis this introduction into the theory of oscillation matrices is enough. In the remainder of this section we will take a closer look at two types of oscillation matrices: Jacobi matrices and one-pair matrices. Jacobi matrices are a special type of tridiagonal matrices, namely irreducible ones, and one-pair matrices are a special type of semiseparable matrices, namely symmetric generator representable semiseparable matrices of semiseparability rank 1. Also attention will be paid to the most important properties of these matrices, which are of interest in the theory of oscillation matrices.

**1.2. Definition and examples.** First some new terms have to be defined:

Definition 36 (Chapter II, Definition 4 in [**83**]). A matrix $A$ is called totally nonnegative (or totally positive) if all its minors of any order are nonnegative (or positive).

In [**83**], an oscillation matrix is defined in the following way (which is an extension of Definition 34):

DEFINITION 37 (Chapter II, Definition 6 in [**83**]). A matrix $A$ is called an oscillation matrix if $A$ is totally nonnegative and there exists a positive integer $\kappa$ such that $A^\kappa$ is totally positive.

Some interesting properties of oscillation matrices:

PROPOSITION 38. *If $A$ is an oscillation matrix, then we have that*

- *$A$ is nonsingular;*
- *also $A^p$ with $p$ an integer is an oscillation matrix;*
- *also $\left(A^{-1}\right)^T$ is an oscillation matrix.*

Some examples of totally positive and totally nonnegative matrices are given. (Note that a totally positive matrix is already an oscillation matrix). The corresponding proofs can be found in [**83**, Chapter II, p. 76].

EXAMPLE 19. A generalized Vandermonde matrix:

$$A = (a_i^{\alpha_k})_{ik} = \begin{pmatrix} a_1^{\alpha_1} & a_1^{\alpha_2} & \dots & a_1^{\alpha_n} \\ a_2^{\alpha_1} & a_2^{\alpha_2} & \dots & a_2^{\alpha_n} \\ \vdots & & \ddots & \vdots \\ a_n^{\alpha_1} & a_n^{\alpha_2} & \dots & a_n^{\alpha_n} \end{pmatrix}$$

with $0 < a_1 < a_2 < \dots < a_n$ and $\alpha_1 < \alpha_2 < \dots < \alpha_n$, is totally positive and therefore an oscillation matrix.

EXAMPLE 20. The Cauchy matrix:

$$A = \left(\frac{1}{x_i + y_k}\right)_{ik} = \begin{pmatrix} \frac{1}{x_1+y_1} & \frac{1}{x_1+y_2} & \cdots & \frac{1}{x_1+y_n} \\ \frac{1}{x_2+y_1} & \frac{1}{x_2+y_2} & \cdots & \frac{1}{x_2+y_n} \\ \vdots & & \ddots & \vdots \\ \frac{1}{x_n+y_1} & \frac{1}{x_n+y_2} & \cdots & \frac{1}{x_n+y_n} \end{pmatrix}$$

with $0 < x_1 < x_2 < \dots < x_n$ and $0 < y_1 < y_2 < \dots < y_n$ is totally positive and therefore an oscillation matrix.

EXAMPLE 21. A one-pair matrix $S$ with elements:

$$s_{i,j} = \begin{cases} u_i v_j & (i \geq j) \\ u_j v_i & (i \leq j) \end{cases}$$

with all the elements $u_i$ and $v_j$ different from zero is totally nonnegative if and only if, all the numbers $u_i$ and $v_j$ have the same sign and:

$$\frac{v_1}{u_1} \leq \frac{v_2}{u_2} \leq \dots \leq \frac{v_n}{u_n}. \tag{14}$$

Moreover the rank of the matrix $S$ is equal to the number of "<" signs in (14) plus one. Note that the fact that the matrix is totally nonnegative does not imply that it is an oscillation matrix.

EXAMPLE 22. A Jacobi matrix $J$:

$$J = \begin{pmatrix} a_1 & b_1 & 0 & \dots & 0 \\ c_1 & a_2 & b_2 & & \\ 0 & c_2 & a_3 & & \\ \vdots & & \ddots & \ddots & b_{n-1} \\ 0 & \dots & 0 & c_{n-1} & a_n \end{pmatrix}$$

with all elements $b_i$ and $c_i$ different from zero, and the successive principal minors positive, is totally nonnegative.

As already stated in the example of the one-pair matrices, the condition that the matrix is totally nonnegative is not sufficient to form an oscillation matrix. the following theorem can help us in the case of the one-pair and Jacobi matrices.

THEOREM 39 (Chapter II, Theorem 10 in [**83**]). *It is necessary and sufficient for a totally nonnegative matrix $A$ that:*

- *$A$ is a nonsingular matrix;*
- *$a_{i,i+1} > 0$ and $a_{i+1,i} > 0$ for all $i = 1, 2, \dots, n-1$;*

*in order to be an oscillation matrix.*

With this theorem one can clearly see that one-pair matrices, plus one extra demand, are oscillation matrices and the only extra demand which has to be placed on the Jacobi matrix is that the sub and superdiagonal elements have to be positive.

We know now what an oscillation matrix is and also that under certain circumstances a tridiagonal and a semiseparable matrix will be oscillation matrices. In the following subsections we will explain, why the eigenvalues and eigenvectors of oscillation matrices are important, in connection with the physical interpretation. Also for the one-pair matrix we will directly from the physical application construct the matrix. Although we have already given a proof in Chapter 1, we will prove here in the old-fashioned way that the inverse of a one-pair matrix is a Jacobi matrix.

**1.3. The inverse of a one-pair matrix.** This section is summarizes the results of [**83**, Chapter II]. The definitions and theorems are adapted such that they fit our notation style. Our main result will be to prove, as Gantmacher and Krein did, that the inverse of a one-pair matrix is a Jacobi one and vice versa. Originally the one-pair matrices were defined as (equivalent to Example 21):

DEFINITION 40. A one-pair matrix is a symmetric matrix $S$ such that

$$s_{ij} = \begin{cases} u_i v_j & (i \geq j) \\ u_j v_i & (i \leq j) \end{cases}$$

where the elements $u_i$ and $v_j$ are chosen arbitrarily.

Interesting to see is that the proof is completely based on properties of determinants of matrices. Around 1900 several books were written, completely devoted to theoretical results concerning determinants of matrices, e.g., [**5, 28, 41, 86, 87, 141, 144, 159**]. Several of these books can be found in the online historical library at Cornell: `http://historical.library.cornell.edu/math/`.

THEOREM 41. *Suppose $A$ is a symmetric Jacobi matrix, i.e., a symmetric tri-diagonal matrix of size $n$:*

$$\begin{pmatrix} a_1 & b_1 & & & \\ b_1 & a_2 & b_2 & & \\ & b_2 & a_3 & \ddots & \\ & & \ddots & \ddots & b_{n-1} \\ & & & b_{n-1} & a_n \end{pmatrix}$$

*with all the $b_i$ different from zero. The inverse of $A$ will be a one-pair matrix.*

The proofs of the different theorems and corollaries are mainly based on formulas for the determinants of matrices. First some notation has to be introduced. Suppose we have an arbitrary $n \times n$ matrix $A$ which is denoted as $A = (a_{i,j})_{i,j\in\{1,\ldots,n\}}$.

DEFINITION 42. Define the matrix $A(i_1, \ldots, i_p; j_1, \ldots, j_p)$ as the matrix:

$$A(i_1, \ldots, i_p; j_1, \ldots, j_p) = (a_{i,j})$$

with the indices belonging to the following sets:

$$\begin{aligned} i &\in \{i_1, \ldots, i_p\} \\ j &\in \{j_1, \ldots, j_p\}. \end{aligned}$$

For the determinant of a matrix we will temporarily use the shorter notation:

$$\det(A) = |A|.$$

Before we can prove the first important statement, a proposition is needed.

PROPOSITION 43. *Suppose a Jacobi matrix (irreducible tridiagonal matrix) $A$ of size $n$ is given. If*

$$\begin{aligned} 1 \le i_1 < i_2 < \ldots < i_p \le n \\ 1 \le j_1 < j_2 < \ldots < j_p \le n \end{aligned} \tag{15}$$

*and*

$$\begin{aligned} & i_1 = j_1, i_2 = j_2, \ldots, i_{\nu_1} = j_{\nu_1}, \\ & i_{\nu_1+1} \neq j_{\nu_1+1}, \ldots, i_{\nu_2} \neq j_{\nu_2}, \\ & i_{\nu_2+1} = j_{\nu_2+1}, \ldots, i_{\nu_3} = j_{\nu_3}, \\ & i_{\nu_3+1} \neq j_{\nu_3+1}, \ldots \end{aligned}$$

*then*

$$\begin{aligned} & |A(i_1, \ldots, i_p; j_1, \ldots, j_p)| \\ & \quad = \quad |A(i_1, \ldots, i_{\nu_1}; j_1, \ldots, j_{\nu_1})| \cdot |A(i_{\nu_1+1}; j_{\nu_1+1})| \ldots \\ & \qquad\quad |A(i_{\nu_2}; j_{\nu_2})| \cdot |A(i_{\nu_2+1}, \ldots, i_{\nu_3}; j_{\nu_2+1}, \ldots, j_{\nu_3})| \ldots \end{aligned}$$

PROOF. We will prove that under the conditions (15), and $i_\nu \neq j_\nu$ the following equations hold:

$$\begin{aligned} & |A(i_1, \ldots, i_p; j_1, \ldots, j_p)| \\ & \quad = \quad |A(i_1, \ldots, i_\nu; j_1, \ldots, j_\nu)| \cdot |A(i_{\nu+1}, \ldots, i_p; j_{\nu+1}, \ldots, j_p)| \\ & \quad = \quad |A(i_1, \ldots, i_{\nu-1}; j_1, \ldots, j_{\nu-1})| \cdot |A(i_\nu, \ldots, i_p; j_\nu, \ldots, j_p)|. \end{aligned} \tag{16}$$

Proving the first of the two equations is enough, to derive the complete desired result. If $i_\nu < j_\nu$ then we have, because the matrix $A$ is tridiagonal:

$$a_{i_\lambda j_\mu} = 0 \quad (\lambda = 1, 2, \ldots, \nu; \mu = \nu + 1, \ldots, p);$$

otherwise, if $i_\nu > j_\nu$ would lead to:

$$a_{i_\lambda j_\mu} = 0 \quad (\lambda = \nu + 1, \ldots, p; \mu = 1, 2, \ldots, \nu).$$

These two last statements say that the matrix is either upper block triangular, or lower block triangular. This proves (16). □

We can now prove that the inverse of a symmetric Jacobi matrix is a so-called one-pair matrix.

PROOF (Proof of Theorem 41). We prove the theorem by explicitely constructing the inverse of the symmetric Jacobi matrix $A$. This matrix will then appear to be a one-pair matrix. Suppose $S$ is the inverse of the matrix $A$. This means that:

$$s_{i,j} = \frac{1}{\det(A)}(-1)^{i+j}|A(1, \ldots, i-1, i+1, \ldots, n; 1, \ldots, j-1, j+1, \ldots, n)|.$$

We distinguish between two cases now:

(1) When $i \leq j$

$$\begin{aligned} s_{i,j} &= \frac{1}{\det(A)}(-1)^{i+j}|A(1, \ldots, i-1; 1, \ldots, i-1)| \cdot |A(i+1, i)| \cdots \\ &\quad |A(j, j-1)| \cdot |A(j+1, \ldots, n; j+1, \ldots, n)| \\ &= \frac{1}{\det(A)}(-1)^{i+j}|A(1, \ldots, i-1; 1, \ldots, i-1)|\, b_i\, b_{i+1} \ldots \\ &\quad b_{j-1}\, |A(j+1, \ldots, n; j+1, \ldots, n)|. \end{aligned}$$

(2) When $i \geq j$ , we can do the same as above and one gets

$$\begin{aligned} s_{i,j} &= \frac{1}{\det(A)}(-1)^{i+j}|A(1, \ldots, j-1; 1, \ldots, j-1)|\, b_j\, b_{j+1} \ldots \\ &\quad b_{i-1}\, |A(i+1, \ldots, n; i+1, \ldots, n)|. \end{aligned}$$

When writing $u_i$ and $v_j$ now as: (Under the assumption that all the $b_i \neq 0$)

$$\begin{aligned} v_i &= \frac{(-1)^i}{\det(A)}|A(1, \ldots, i-1; 1, \ldots, i-1)|\, b_i\, b_{i+1} \ldots b_{n-1} \\ u_i &= \frac{(-1)^i |A(i+1, \ldots, n; i+1, \ldots, n)|}{b_i\, b_{i+1} \ldots b_{n-1}}, \end{aligned}$$

we get that

$$s_{i,j} = \begin{cases} u_i v_j & (i \geq j) \\ u_j v_i & (i \leq j) \end{cases}.$$

This proves the theorem. □

The following properties are necessary to prove that the inverse of a one-pair matrix is an irreducible tridiagonal matrix.

PROPOSITION 44. *Suppose we have a one-pair matrix $S$ which is generated by the vectors $u$ and $v$. If*

$$1 \leq i_1, j_1 < i_2, j_2 < \ldots < i_p, j_p \leq n \tag{17}$$

*then*

$$S(i_1, \ldots, i_p; j_1, \ldots, j_p) = v_{\alpha_1} \begin{vmatrix} u_{\beta_1} & u_{\alpha_2} \\ v_{\beta_1} & v_{\alpha_2} \end{vmatrix} \begin{vmatrix} u_{\beta_2} & u_{\alpha_3} \\ v_{\beta_2} & v_{\alpha_3} \end{vmatrix} \ldots \begin{vmatrix} u_{\beta_{p-1}} & u_{\alpha_p} \\ v_{\beta_{p-1}} & v_{\alpha_p} \end{vmatrix} u_{\beta_p}$$

*where*

$$\alpha_\nu = \min(i_\nu, j_\nu) \quad \beta_\nu = \max(i_\nu, j_\nu).$$

PROOF. Because the matrix $S$ is a symmetric matrix, we can, without loss of generality, assume that $i_2 \leq j_2$. This means that $\alpha_2 = i_2$ and $\beta_2 = j_2$, leading to the following equality:

$$\begin{aligned} & |S(i_1, \ldots, i_p; j_1, \ldots, j_p)| \\ = \quad & \det \begin{pmatrix} u_{\beta_1} v_{\alpha_1} & u_{j_2} v_{i_1} & u_{j_3} v_{i_1} & \ldots & u_{j_p} v_{i_1} \\ u_{i_2} v_{j_1} & u_{j_2} v_{i_2} & u_{j_3} v_{i_2} & \ldots & u_{j_p} v_{i_2} \\ \vdots & & \ddots & & \end{pmatrix}. \end{aligned}$$

Subtracting from the first row the second one multiplied with $v_{i_1}/v_{i_2}$ gives the following equation:

$$\begin{aligned} & |S(i_1, \ldots, i_p; j_1, \ldots, j_p)| \\ = \quad & \left( v_{\alpha_1} u_{\beta_1} - \frac{v_{j_1} u_{i_2} v_{i_1}}{v_{i_2}} \right) |S(i_2, \ldots, i_p; j_2, \ldots, j_p)| \\ = \quad & \frac{v_{\alpha_1}}{v_{\alpha_2}} \begin{vmatrix} u_{\beta_1} & u_{\alpha_2} \\ v_{\beta_1} & v_{\alpha_2} \end{vmatrix} |S(i_2, \ldots, i_p; j_2, \ldots, j_p)|. \end{aligned}$$

Applying the equation above successively and using the fact that

$$S(i_p; j_p) = v_{\alpha_p} u_{\beta_p}$$

gives the desired result. □

One more proposition about the minors of a one-pair matrix is needed.

PROPOSITION 45. *Suppose we have a one-pair matrix $S$. If*

$$1 \leq i_1 < i_2 < \ldots < i_p \leq n$$
$$1 \leq j_1 < j_2 < \ldots < j_p \leq n,$$

*but equation (17) is not satisfied then:*

$$|S(i_1, \ldots, i_p; j_1, \ldots, j_p)| = 0.$$

PROOF. Using Proposition 44 and assuming that

$$1 \leq i_1, j_1 < i_2, j_2 < \ldots < i_r, j_r$$

whereas for example $j_r > i_{r+1}$, we get that

$$\begin{aligned} & |S(i_1, \ldots, i_p; j_1, \ldots, j_p)| \\ = \quad & \frac{u_{\alpha_1}}{u_{\alpha_2}} \begin{vmatrix} v_{\beta_1} & v_{\alpha_2} \\ u_{\beta_1} & u_{\alpha_2} \end{vmatrix} \ldots \begin{vmatrix} v_{\beta_{r-1}} & v_{\alpha_r} \\ u_{\beta_{r-1}} & u_{\alpha_r} \end{vmatrix} |S(i_r, \ldots, i_p; j_r, \ldots, j_p)|. \end{aligned}$$

Because $j_r > i_{r+1}$, this means that the last determinant is zero. □

Now the theorem stating that the inverse of a one-pair matrix is a Jacobi matrix can be stated.

THEOREM 46. *Suppose $S$ is a one-pair matrix with all the elements of the generators different from zero, then the inverse of $S$ is a tridiagonal matrix with all the sub and superdiagonal elements different from zero.*

PROOF. It is easily proved by using Propositions 44 and 45. □

**1.4. The example of the one-pair matrix.** This subsection is based on [**83**, Section 7, Chapter III] Suppose that we have a string fastened in the points $x = 0$ and $x = l$, with a tension $T$. We will now deduce the influence function $K(x, s)$ of this system.

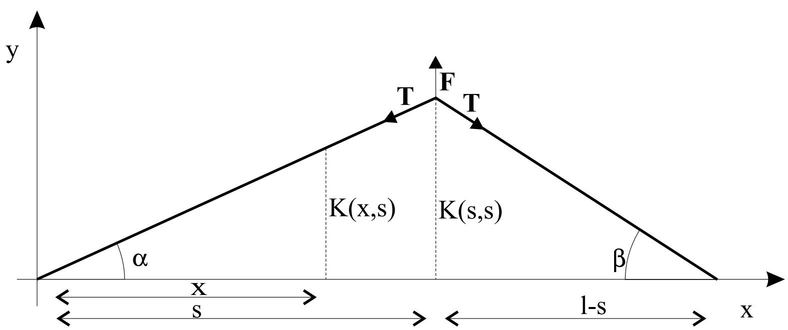


FIGURE 3.1: Constructing the influence function of a string

Figure 1.4 should help to understand the following part. If we pull the string with a unit force $F$ in the point $s$, we know that the tension must satisfy the following equality

$$T(\sin(\alpha) + \sin(\beta)) = 1.$$

Because the angles $\alpha$ and $\beta$ are very small we can assume that

$$\begin{aligned} \sin(\alpha) &= \frac{K(s,s)}{s} \\ \sin(\beta) &= \frac{K(s,s)}{l-s}. \end{aligned}$$

Using these equations combined with the equation in $T$ we obtain

$$K(s,s) = \frac{(l-s)s}{Tl}.$$

When we want to calculate the deflection at an arbitrary point $x$ now, we use the trigonometric equality:

$$\frac{K(x,s)}{x} = \frac{K(s,s)}{s}$$

which gives us the following equation:

$$K(x,s) = \begin{cases} \dfrac{x(l-s)}{Tl} & (x \leq s), \\ \dfrac{s(l-x)}{Tl} & (x \geq s). \end{cases}$$

Taking now a finite number of points on the string: $x_1 < x_2 < \ldots < x_n$, we have that:

$$K(x_i, x_k) = \begin{cases} u_i v_k & (i \geq k) \\ u_k v_i & (i \leq k) \end{cases}$$

with

$$v_i = \frac{x_i}{Tl}, \text{ and } u_k = l - x_k \text{ for } (i = 1, 2, \ldots, n)$$

which clearly is a one-pair matrix. Moreover, if one of the two ends of the string is not fixed, one gets again a one-pair matrix.

**1.5. Some other interesting applications.** This subsection is based on [**83**, Section 8-11, Chapter III]. Jacobi matrices are closely related to oscillations of Sturm systems. We will not go into the details of what exactly a Sturm system is but we will give some examples. For example an oscillating thread with beads. Also a shaft on which a number of disks are fastened, all turning at the same speed, gives rise to a Sturm system when looking at the motion of the shaft, under an initial impulse. Also a pendulum, with not one mass, but different masses can be considered as a Sturm system. Such a pendulum consists of a thread attached at a fixed point, and hanging down, at certain places at the thread, masses are attached. The swinging movement of the system satisfies the conditions for a Sturm system.

When one investigates the influence function of a rod, one constructs an influence function which is in fact the sum of two one-pair matrices. This means that the upper triangular part can be considered as coming from a rank 2 and also the lower triangular part can be considered as coming from a rank 2 matrix.

**1.6. The connection with eigenvalues and eigenvectors.** To see the connection with the physical interpretation we have to take a closer look at the following theorem, but first we need to define what is meant by the number of sign changes. Suppose we have a sequence of numbers $u_1, u_2, \ldots, u_n$, the number of sign changes reading this sequence from the left to the right, is of course dependent of the signs we give to the $u_i$'s which are equal to zero. Nevertheless we can speak about the minimal number of sign changes and the maximum number of sign changes.

THEOREM 47 (Theorem 6 in [**83**]). *Suppose $A$ is an oscillation matrix, then this matrix has the following properties:*

(1) *All the eigenvalues of $A$ are positive and single.*
(2) *If $v_k$ is an eigenvector of $A$, corresponding to the eigenvalue $\lambda_k$, then for every sequence of coefficients $c_p, c_{p+1}, \ldots, c_q$ (with $1 \leq p \leq q \leq n$ and not all $c_i$ equal to zero) we have that the number of sign changes in the coordinates of the vector:*

$$u = c_p v_p + c_{p+1} v_{p+1} + \cdots c_q v_q$$

*lies between $p-1$ and $q-1$. More precisely, for the vector $v_k$ there are exactly $k-1$ sign changes.*

(3) *The nodes of two successive eigenvectors $v_k$ and $v_{k+1}$ alternate.*

One can fairly easily see the connection between the properties of the oscillations in Section 1.1 and Theorem 47. Property 1 of Theorem 47 corresponds to Property 1 in Section 1.1. Property 2 of the theorem corresponds with 2 and 4 of Section 1.1 and Property 3 corresponds with 3 in Section 1.1. This reveals the close connection between the eigenvalues and eigenvectors of oscillation matrices and the properties of the oscillations of a segmental continuum.

## 2. Semiseparable matrices as covariance matrices

This section is mainly inspired by the book of A. G. Franklin [**95**]. (More information about the connection between semiseparable and covariance matrices can be found in [**8, 32, 78, 96, 97, 123, 146, 148, 157**].) Several covariance matrices from multivariate distributions will be constructed. Here only these matrices will be considered which have very strong connections with semiseparable matrices.

In the theory of multivariate analysis, the joint distribution of random variables is considered. Suppose $n$ random variables $X_1, \ldots, X_n$ are given. Statistical analysis is quite often concerned with the analysis of the covariance matrices $V$. These matrices have as diagonal elements the variances of the variables $X_i$ and as off-diagonal elements $v_{ij}$ the covariances between the variables $X_i$ and $X_j$. These matrices are symmetric and they contain very important information concerning relations between the set of random variables. Especially the inverse of these matrices is very important for statistical analysis (see, e.g. [**32, 97, 142**]).

In several examples, these matrices have structure, so-called patterned matrices in the book [**95**]. These structures are sometimes semiseparable, tridiagonal or semiseparable plus diagonal matrices, whose inverses can quite often be calculated in an easy way.

Here we will construct some of these matrices to show that semiseparable matrices also can be found in the field of statistics. The theorems proving how to calculate the inverses of these matrices are not explicitely given, they can be found in [**95**].

**2.1. The multinomial distribution.** Before defining the multinomial distribution, the binomial distribution is discussed. These comments are based on the book [**180**] and on the web pages: `http://www.stat.yale.edu/Courses/1997-98/101/binom.htm`. The use of the distribution is explained with an example. Briefly the binomial distribution describes the number of "success" outcomes when a Bernoulli experiment is repeated $n$ times, independently.

More precisely: the binomial distribution describes the behavior of a count variable $X$, which counts the number of successes in $n$ observations, if the following conditions are satisfied:

- The number of observations $n$ is fixed.
- Each observation is independent.
- Each observation represents one of two outcomes ("success" or "failure").
- The probability of "success" $p$ is the same for each outcome.

The binomial distribution is denoted as $B(n,p)$, with $n$ denoting the number of observations and $p$ the chance of success. This distribution is used in several examples, for example in:

EXAMPLE 23. Suppose we have a group of individuals with a certain gene. These people have a 0.60 probability of getting a certain disease. If a study is performed on 300 individuals, with this specific gene, then the distribution of the variable describing the number of people who will get the disease has the following binomial distribution: $B(300, 0.6)$.

EXAMPLE 24. The number of sixes rolled by a single die in 30 rolls has a binomial distribution $B(30, 1/6)$.

The distribution function for the binomial distribution $B(n,p)$ satisfies the following equation:

$$\mathrm{P}(X=k) = C(n,k)p^k(1-p)^{n-k}.$$

where

$$C(n,k) = \frac{n!}{k!(n-k)!}.$$

The mean and the variance of this distribution can be calculated, using the formulas above and are: (denote the mean as $\mu_X$ and the variance as $\sigma_X^2$)

$$\mu_X = np \qquad \sigma_X^2 = np(1-p).$$

Before giving the distribution function we will try to explain what is meant with a multinomial distribution. A multinomial trials process is a sequence of independent, identically distributed random variables $U_1, U_2, \ldots$, where each random variable can take now $k$ values. For the Bernoulli process, this corresponds to $k=2$, (success and failure). Therefore this is a generalization of a Bernoulli trials process. We denote the outcomes by the integers $1, 2, \ldots, k$, for simplicity. This means that for a trial variable $U_j$ we can write the distribution function in the following way.

$$p_i = \mathrm{P}(U_j = i) \text{ for } i = 1, 2, \ldots, k \text{ (and for any } j \text{ ).}$$

Of course $p_i > 0$ for each $i$ and $p_1 + p_2 + \cdots + p_k = 1$.

As with the binomial distribution, we are interested in the variables counting the number of times each outcome has occurred, where in the binomial case one variable for counting was enough, here we need $k-1$. Thus, let (with # we denote the cardinality of the set)

$$Z_i = \#\{j \in \{1, 2, \ldots, n\} \text{ for which } U_j = i\} \text{ for } i = 1, 2, \ldots, k,$$

where $n$ is the number of observations.

Note that

$$Z_1 + Z_2 + \cdots + Z_k = n.$$

So if we know the values of $k-1$ of the counting variables, we can find the value of the remaining counting variable as already mentioned before. Generalizing the binomial distribution we get the following function: (More information about these distributions can be found at `http://www.math.uah.edu/statold/bernoulli/`)

$$\mathrm{P}(Z_1 = j_1, Z_2 = j_2, \ldots, Z_n = j_k) = kp_1^{j_1}p_2^{j_2}\ldots p_k^{j_k}$$

with $j_1 + j_2 + \cdots + j_k = n$ and $p_1 + p_2 + \cdots + p_k = 1$. Before we start calculating the covariance matrix, we will give an example:

EXAMPLE 25. It is very easy to see that the dice experiment also fits in here. For example if one rolls 10 dice, we can calculate the probability that 1 and 2 occur once, and the other occur all two times. To calculate this, one needs the multinomial distribution.

For this distribution: $\mathrm{E}(Z_i) = np_i$, $\mathrm{Var}(Z_i) = np_i(1-p_i)$ and $\mathrm{Cov}(Z_i, Z_j) = np_ip_j$. As an example we calculate the mean using the following binonium, and multinonium formulas:

$$\begin{aligned}
C(n,j) &= \frac{n!}{(n-j)!j!} \\
(x+y)^n &= \sum_{j=0}^{n} C(n,j)x^{n-j}y^j \\
C(n;j_1,\ldots,j_k) &= \frac{n!}{j_1!\ldots j_k!} \\
(x_1+x_2+\cdots+x_k)^n &= \sum_{j_1+j_2+\cdots+j_k=n} C(n;j_1,\ldots,j_k)x_1^{j_1}\ldots x_k^{j_k}.
\end{aligned}$$

For this distribution we calculate the mean of the variable $X_1$, all the other variances and covariances can be calculated in a complete analogous way.

$$\begin{aligned}
\mathrm{E}(X_1) &= \sum_{j_1}\cdots\sum_{j_k} j_1\,\mathrm{P}(X_1 = j_1,\ldots,X_k = j_k) \\
&= \sum_{j_1}\cdots\sum_{j_k} j_1 C(n;j_1,\ldots,j_k)p_1^{j_1}\ldots p_k^{j_k} \\
&= \sum_{j_1} j_1 p_1^{j_1}\left(\sum_{j_2}\cdots\sum_{j_k}\frac{n!}{j_1!\ldots j_k!}p_2^{j_2}\cdots p_k^{j_k}\right)
\end{aligned}$$

$$\begin{aligned}
&= \sum_{j_1}\frac{j_1}{j_1!}p_1^{j_1}\left(\sum_{j_2}\cdots\sum_{j_k}\frac{n!(n-j_1)!}{(n-j_1)!j_2!\ldots j_k!}p_2^{j_2}\cdots p_k^{j_k}\right) \\
&= \sum_{j_1}\frac{j_1 n!}{j_1!(n-j_1)!}p_1^{j_1}(p_2+\cdots+p_k)^{j_2+\cdots+j_k} \\
&= \sum_{j_1} j_1\frac{n!}{j_1!(n-j_1)!}p_1^{j_1}(1-p_1)^{n-j_1} \\
&= \mathrm{E}(Z) \\
&= np_1
\end{aligned}$$

where $Z$ has a binomial distribution $B(n, p_1)$, which gives us the last equality. The covariance matrix of this distribution looks like

$$V = n \begin{pmatrix} p_1(1-p_1) & -p_1p_2 & -p_1p_3 & \dots & -p_1p_k \\ -p_1p_2 & p_2(1-p_2) & -p_2p_3 & \dots & -p_2p_k \\ -p_1p_3 & -p_2p_3 & p_3(1-p_3) & \dots & -p_3p_k \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -p_1p_k & -p_2p_k & -p_3p_k & \dots & p_k(1-p_k) \end{pmatrix}.$$

This matrix can be rewritten as a semiseparable plus diagonal matrix in the following way. Denote:

$$\begin{aligned} u &= [p_1, p_2, p_3, \dots, p_n]^T \\ v &= [-p_1, -p_2, -p_3, \dots, -p_n]^T \\ D &= \operatorname{diag}(p_1, p_2, \dots, p_n), \end{aligned}$$

then the matrix can be written, in a more compact form as

$$V = D + uv^T.$$

This is a semiseparable plus diagonal matrix.

The main reason of writing the covariance matrices in this form, is the simple expression of the inverses of this type of matrices. The book [**95**] states several theorems about the inverses of tridiagonal and semiseparable matrices, resulting in an inversion formula for this type of matrices, namely:

$$V^{-1} = D^{-1} + \gamma \hat{u}\hat{v}^T$$

with $\hat{u}_i = u_i/d_i$, $\hat{v}_i = v_i/d_i$ and $\gamma = -\left(1 + \alpha \sum_{i=1}^n u_i v_i d_i^{-1}\right)^{-1}$, with $d_i$ as the diagonal elements of $D$. This leads to the following explicit structure of the matrix $V^{-1}$,

$$V^{-1} = \begin{pmatrix} \frac{1}{p_1} + \gamma & \gamma & \gamma & \dots & \gamma \\ \gamma & \frac{1}{p_2} + \gamma & \gamma & \dots & \gamma \\ \gamma & \gamma & \frac{1}{p_3} + \gamma & & \vdots \\ \vdots & \vdots & & \ddots & \gamma \\ \gamma & \gamma & \dots & \gamma & \frac{1}{p_n} + \gamma \end{pmatrix}.$$

This matrix is clearly again semiseparable plus diagonal, as we expected according to the theorems of Chapter 1.

**2.2. Some other matrices.** To conclude this section we will briefly show two other matrices from the book [**156**]. No calculations are included anymore because they become too complicated For a sample of size $k$ of order statistics, with an exponential density, the covariance matrix looks like (the matrix is not symmetric anymore because the order has become important now):

$$\begin{pmatrix} \frac{1}{k^2} & \frac{1}{k^2} & \frac{1}{k^2} & \cdots & \frac{1}{k^2} \\ \frac{1}{k^2} & \frac{1}{k^2}+\frac{1}{(k-1)^2} & \frac{1}{k^2}+\frac{1}{(k-1)^2} & \cdots & \frac{1}{k^2}+\frac{1}{(k-1)^2} \\ \frac{1}{k^2} & \frac{1}{k^2}+\frac{1}{(k-1)^2} & \sum_{j=1}^{3}\frac{1}{(k-j+1)^2} & \cdots & \sum_{j=1}^{3}\frac{1}{(k-j+1)^2} \\ \vdots & \vdots & \vdots & & \vdots \\ \frac{1}{k^2} & \frac{1}{k^2}+\frac{1}{(k-1)^2} & \sum_{j=1}^{3}\frac{1}{(k-j+1)^2} & \cdots & \sum_{j=1}^{k}\frac{1}{(k-j+1)^2} \end{pmatrix}.$$

This matrix is clearly a semiseparable matrix and the inverse can easily be calculated by the algorithms proposed in [**95**], and looks like:

$$\begin{pmatrix} k^2+(k-1)^2 & -(k-1)^2 & 0 & 0 & \cdots \\ -(k-1)^2 & (k-1)^2+(k-2)^2 & -(k-2)^2 & 0 & \cdots \\ 0 & -(k-2)^2 & (k-2)^2+(k-3)^2 & -(k-3)^2 & \\ 0 & 0 & -(k-3)^2 & \ddots & \ddots \\ \vdots & \vdots & & \ddots & \end{pmatrix}$$

The following matrix arises in the statistical theory of ordered observations:

$$\begin{pmatrix} k & k-1 & k-2 & k-3 & \cdots & 1 \\ k-1 & 2(k-1) & 2(k-2) & 2(k-3) & \cdots & 2 \\ k-2 & 2(k-2) & 3(k-2) & 3(k-3) & \cdots & 3 \\ k-3 & 2(k-3) & 3(k-3) & 4(k-3) & \cdots & 4 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 2 & 3 & 4 & \cdots & k \end{pmatrix}.$$

This matrix is again symmetric and semiseparable. The last matrix arises in the theory of stationary time series:

$$\sigma^2 \begin{pmatrix} 1 & \rho & \rho^2 & \rho^3 & \cdots & \rho^{n-1} \\ \rho & 1 & \rho & \rho^2 & \cdots & \rho^{n-2} \\ \rho^2 & \rho & 1 & \rho & \cdots & \rho^{n-3} \\ \rho^3 & \rho^2 & \rho & 1 & \cdots & \rho^{n-4} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \rho^{n-1} & \rho^{n-2} & \rho^{n-3} & \rho^{n-4} & \cdots & 1 \end{pmatrix}.$$

Also this matrix is symmetric and semiseparable. The details, of the origin of these matrices are not included, this would lead us too far away from the basic subject of the thesis. An explicit formula for the inverses of these matrices can also be calculated rather easily. The inverses and the corresponding theorems can be found in [**95**].

We will end this section with a quote ([**95**, Introduction])

> The theory of multivariate analysis often centers around an analysis of a covariance matrix $V$. When this is the case, it may be necessary to find the determinant of $V$, the characteristic roots of $V$, the inverse of $V$ if it exists, and perhaps to determine these and other quantities for certain submatrices of $V$.

The quote shows that it can be important to calculate the eigenvalues of these covariance matrices.

## 3. Discretization of integral equations

In this section a discretization of a special integral equation is made. The discretization will lead to a semiseparable system of equations. The resulting matrix will satisfy the semiseparable plus diagonal structure. More information about this topic can be found for example in [**100, 115**].

We discretize the following integral equation:

$$y(x) - \int_0^1 k(x,t)y(t)dt = a(x). \tag{18}$$

We want to compute the function $y(x)$ while $a(x)$ and $k(x,t)$ are known. The kernel $k(x,t)$ is called a Green's kernel, satisfying the following properties:

$$k(x,t) = \begin{cases} G(x)F(t) & x \leq t \\ F(x)G(t) & x > t \end{cases}.$$

More information about discretization techniques can be found in one of the following books [**27, 40, 84**]. We use the following discretization scheme, with the trapezoidal rule [**84**, p. 154]. Suppose a function $f(x)$ is given, and we want to integrate it in the interval $[a,b]$. Divide the interval in equal spaced smaller intervals of length $h$ by using the points $x_i$. Denote $f(x_i) = f_i$, then we get the following discretization scheme:

$$\int_a^b f(x)dx \approx \frac{h}{3}\left(f_0 + 4f_1 + 2f_2 + \cdots + 4f_{n-1} + f_n\right).$$

We will use this scheme to discretize the integral equation (18). Slicing the interval $[0,1]$ with a distance $h$ between the successive points $t_i$. Denoting $G(x_i) = G_i$, $F(x_i) = F_i$ and $a(x_i) = a_i$, we get:

$$\begin{aligned} a(x) \approx \quad & y(x) - \frac{h}{3}\Big(k(x,t_0)y(t_0) + 4k(x,t_1)y(t_1) + 2k(x,t_2)y(t_2) + \ldots \\ & + 4k(x,t_{n-1})y(t_{n-1}) + k(x,t_n)y(t_n)\Big). \end{aligned}$$

Substituting for $x$ the different values of $x_i = t_i$, we get the following system of equations:

$$\begin{aligned} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} &= \left(-\frac{h}{3}\right) \begin{pmatrix} G_0F_0 - \frac{3}{h} & 4G_0F_1 & 2G_0F_2 & \ldots & 4G_0F_{n-1} & G_0F_n \\ G_0F_1 & 4G_1F_1 - \frac{3}{h} & 2G_1F_2 & \ldots & 4G_1F_{n-1} & G_1F_n \\ G_0F_2 & 4G_1F_2 & \ldots & & & \\ \vdots & & \ddots & & & \\ G_0F_n & & & & & G_nF_n - \frac{3}{h} \end{pmatrix} \begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} \\ &= \left(-\frac{h}{3}\right) \begin{pmatrix} G_0F_0 - \frac{3}{h} & G_0F_1 & G_0F_2 & \ldots & G_0F_{n-1} & G_0F_n \\ G_0F_1 & G_1F_1 - \frac{3}{4h} & G_1F_2 & \ldots & G_1F_{n-1} & G_1F_n \\ G_0F_2 & G_1F_2 & \ldots & & & \\ \vdots & & \ddots & & & \\ G_0F_n & & & & & G_nF_n - \frac{3}{h} \end{pmatrix} \begin{pmatrix} y_0 \\ 4y_1 \\ 2y_2 \\ \vdots \\ y_n \end{pmatrix}. \end{aligned}$$

The last system of equations is a semiseparable plus diagonal matrix. Later on in this thesis, we will solve this system of equations in a stable and fast way.

When one wants to solve two point boundary value problems as in [**161**], one can sometimes translate these problems into a second order integral equation with a Green's kernel. In general this kernel is not semiseparable anymore, but so-called recursively semiseparable or sequentially semiseparable. This means that the matrices have the form as given in Figure 3, for which all the white blocks have the same rank. We will not go into the details of this type of structure, but the reader should

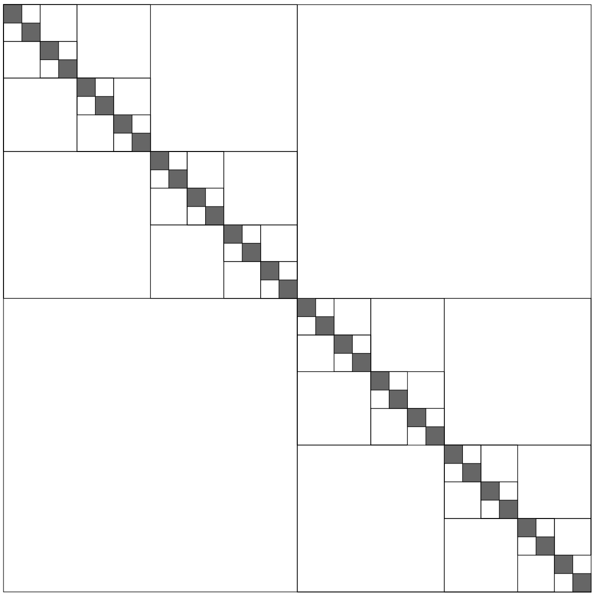

FIGURE 3.2: A recursively semiseparable matrix

know that this type of matrices appears in several types of applications, and there are already several techniques for solving this type of matrices, e.g. [**33, 34, 36, 48, 56**].

## 4. Orthogonal rational functions

This section will give a brief example in which way orthogonal rational functions are connected to semiseparable matrices. This section is mainly based on the paper [**175**], but also the papers [**25, 26, 67, 68, 176**] are closely related to this problem.

Suppose we have a vector space $\mathcal{V}_n$ of all proper rational functions having possible poles in $y_1, y_2, \ldots, y_n$:

$$\mathcal{V}_n = \text{Span}\left\{1, \frac{1}{z-y_1}, \frac{1}{z-y_2}, \ldots, \frac{1}{z-y_n}\right\}$$

with real numbers $y_1, y_2, \ldots, y_n$ all different from zero. On this vector space one can define the following bilinear form:

DEFINITION 48. Given the real numbers $z_0, z_1, \ldots, z_n$ which are, together with the $y_i$ all different from each other, and the weights $w_i$ which are different from zero.

Then one can define for two elements $\phi, \varphi$ of $\mathcal{V}_n$ the following inner product:

$$\langle \phi, \varphi \rangle = \sum_{i=0}^{n} w_i^2 \phi(z_i) \varphi(z_i).$$

The aim is to compute an orthonormal basis $\alpha_0(z), \alpha_1(z), \ldots, \alpha_n(z)$ such that

$$\alpha_j \in \mathcal{V}_j \backslash \mathcal{V}_{j-1} \quad (\mathcal{V}_{-1} = \emptyset), \quad \langle \alpha_i, \alpha_j \rangle = \delta_{i,j}.$$

If we define $w = (w_1, w_2, \ldots, w_n)^T$, $e_1 = (1, 0, \ldots, 0)^T$, $D_y = \operatorname{diag}(y_0, y_1, \ldots, y_n)$ with $y_0$ arbitrary and $D_z = \operatorname{diag}(z_0, z_1, \ldots, z_n)$, it is proved in [**175**] that if one solves an inverse eigenvalue problem of the following form:

$$\begin{aligned} Q^T w &= e_1 \|w\| \\ Q^T D_z Q &= S + D_y \end{aligned}$$

where $S$ is a symmetric generator representable semiseparable matrix, then the columns of the orthogonal $Q$ satisfy the following recurrence relation:

$$\Big(D_z - y_{i+1} I\Big) b_j Q_{j+1} = \Big(I - (D_z - y_j I) a_j\Big) Q_j - \Big(D_z - y_{j-1} I\Big) b_{j-1} Q_{j-1}$$

with

$$\begin{aligned} j &= 0, 1, \ldots, n-1 \\ Q_0 &= w / \|w\| \\ Q_{-1} &= 0 \end{aligned}$$

and the elements $a_i$ and $b_i$ are respectively the diagonal and subdiagonal elements of the matrix $T = S^{-1}$ which is a tridiagonal matrix.

Even more, if we define now the functions $\alpha_j(z)$ using the following recursion relation

$$\Big(z - y_{j+1}\Big) b_j \alpha_{j+1}(z) = \alpha_j(z) \Big(1 - (z - y_j) a_j\Big) - \alpha_{j-1}(z) \Big(z - y_{j-1}\Big) b_{j-1}$$

with $\alpha_0(z) = a / \sqrt{\sum w_i^2}$, $\alpha_{-1}(z) = 0$, and the $a_i$'s and $b_i$'s are respectively the diagonal and subdiagonal elements of the matrix $T = S^{-1}$, then we have the following relation between $Q_j$'s and the $\alpha_j$'s

$$Q_j = \operatorname{diag}(w) (\alpha_j(z_i))_{i=0}^{n}$$

and

$$\alpha_j \in \mathcal{V}_j \backslash \mathcal{V}_{j-1} \quad (\mathcal{V}_{-1} = \emptyset), \quad \langle \alpha_i, \alpha_j \rangle = \delta_{i,j}.$$

This means that we can calculate the desired orthonormal basis by solving an inverse eigenvalue problem for semiseparable matrices. Even though this is a very interesting topic, we will not go into the details of this. More information about more general problems, and how to solve this inverse eigenvalue problem, can be found in the papers mentioned in the beginning of this section.

## 5. An historical overview of the literature

In this section an historical overview of papers and books closely related to semiseparable matrices is given. For all the references a small summary of the results is included. Also different types of applications and extensions of the class of semiseparable matrices are mentioned. It can be seen, that interesting results are often rediscovered in different fields. Unfortunately we were not able to retrieve the following papers: [**194**] cited in [**97, 127**]; the papers cited in [**165**]; [**63**]; some citations of [**171**]; the papers [**12, 13**].

The name semiseparable matrix finds its origin in the discretization of kernels. A separable kernel $k(x, y)$ means that the kernel is of the following form $k(x, y) = g(x)f(y)$. Associating a matrix with this kernel gives a rank 1 matrix. This matrix can be written as the outer product of two vectors. A semiseparable kernel satisfies the following properties (sometimes also called a Green's kernel [**116**, p. 110]):

$$k(x, y) = \begin{cases} g(x)f(y) & \text{if} \quad x \le y \\ g(y)f(x) & \text{if} \quad y \le x \end{cases} .$$

Associating a matrix with this semiseparable kernel, gives us a generator representable semiseparable matrix.

1937 [**82**] In this paper of Gantmacher and Krein, they proved that the inverse of a symmetric Jacobi matrix (this corresponds to a symmetric irreducible tridiagonal matrix) is a one-pair matrix (this corresponds to a symmetric generator representable semiseparable matrix) via explicit calculations.

1950 [**83**] In the book of Gantmacher and Krein, the theory of [**82**] was included. This book is often referred to as the first book in which the inverse of an irreducible tridiagonal matrix was calculated.

1953 [**11**] Berger and Saibel provide in this paper an explicit formula for calculating the inverse of a continuant matrix (this is a tridiagonal matrix, not necessarily symmetric), based on the $LU$-decomposition of the original continuant matrix. Moreover, no conditions of nonzeroness are placed on the elements of the continuant matrix.

1956 [**149**] In this paper the authors Roy and Sarhan invert very specific matrices arising in statistical applications, e.g., a lower triangular semiseparable matrix, 2 types of specific semiseparable matrices and also a semiseparable plus diagonal matrix are inverted.

1957 [**4**] In this paper, S. O. Asplund, the father of E. Asplund proves the same as Gantmacher and Krein, by calculating the inverse via techniques for solving finite boundary value problems. A brief remark states that higher order band matrices have as inverses higher order Green's matrices (The Green's matrix can be considered as a generator representable semiseparable matrix).

1957 [**3**] E. Asplund formulates for the first time a theorem stating that the inverse of an invertible lower $\{p\}$-Hessenberg matrix (with no restrictions on the elements and called a $\{p\}$-band matrix in the paper) is an invertible lower $\{p\}$-Hessenberg-like matrix (with the definition of the ranks of subblocks) and vice versa. A $\{p\}$-Hessenberg-like matrix is referred to as a Green's matrix of order $p$ in the paper. No explicit formulas for the inverse are presented, only theoretical results about the structure. More precisely, he also proves, then if the elements on the $p$th superdiagonal of the lower $\{p\}$-Hessenberg matrix are different from zero, that the inverse is a generator representable lower $\{p\}$-Hessenberg-like matrix.

1959 [**97**] In this paper by Greenberg and Sarhan, the papers [**194, 149**] are generalized and applied to several types of matrices arising in statistical applications. A relation is introduced which needs to be satisfied, such that the inverse of the matrix is a diagonal matrix of type $r$ (these diagonal matrices of type $r$ correspond to band matrices of width $r$). Different semiseparable matrices are given and the relation is investigated for $r$ equal to $1, 2$ and 3, thereby proving that these matrices have a banded inverse.

1960 [**148**] As the statistical research at that time was interested in fast calculations for so-called patterned matrices, the authors Roy, Greenberg and Sarhan designed an order $n$ algorithm for calculating the determinant. The patterned matrices are semiseparable and semiseparable plus diagonal matrices.

1961 [**158**] Schechter provides a method for inverting nonsymmetric block tridiagonal matrices for blocks which have the same size and are invertible, based on the $LDU$-decomposition of these matrices. The applications of this method can be found in the field of partial differential equations.

1964 [**166**] Ting describes, based on the $LU$ decomposition of a tridiagonal matrix $T$ (not necessarily symmetric) a special decomposition of $T^{-1}$ as a sum of inverses of bidiagonal matrices, multiplied with a diagonal matrix. This can be used for solving various systems of equations with the same tridiagonal matrix $T$.

1965 [**167**] Published in a Japanese Journal, and translated afterwards to [**168**].

1966 [**168**] Torii, provides explicit formulas for calculating the inverse of a nonsymmetric tridiagonal matrix. The stability of solving a tridiagonal system of equations via inversion is investigated.

1967 [**32**] The author Chandan inverts a symmetric semiseparable plus identity matrix, and concludes that it is again a symmetric semiseparable matrix plus the identity. This matrix results from a statistical problem.

1968 [**123**] The author Kounias provides explicit formulas for inverting some patterned matrices. For example, a tridiagonal Toeplitz matrix and a matrix completely filled with one element plus a diagonal are inverted.

1969 [**172**] Uppuluri and Carpenter present an exact formula to compute the inverse of a specific covariance matrix, which is a symmetric tridiagonal Toeplitz matrix.

1970 [**2**] Allgower provides a method for calculating the inverse of banded Toeplitz matrices.

1970 [**120**] Kershaw provides bounds between which the elements of the inverse of a tridiagonal matrix with positive off-diagonal elements will lie. The elements of the super and the subdiagonal of the tridiagonal are related in the following way: denote with $T$ the tridiagonal matrix, then we have: $T_{i,i+1} = 1 - T_{i-1,i}$ for $1 < i < n$.

1970 [**173**] Uppuluri and Carpenter calculate the inverse of a tridiagonal Toeplitz matrix (not necessarily symmetric) via the associated difference equation.The results are based on [**123**].

1970 [**30**] Explicit formulas for the inverse of a nonsymmetric tridiagonal matrix are given by Capovani (no demands are placed on the elements of the tridiagonal). The formulas are adapted for the tridiagonal Toeplitz case and the formulas are applicable to block tridiagonal matrices (for which all the blocks have the same size) as well.

1971 [**31**] Capovani proves that the inverse of a nonsymmetric tridiagonal matrix with the super and subdiagonal elements different from zero, also has as inverse a generator representable matrix, where the upper and lower triangular part have different generators. Using these results the author proves that a pentadiagonal matrix can be written as the product of two tridiagonal matrices, for which one of them is symmetric.

1971 [**6**] Baranger and Duc-Jacquet prove that the inverse of a generator representable semiseparable matrix (called "une matrice factorisable" in the paper) is a tridiagonal matrix. They explicitely calculate the inverse of a generator representable semiseparable matrix.

1972 [**110**] Hoskins and Ponzo provide formulas for calculating the inverse of a specific band matrix of dimension $2r + 1$, with the binomial coefficients in the expansion of $(x - 1)^{2r}$ in each row and column. Such a matrix arises for example in the solution of partial difference equations.

1972 [**143**] Rehnqvist presents here a method for inverting a very specific Toeplitz band matrix. Via multiplication with another invertible Toeplitz matrix and reordering of the elements one can calculate the inverse of this matrix.

1972 [**165**] D. Szynal and J. Szynal present two theorems for the existence of the inverse of a Jacobi matrix and two methods for calculating the inverse. The results can be rewritten in block form such that they can be applied to specific band matrices. No specific demands of symmetry or nonzeroness of the elements have to be fulfilled.

1973 [**111**] Hoskins and Thurgur invert a specific type of band matrix (see [**110**]). The inversion formulas are obtained by calculating the $LU$-decomposition of the band matrix and formulas are given to compute the determinant and the infinity norm of this inverse.

1973 [**122**] Kounadis deduces in this paper a recursive formula for inverting symmetric block tridiagonal matrices, for which all the blocks are of the same dimension. This technique is also applied to the class of tridiagonal matrices.

1973 [**24**] Bukhberger and Emel'yanenko present in this paper a computational method for inverting symmetric tridiagonal matrices, for which all the elements are different from zero. The theoretical results are applicable in a physical application which studies the motion of charged particles in a particular environment. The results are identical to the ones in the book [**83**].

1974 [**170**] Trench presents a method for inverting $\{p, q\}$-banded Toeplitz matrices by exploiting the banded structure.

1976 [**14**] Bevilacqua and Capovani extend the results of the papers [**97**] and [**1, 2**] to band matrices and to block band matrices (not necessarily symmetric). Formulas are presented for inverting band matrices whose elements on the extreme diagonals are different from zero. The results are extended to block band matrices.

1976 [**107**] The summary of the paper, as provided by the authors Hoskins and McMaster : For a symmetric positive definite Toeplitz matrix of band width five and order $n$, those regions where the elements of the inverse alternate in sign are determined.

1977 [**108**] Hoskins and McMaster investigate properties of the inverse of banded Toeplitz matrices as in [**107**]. More precisely, they investigate the properties of a band matrix of width 4 and a band matrix of width 5 coming from a boundary value problem.

1977 [**10**] Berg provides in this paper for the first time an explicit technique for inverting a band matrix that is not symmetric, and has for the upper and lower part different bandwidths. We could already assume these results based on the paper by Asplund, but here explicit formulas are given. A disadvantage is again the strong assumption that the elements on the extreme super and subdiagonals have to be different from zero.

1977 [**43**] Demko proves theorems bounding the size of the elements of the inverse of band matrices in terms of the norm of the original matrix, the distance towards the diagonal and the bandwidth. In particular it is shown that the size of the elements decays exponentially to zero if one goes further and further away from the diagonal.

1977 [**138**] Neuman provides algorithms for inverting tri and pentadiagonal matrices (not necessarily symmetric) via the $UL$ decomposition. Under some additional conditions the numerical stability of the algorithm is proved.

1977 [**174**] Valvi determines several explicit formulas for inverting specific patterned matrices arising in statistical applications. These matrices are specific types of semiseparable and semiseparable plus diagonal matrices.

1978 [**109**] Bounds on the infinity norm of the inverse of Toeplitz band matrices with band width 5 are derived by Hoskins and McMaster.

1978 [**99**] Greville provides conditions on band matrices such that the inverse of the band matrix becomes a Toeplitz matrix. This results in Toeplitz matrices which are also semiseparable.

1978 [**139**] Oohashi proves that the elements of the inverse of a band matrix can be expressed in terms of the solution of a homogeneous difference equation, related to the original band matrix. In this way explicit formulas for calculating the inverse are obtained. The band matrix does not need to have the same lower and upper band size, but the elements on the extreme diagonals need to be different from zero. The results are an extension of the results proved in [**168**] for tridiagonal matrices.

1978 [**8**] Barrett and Feinsilver provide a probabilistic proof, stating that the inverse of a symmetric generator representable semiseparable matrix, is a symmetric tridiagonal matrix. Note that the generator representable matrices are not characterized by two generators, but via the rank 1 assumptions in the lower and upper triangular part. The results are restricted to positive definite and symmetric matrices.

1979 [**113**] Ikebe provides in this paper an algorithm for inverting an upper Hessenberg matrix under the assumption that the subdiagonal elements are different from zero. It is proved that the inverse matrix has the lower triangular part representable with two generators. An extension towards block Hessenberg matrices is included.

1979 [**191**] Yamammoto and Ikebe propose formulas for inverting band matrices with different bandwidths under the assumption that the elements on the extreme diagonals are different from zero.

1979 [**7**] Barrett formulates another type of theorem connected to the inverses of tridiagonal matrices. In most of the preceding papers one assumed the sub and superdiagonal elements of the corresponding tridiagonal matrix to be different from zero. In this paper only one condition is left, it assumes that the diagonal elements of the symmetric semiseparable matrix are different from zero. Moreover, the proof is also suitable for nonsymmetric matrices. The theorems presented in this paper are very close to the final version result, stating that the inverse of a tridiagonal matrix is a semiseparable matrix, satisfying the rank definition.

1979 [**160**] Singh gives explicit formulas for inverting a lower block bidiagonal matrix. The blocks on the diagonal have to be invertible.

1980 [**105**] The author Haley splits the band system $Bx = u$ into two blocks, which can via recurrence divided once more. This leads to a method for calculating explicit inverses of banded matrices. As example tridiagonal and Toeplitz tridiagonal matrices are considered.

1981 [**9**] This paper should be considered as one of the most important papers concerning the inverses of band matrices. Barrett and Feinsilver provide a general framework as presented in the first chapter of this thesis. General theorems and proofs considering the vanishing of minors when looking at the matrices and their inverses are given, thereby characterizing the complete class of band and semiseparable matrices, without excluding cases in which there appear zeros. The results are a straightforward consequence of paper [**7**].

1982 [**127**] Lewis provides an explicit formula which can be used to compute the inverse of tridiagonal matrices. The matrix does not necessarily need to be symmetric, nor all elements have to be different from zero. Interesting is also a new kind of representation for nonsymmetric semiseparable matrices, which does not use 4 but three vectors $x$,$y$ and $z$, where the elements of $S$ are of the following form:

$$s_{ij} = \begin{cases} x_i y_j z_j, & i \geq j, \\ y_i x_j z_j, & i \leq j. \end{cases}$$

The explicit formula is a generalization of the theorem for symmetric matrices by Yoshimasa [**194**].

1984 [**44**] Decay rates for the inverse of band matrices are obtained by Demko, Moss and Smith. It is shown that the decay rate depends on the spectrum of the matrix $AA^T$, for a band matrix $A$.

1984 [**145**] Rizvi derives, based on the formulas proposed by Barrett and Feinsilver and on the $LU$-decomposition, formulas for inverting a quasi-tridiagonal matrix, which is in fact a block tridiagonal matrix (for which all the blocks have the same size). Necessary and sufficient conditions, for which a special block matrix has as inverse a quasi-tridiagonal matrix are derived.

1985 [**89**] Gohberg, Kailath and Koltracht provide a method for solving higher order semiseparable plus diagonal systems of equations. The class of matrices needs to be the strongly regular.

1986 [**131**] Mattheij and Smooke provide in this paper a method for deriving the explicit inverse of tridiagonal matrices, as coming from nonlinear boundary value problems. The formulas are also suitable for the block tridiagonal case.

1986 [**147**] Romani investigates the demands on a symmetric band matrix, such that its inverse can be written as the sum of inverses of irreducible tridiagonal matrices, because this is not true in general.

1986 [**150**] Rózsa investigates sufficient conditions to be placed on a semiseparable matrix, in order to have a nonsingular $\{p, q\}$-semiseparable matrix. This paper is to our knowledge the first paper in which there appeared the now well-known name of semiseparable matrices. A theorem is included stating that the inverse of a strict $\{p, q\}$-band matrix is a generator representable $\{p, q\}$-semiseparable matrix.

1986 [**73**] In this paper, Fiedler translates the abstract formulation in [**102**] towards the matrix case. The resulting theorem is called the nullity theorem in this thesis. This theorem is very useful in the field of semiseparable and tridiagonal matrices as shown in Chapter 1 of this thesis.

1986 [**29**] A generalization of the papers [**113, 191**] is presented by Cao and Stewart, for Hessenberg matrices with a larger bandwidth and block Hessenberg matrices, for which the blocks do not necessary need to have the same dimension. The implementation described is suitable for parallel computations.

1987 [**151**] Rózsa proves, based on linear difference equations, the same results as in [**150**]. A small section is dedictated to band Toeplitz matrices.

1987 [**74**] In this paper Fiedler proves that the off-diagonal rank of a matrix is maintained under inversion.

1988 [**15**] Bevilacqua, Codenotti and Romani present a method to solve a block tridiagonal system in a parallel way, by exploiting the structure of the inverse of the block tridiagonal matrix.

1988 [**60**] Eijkhout and Polman provide decay rates for the inverse of band matrices which are close to Toeplitz matrices.

1989 [**152**] The authors Rózsa, Bevilacqua, Favati and Romani, present generalizations of their previous papers towards methods for computing the inverse of block tridiagonal and block band matrices. The results are very general and the blocks do not all need to be square or of the same dimension. The paper contains a lot of interesting references connected to the theory of semiseparable and tridiagonal matrices.

1990 [**17**] The authors Bevilacqua, Lotti and Romani, present two types of algorithms for reducing the total amount of storage locations needed by the inverse of a band matrix.

1990 [**18**] The authors Bevilacqua, Romani and Lotti provide a method for parallel inversion of band matrices. No other assumption is required than the nonsingularity of some of the principal submatrices.

1991 [**70**] Favati, Lotti, Romani and Rózsa provide theorems and formulas for the inverse of generalized block band matrices. The paper covers many different cases in a very general framework.

1991 [**153**] A new proof is included by Rózsa, Bevilacqua, Romani and Favati stating that the inverse of a $\{p,q\}$-semiseparable matrix is a strict $\{p,q\}$-band matrix. This new proof leads to a recursive scheme for calculating the inverse.

1992 [**133**] In this paper by Meurant many references connected to the inverse of semiseparable and band matrices are included. Some results concerning the inverse of symmetric tridiagonal and block tridiagonal matrices are reviewed, based on the Cholesky decomposition. Also results concerning the decay of the elements of the inverse are obtained.

1993 [**71**] Fiedler formulates general theorems connected to the structured ranks of matrices and their inverses.

1993 [**23**] The authors Brualdi and Massey generalize some of the results of [**71**], for structures in which the diagonal is not included.

1994 [**154**] Rózsa, Romani and Bevilacqua provide results similar to the ones in [**70**]. But the results are now proved via the nullity theorem, which is also proved in this paper.

1996 [**53**] Eidelman and Gohberg present an order $O(n)$ algorithm for calculating the inverse of a generator representable plus diagonal semiseparable matrix.

1997 [**55**] Eidelman and Gohberg present a look ahead recursive algorithm to compute the triangular factorization of generator representable semiseparable matrices plus a diagonal.

1997 [**54**] Eidelman and Gohberg present a fast and numerically stable algorithm for inverting semiseparable plus diagonal matrices. No conditions, except nonsingularity, are placed on the matrices.

1998 [**132**] McDonald, Nabben, Neumann, Schneider and Tsatsomeros pose properties on the class of generator semiseparable matrices such that their tridiagonal inverses belong to the class of $Z$-matrices.

1999 [**61**] Elsner investigates in more detail the inverses of band and Hessenberg matrices. The different cases, concerning generator representable semiseparable matrices and semiseparable matrices for which the special subblocks have low rank are included.

1997 [**121**] Koltracht provides in this paper a new method for solving higher order semiseparable systems of equations. The semiseparable matrix is thereby transformed into a narrow band matrix.

1999 [**137**] Nabben provides upper and lower bounds for the entries of the inverse of diagonally dominant tridiagonal matrices.

1999 [**136**] Decay rates for the inverse of special tridiagonal and band matrices are given by Nabben.

2000 [**35**] Chandrasekaran and Gu present an algorithm to transform a generator representable semiseparable matrix plus band matrix into a similar tridiagonal matrix.

2000 [**171**] Tyrtyshnikov expands the class of generator representable matrices towards a class called weakly semiseparable matrices. He proves that the inverse of a $(p,q)$-band matrix is a $(p,q)$-weakly semiseparable matrix, but the converse does not necessarily holds. This means that the class of $(p,q)$-weakly semiseparable matrices is a little more general than our class of $(p,q)$-semiseparable matrices.

2001 [**66**] This paper by Fasino and Gemignani provides for semiseparable matrices, also singular ones, a sparse structured representation. More precisely, it is shown that a semiseparable matrix $A$, always can be written as the inverse of a block tridiagonal matrix plus a sparse, low rank matrix $Z$.

2001 [**68**] Fasino and Gemignani, study in this paper the direct and the inverse eigenvalue problem of diagonal plus semiseparable matrices.

2001 [**128**] Mastronardi, Chandrasekaran and Van Huffel present an order $O(n)$ algorithm to solve a system of equations where the coefficient matrix is a generator representable semiseparable plus diagonal matrix. The algorithm is suitable for an implementation on two processors.

2002 [**178**] Van Camp, Van Barel and Mastronardi provide two fast algorithms for solving diagonal plus generator representable semiseparable systems of equations. The solution method consists of an effective calculation of the $QR$-factorization of this type of matrices.

2002 [**175**] In this paper, Van Barel, Fasino, Gemignani and Mastronardi investigate the relation between orthogonal rational functions and generator representable semiseparable plus diagonal matrices.

2002 [**135**] Mullhaupt and Riedel derive properties for matrices having weakly lower triangular rank equal to $d$. Decompositions of these matrices as a product of a unitary matrix and an upper $\{p\}$-Hessenberg matrix are provided.

2002 [**85**] Fasino and Gemignani describe an order $O(n)$ solver for banded plus semiseparable systems of equations. The algorithm exploits the structure of the inverse of the semiseparable matrix.

2003 [**37**] Chandrasekaran and Gu present a method for solving systems whose coefficient matrix is a semiseparable plus band matrix.

2003 [**38**] Chandrasekaran and Gu present a divide and conquer method to calculate the eigendecomposition of a symmetric generator representable semiseparable plus a block diagonal matrix.

2003 [**129**] Mastronardi, Chandrasekaran and Van Huffel, provide an algorithm to transform a symmetric generator representable semiseparable plus diagonal matrix into a similar tridiagonal one. Also a second algorithm to reduce an unsymmetric generator representable semiseparable plus diagonal matrix to a bidiagonal one by means of orthogonal transformations is included.

2003 [**130**] Mastronardi, Van Camp and Van Barel present a divide and conquer algorithm to compute the eigendecomposition of diagonal plus generator representable semiseparable matrices.

2003 [**69**] Fasino, Mastronardi and Van Barel propose two new algorithms for transforming diagonal plus generator representable semiseparable matrices to tridiagonal or bidiagonal form. See also [**129**].

2003 [**72**] Fiedler investigates the properties of so-called basic matrices, these matrices have weakly lower and weakly upper triangular rank 1, in fact this class is closely related to the matrices.

2003 [**19**] Bini, Gemignani and Pan derive an algorithm for performing a step of $QR$ on a generalized semiseparable matrix, the lower part of this generalized matrix is in fact lower triangular semiseparable, the authors present an alternative representation, consisting of 3 vectors, to represent this part.

2003 [**16**] Bevilacqua and Del Corso investigate in this report the existence and the uniqueness of a unitary similarity transformation of a symmetric matrix into semiseparable form. Also the implementation of a $QR$-step on a semiseparable matrix without shift is investigated.

2004 [**164**] Strang and Nguyen investigate in this paper the nullity theorem as provided by Fiedler and Markham, and also the theoretical results from Barrett and Feinsilver.

2004 [**177**] Van Barel, Van Camp and Mastronardi provide an orthogonal similarity transformation to reduce symmetric matrices into a similar semiseparable one of rank $k$.

2004 [**77**] Fiedler investigates the properties of a class called Generalized Hessenberg matrices, these matrices have weakly lower triangular rank 1.

2004 [**65**] Fasino proves that any Hermitian matrix, with pairwise distinct eigenvalues can be transformed into a similar diagonal plus semiseparable matrix, with a prescribed diagonal. Moreover, it is proved that the unitary transformation in this similarity reduction is the unitary matrix in the $QR$-factorization of a suitably defined Krylov subspace.

In the remaining part of this section we will briefly mention some new classes of matrices closely related to semiseparable matrices. Currently the class of sequentially semiseparable matrices is investigated thoroughly. (This class is sometimes also called quasiseparable, or recursively semiseparable.) Interesting algorithms and the definition of this class of matrices can be found in [**33, 34, 36, 48, 49, 52, 56, 58**]. Recently a new class of matrices, $\mathcal{H}$-matrices is introduced. This class is also closely related to the class of semiseparable and sequentially semiseparable matrices (see [**21, 103, 104**] and the references therein).

Semiseparable matrices also appear in various types of applications, e.g., the field of integral equations [**92, 100, 115**], operator theory [**57, 59, 101**], boundary value problems [**92, 98, 126, 161**], in the theory of Gauss-Markov processes [**117**], time varying linear systems [**48, 88**], in statistics [**95**], acoustic and electromagnetic scattering theory [**39**], signal processing [**134, 135**], numerical integration and differentiation [**161**] and rational interpolation [**175**]. Also in the biological field applications exist resulting directly in semiseparable matrices [**62**].

Our contribution in this field of semiseparable matrices consists of the development of a stable representation for the class of semiseparable matrices as we defined them. Also the reduction algorithms as they will proposed in Part 2 and the implicit $QR$-algorithms for semiseparable matrices as proposed in Part 3 are new.

## Conclusions

In this chapter we showed that the interest in the class of semiseparable matrices is not only of theoretical nature. In various fields semiseparable matrices are used. We briefly mentioned four different fields in which these matrices arise. We pointed out that semiseparable matrices, under some assumptions, can be seen as oscillation matrices. Some covariance matrices are also semiseparable or semiseparable plus diagonal matrices. Finally we indicated the role these matrices play in the field of integral equations and orthogonal rational functions.

In the last section we provided an extensive historical overview of publications closely related to semiseparable matrices. This overview covers theoretical aspects as well as applications related to this class of matrices.

# Part 2

# The reduction of matrices to semiseparable matrices

CHAPTER 4

# Different algorithms for reducing matrices to semiseparable form

It is known from Part 1 that semiseparable and tridiagonal matrices are closely related to each other. Invertible semiseparable matrices have as inverse a tridiagonal matrix and vice versa. One might wonder if this close relation between these two classes of matrices can be extended. For example, one can reduce any symmetric matrix via orthogonal similarity transformations into a similar tridiagonal one. In this chapter we will construct algorithms for reducing matrices to semiseparable form.

In the first section an algorithm will be constructed to reduce any symmetric matrix into a similar semiseparable one by means of orthogonal transformations. As one will see this algorithm is closely related to the tridiagonalization of a symmetric matrix. In fact a sequence of extra Givens transformations is performed to create the semiseparable structure.

The second section focusses on applying this transformation on non symmetric matrices. It will transform the matrix into a similar Hessenberg-like matrix.

In the third section we do not work with similarity transformations anymore. In correspondence with the reduction of an arbitrary matrix by orthogonal transformations to a bidiagonal one, we construct an algorithm which reduces a matrix into an upper triangular semiseparable one.

## 1. An orthogonal similarity reduction of a symmetric matrix to a symmetric semiseparable one

An algorithm to transform a symmetric matrix into a semiseparable one by orthogonal similarity transformations is presented in this section. The constructive proof of the next theorem, provides the algorithm.

Theorem 49. *Let $A$ be a symmetric matrix. Then there exists an orthogonal matrix $U$ such that*

$$U^T AU = S,$$

*where $S$ is a semiseparable matrix.*

Proof. The constructive proof made by induction on the rows of the matrix $A$. Let $A_0^{(1)} = A$. We will often, briefly denote $A_0^{(i)}$ as $A^{(i)}$. Let $G_i^{(l)}$ be a Givens transformation, such that the product $A_{i-1}^{(l)} G_i^{(l)}$ has the entry $(n-l+1, i)$ annihilated and the $i$th and the $(i+1)$th columns of $A_{i-1}^{(l)}$ modified.

Step 1 We start by constructing a similarity transformation which makes the last two rows (columns) linearly dependent in the lower (upper) triangular part. To this end, we multiply $A_0^{(1)}$ to the left by $G_1^{(1)^T}$ and to the right by $G_1^{(1)}$ to annihilate the elements in position $(1,n)$ and $(n,1)$ in $A_0^{(1)}$, respectively. The arrows denote the columns and rows which will be affected by transforming the matrix $A_0^{(1)}$ into $G_1^{(1)^T} A_0^{(1)} G_1^{(1)}$:

$$\begin{array}{cc} & \begin{array}{ccccc} \downarrow & \downarrow & & & \end{array} \\ \begin{array}{c} \rightarrow \\ \rightarrow \\ \\ \\ \\ \end{array} & \begin{pmatrix} \times & \times & \cdots & \times & \otimes \\ \times & \ddots & & \times & \times \\ \vdots & & \ddots & \vdots & \vdots \\ \times & \times & \cdots & \times & \times \\ \otimes & \times & \cdots & \times & \times \end{pmatrix} \end{array} \xrightarrow{G_1^{(1)^T} A_0^{(1)} G_1^{(1)}} \begin{pmatrix} \times & \times & \cdots & \times & 0 \\ \times & \ddots & & \times & \times \\ \vdots & & \ddots & \vdots & \vdots \\ \times & \times & \cdots & \times & \times \\ 0 & \times & \cdots & \times & \times \end{pmatrix}$$

$$\Updownarrow$$

$$A_0^{(1)} \quad \xrightarrow{G_1^{(1)^T} A_0^{(1)} G_1^{(1)}} \quad A_1^{(1)}.$$

Note that $\times$ denotes arbitrary elements of the matrix, while $\otimes$ denotes the elements which will be annihilated by the orthogonal similarity transformation. Summarizing, we obtain that $A_1^{(1)} = G_1^{(1)^T} A_0^{(1)} G_1^{(1)}$. Continuing this process of annihilating all the elements in the last row (column), except for the element in position $(n, n-1)$ $((n-1, n))$, we get

$$A_{n-2}^{(1)} = \begin{pmatrix} \times & \times & \cdots & \times & 0 \\ \times & \ddots & & \vdots & \vdots \\ \vdots & & \ddots & \times & 0 \\ \times & \cdots & \times & \times & \times \\ 0 & \cdots & 0 & \times & \times \end{pmatrix}.$$

Multiplying $A_{n-2}^{(1)}$ to the left by $G_{n-1}^{(1)^T}$, we have the following situation,

$$\begin{array}{cc} \begin{array}{c} \\ \\ \\ \rightarrow \\ \rightarrow \end{array} & \begin{pmatrix} \times & \times & \cdots & \times & 0 \\ \times & \ddots & & \vdots & \vdots \\ \vdots & & \ddots & \times & 0 \\ \times & \cdots & \times & \times & \otimes \\ 0 & \cdots & 0 & \times & \times \end{pmatrix} \end{array} \xrightarrow{G_{n-1}^{(1)^T} A_{n-2}^{(1)}} \begin{pmatrix} \times & \times & \cdots & \times & 0 \\ \times & \ddots & & \vdots & \vdots \\ \vdots & & \ddots & \times & 0 \\ \boxtimes & \cdots & \boxtimes & \times & 0 \\ \boxtimes & \cdots & \boxtimes & \times & \times \end{pmatrix}$$

$$\Updownarrow$$

$$A_{n-2}^{(1)} \quad \xrightarrow{G_{n-1}^{(1)^T} A_{n-2}^{(1)}} \quad \tilde{A}_{n-2}^{(1)},$$

i.e., the last two rows are proportional with exception of the entries in the last two columns, (to emphasize the linear dependency among the rows (columns) we denote by $\boxtimes$ the elements of the matrix belonging to these rows (columns)). Let $\tilde{A}^{(1)}_{n-2} = {G^{(1)}_{n-1}}^T A^{(1)}_{n-2}$. Multiplying now $\tilde{A}^{(1)}_{n-2}$ to the right by $G^{(1)}_{n-1}$, the last two columns become linearly dependent above and on the main diagonal, i.e., the last two columns form an upper semiseparable part. Because of symmetry, the last two rows become linearly dependent below and on the main diagonal and form a lower semiseparable part:

$$\begin{array}{c} \begin{array}{ccccc} & & & \downarrow & \downarrow \end{array} \\ \left(\begin{array}{ccccc} \times & \times & \cdots & \times & 0 \\ \times & \ddots & & \vdots & \vdots \\ \vdots & & \ddots & \times & 0 \\ \boxtimes & \cdots & \boxtimes & \times & 0 \\ \boxtimes & \cdots & \boxtimes & \times & \times \end{array}\right) \end{array} \xrightarrow{\tilde{A}^{(1)}_{n-2}G^{(1)}_{n-1}} \left(\begin{array}{ccccc} \times & \times & \cdots & \boxtimes & \boxtimes \\ \times & \ddots & & \vdots & \vdots \\ \vdots & & \ddots & \boxtimes & \boxtimes \\ \boxtimes & \cdots & \boxtimes & \boxtimes & \boxtimes \\ \boxtimes & \cdots & \boxtimes & \boxtimes & \boxtimes \end{array}\right)$$

$$\Updownarrow$$

$$\tilde{A}^{(1)}_{n-2} \xrightarrow{\tilde{A}^{(1)}_{n-2}G^{(1)}_{n-1}} A^{(1)}_{n-1}.$$

To start the next step, $A^{(2)}_0$ is defined as $A^{(2)}_0 = A^{(1)}_{n-1}$.

Step $k$ Let $k = n - j,\ 1 < j < n$. Assume by induction that $A^{(k)}_0$ has the lower (upper) triangular part already semiseparable from row $n$ up to row $j+1$ (column $j+1$ to $n$). We will now prove that we can make the lower (upper) triangular part semiseparable up to row $j$ (column $j$). Let us denote the lower (and upper)triangular elements which form a semiseparable part with $\boxtimes$. Matrix $A^{(k)}_0$ looks like:

$$A^{(k)}_0 = \left(\begin{array}{cccccc} \times & \cdots & \times & \boxtimes & \cdots & \boxtimes \\ \vdots & \ddots & \vdots & \vdots & & \vdots \\ \times & \cdots & \times & \boxtimes & \cdots & \boxtimes \\ \boxtimes & \cdots & \boxtimes & \boxtimes & \cdots & \boxtimes \\ \vdots & & \vdots & \vdots & \ddots & \vdots \\ \boxtimes & \cdots & \boxtimes & \boxtimes & \cdots & \boxtimes \end{array}\right) \begin{array}{l} \leftarrow\ 1 \\ \vdots \\ \leftarrow\ j \\ \leftarrow j+1 \\ \vdots \\ \leftarrow\ n \end{array}.$$

We remark that the lower right $k \times k$ block of $A^{(k)}_0$ is already semiseparable. Because the submatrix of $A^{(k)}_0$ consisting of the first $j$ columns and the last $n-j$ rows is already semiseparable, it has rank $\leq 1$. Hence, we can construct a Givens transformation $G^{(k)}_1$ such that multiplying $A^{(k)}_0$ to the left by ${G^{(k)}_1}^T$ will annihilate all elements $(1, j+1)$, $(1, j+2)$ up to $(1, n)$. In a similar way, we can construct the Givens transformations $G^{(k)}_2, \ldots, G^{(k)}_{j-2}$ and $G^{(k)}_{j-1}$ such that multiplying $A^{(k)}_1$ on the left with ${G^{(k)}_{j-1}}^T \ldots {G^{(k)}_2}^T$, will make zero elements in rows 2 up to $j-1$ the columns $j+1$ up to $n$.

$$
\begin{array}{r}
\\ \\ \\ j \rightarrow \\ j+1 \rightarrow \\ \\ \\ \\
\end{array}
\left(\begin{array}{cccccccc}
\times & \cdots & \times & \times & 0 & 0 & \cdots & 0 \\
\vdots & \ddots & \vdots & \vdots & \vdots & \vdots & & \vdots \\
\times & \cdots & \times & \times & 0 & 0 & \cdots & 0 \\
\times & \cdots & \times & \times & \otimes & \otimes & \cdots & \otimes \\
0 & \cdots & 0 & \boxtimes & \boxtimes & \boxtimes & \cdots & \boxtimes \\
0 & \cdots & 0 & \boxtimes & \boxtimes & \boxtimes & \cdots & \boxtimes \\
\vdots & & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & \cdots & 0 & \boxtimes & \boxtimes & \boxtimes & \cdots & \boxtimes
\end{array}\right)
\xrightarrow{G_j^{(k)^T} A_{j-1}^{(k)}}
\begin{array}{c}
\begin{array}{cccccccc}
 & & & j & j+1 & & & \\
 & & & \downarrow & \downarrow & & &
\end{array} \\
\left(\begin{array}{cccccccc}
\times & \cdots & \times & \times & 0 & 0 & \cdots & 0 \\
\vdots & \ddots & \vdots & \vdots & \vdots & \vdots & & \vdots \\
\times & \cdots & \times & \times & 0 & 0 & \cdots & 0 \\
\boxtimes & \cdots & \boxtimes & \times & 0 & 0 & \cdots & 0 \\
\boxtimes & \cdots & \boxtimes & \boxtimes & \boxtimes & \boxtimes & \cdots & \boxtimes \\
0 & \cdots & 0 & \otimes & \boxtimes & \boxtimes & \cdots & \boxtimes \\
\vdots & & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & \cdots & 0 & \otimes & \boxtimes & \boxtimes & \cdots & \boxtimes
\end{array}\right)
\end{array}
$$

$$
\xrightarrow{G_j^{(k)^T} A_{j-1}^{(k)} G_j^{(k)}}
\left(\begin{array}{cccccccc}
\times & \cdots & \times & \boxtimes & \boxtimes & 0 & \cdots & 0 \\
\vdots & \ddots & \vdots & \vdots & \vdots & \vdots & & \vdots \\
\times & \cdots & \times & \boxtimes & \boxtimes & 0 & \cdots & 0 \\
\boxtimes & \cdots & \boxtimes & \boxtimes & \boxtimes & 0 & \cdots & 0 \\
\boxtimes & \cdots & \boxtimes & \boxtimes & \boxtimes & \boxtimes & \cdots & \boxtimes \\
0 & \cdots & 0 & 0 & \boxtimes & \boxtimes & \cdots & \boxtimes \\
\vdots & & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & \cdots & 0 & 0 & \boxtimes & \boxtimes & \cdots & \boxtimes
\end{array}\right).
$$

Figure 4.1: The transformation $A_j^{(k)} = G_j^{(k)^T} A_{j-1}^{(k)} G_j^{(k)}$ makes the first $j$ entries of rows (columns) $j$ and $j+1$ proportional.

Applying these similarity transformations, we obtain the following matrix

$$
\begin{aligned}
A_{j-1}^{(k)} &= G_{j-1}^{(k)^T} \cdots G_1^{(k)^T} A_0^{(k)} G_1^{(k)} \cdots G_{j-1}^{(k)} \\
&= \left(\begin{array}{ccccccc}
\times & \cdots & & \times & 0 & \cdots & 0 \\
\vdots & \ddots & & \vdots & \vdots & & \vdots \\
 & & \ddots & \times & 0 & \cdots & 0 \\
\times & \cdots & \times & \times & \boxtimes & \cdots & \boxtimes \\
0 & \cdots & 0 & \boxtimes & \boxtimes & \cdots & \boxtimes \\
\vdots & & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & \cdots & 0 & \boxtimes & \boxtimes & \cdots & \boxtimes
\end{array}\right)
\begin{array}{l}
\leftarrow \ 1 \\ \vdots \\ \vdots \\ \leftarrow \ j \\ \leftarrow j+1 \\ \vdots \\ \leftarrow \ n
\end{array}.
\end{aligned}
$$

Since the rows $j$ and $j+1$ are proportional for the indexes of columns greater than $j$, also the Givens transformation $G_j^{(k)^T}$ can be constructed, annihilating all the entries in row $j$ with column index greater than $j$. Furthermore, the product of $G_j^{(k)^T} A_{j-1}^{(k)}$ to the right by $G_j^{(k)}$ makes the columns $j$ and $j+1$ proportional in the first $j$ entries. The latter similarity transformation is depicted in Figure 4.1.

To retrieve the semiseparable structure, below the $(j+1)$th row and to the right of the $(j+1)$th column, $n-j-1$ more similarity Givens transformations $G_k$, $k = j+1, j+2, \ldots, n-1$ are considered. Each of these is chosen in such a way that when the Givens transformation $G_k^T$ is applied to the left, the elements $k+1, k+2, \ldots, n$ are annihilated in row $k$, using the corresponding elements of row $k+1$. To obtain a similar matrix, we apply $G_k$ also on the right, thereby adding one more row (column) to the semiseparable part. The whole process is summarized in Figure 4.2.

$$
\begin{array}{cc}
 & \begin{matrix}\downarrow & \downarrow & & & & & & & \end{matrix} \\
\begin{matrix}\rightarrow\\ \rightarrow\\ \\ \\ \\ \\ \\ \\ \end{matrix} &
\begin{pmatrix}
\times&\times&\times&\times&\times&\otimes&\otimes&\otimes&\otimes\\
\times&\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
\times&\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
\times&\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
\times&\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
\otimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
\otimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
\otimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
\otimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes
\end{pmatrix}
\end{array}
\rightarrow
\begin{array}{cc}
 & \begin{matrix} & \downarrow & \downarrow & & & & & & \end{matrix} \\
\begin{matrix} \\ \rightarrow\\ \rightarrow\\ \\ \\ \\ \\ \\ \end{matrix} &
\begin{pmatrix}
\times&\times&\times&\times&\times&0&0&0&0\\
\times&\times&\times&\times&\times&\otimes&\otimes&\otimes&\otimes\\
\times&\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
\times&\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
\times&\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
0&\otimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
0&\otimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
0&\otimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
0&\otimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes
\end{pmatrix}
\end{array}
\rightarrow
\begin{array}{cc}
 & \begin{matrix} & & \downarrow & \downarrow & & & & & \end{matrix} \\
\begin{matrix} \\ \\ \rightarrow\\ \rightarrow\\ \\ \\ \\ \\ \end{matrix} &
\begin{pmatrix}
\times&\times&\times&\times&\times&0&0&0&0\\
\times&\times&\times&\times&\times&0&0&0&0\\
\times&\times&\times&\times&\times&\otimes&\otimes&\otimes&\otimes\\
\times&\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
\times&\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
0&0&\otimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
0&0&\otimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
0&0&\otimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
0&0&\otimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes
\end{pmatrix}
\end{array}
\rightarrow
$$

$$
\begin{array}{cc}
 & \begin{matrix} & & & \downarrow & \downarrow & & & & \end{matrix} \\
\begin{matrix} \\ \\ \\ \rightarrow\\ \rightarrow\\ \\ \\ \\ \end{matrix} &
\begin{pmatrix}
\times&\times&\times&\times&\times&0&0&0&0\\
\times&\times&\times&\times&\times&0&0&0&0\\
\times&\times&\times&\times&\times&0&0&0&0\\
\times&\times&\times&\times&\times&\otimes&\otimes&\otimes&\otimes\\
\times&\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
0&0&0&\otimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
0&0&0&\otimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
0&0&0&\otimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
0&0&0&\otimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes
\end{pmatrix}
\end{array}
\rightarrow
\begin{array}{cc}
 & \begin{matrix} & & & & \downarrow & \downarrow & & & \end{matrix} \\
\begin{matrix} \\ \\ \\ \\ \rightarrow\\ \rightarrow\\ \\ \\ \end{matrix} &
\begin{pmatrix}
\times&\times&\times&\times&\times&0&0&0&0\\
\times&\times&\times&\times&\times&0&0&0&0\\
\times&\times&\times&\times&\times&0&0&0&0\\
\times&\times&\times&\times&\times&0&0&0&0\\
\times&\times&\times&\times&\times&\otimes&\otimes&\otimes&\otimes\\
0&0&0&0&\otimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
0&0&0&0&\otimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
0&0&0&0&\otimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
0&0&0&0&\otimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes
\end{pmatrix}
\end{array}
\rightarrow
\begin{array}{cc}
 & \begin{matrix} & & & & & \downarrow & \downarrow & & \end{matrix} \\
\begin{matrix} \\ \\ \\ \\ \\ \rightarrow\\ \rightarrow\\ \\ \end{matrix} &
\begin{pmatrix}
\times&\times&\times&\times&\boxtimes&\boxtimes&0&0&0\\
\times&\times&\times&\times&\boxtimes&\boxtimes&0&0&0\\
\times&\times&\times&\times&\boxtimes&\boxtimes&0&0&0\\
\times&\times&\times&\times&\boxtimes&\boxtimes&0&0&0\\
\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&0&0&0\\
\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\otimes&\otimes&\otimes\\
0&0&0&0&0&\otimes&\boxtimes&\boxtimes&\boxtimes\\
0&0&0&0&0&\otimes&\boxtimes&\boxtimes&\boxtimes\\
0&0&0&0&0&\otimes&\boxtimes&\boxtimes&\boxtimes
\end{pmatrix}
\end{array}
\rightarrow
$$

$$
\begin{array}{cc}
 & \begin{matrix} & & & & & & \downarrow & \downarrow & \end{matrix} \\
\begin{matrix} \\ \\ \\ \\ \\ \\ \rightarrow\\ \rightarrow\\ \end{matrix} &
\begin{pmatrix}
\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&0&0\\
\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&0&0\\
\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&0&0\\
\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&0&0\\
\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&0&0\\
\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&0&0\\
\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\otimes&\otimes\\
0&0&0&0&0&0&\otimes&\boxtimes&\boxtimes\\
0&0&0&0&0&0&\otimes&\boxtimes&\boxtimes
\end{pmatrix}
\end{array}
\rightarrow
\begin{array}{cc}
 & \begin{matrix} & & & & & & & \downarrow & \downarrow \end{matrix} \\
\begin{matrix} \\ \\ \\ \\ \\ \\ \\ \rightarrow\\ \rightarrow \end{matrix} &
\begin{pmatrix}
\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&\boxtimes&0\\
\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&\boxtimes&0\\
\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&\boxtimes&0\\
\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&\boxtimes&0\\
\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&0\\
\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&0\\
\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&0\\
\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\otimes\\
0&0&0&0&0&0&0&\otimes&\boxtimes
\end{pmatrix}
\end{array}
\rightarrow
\begin{pmatrix}
\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
\times&\times&\times&\times&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes\\
\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes&\boxtimes
\end{pmatrix}.
$$

Figure 4.2: Description of the similarity transformations used to retrieve the semiseparable structure.

This proves the theorem. $\square$

In Chapter 6, the details are given how to implement this algorithm using the Givens-vector representation for the semiseparable part of the matrices involved. This will result in an $O(n^3)$ algorithm to compute the Givens-vector representation of a semiseparable matrix similar to the original symmetric matrix. Taking a closer look at the algorithm reveals that the first Givens transformations at each step in the reduction algorithm can be replaced by a corresponding Householder transformation. In step $k = n - j$ of the reduction a similarity transformation of the following form is performed ${G_{j-1}^{(k)}}^T \ldots {G_2^{(k)}}^T A_0^{(k)} G_2^{(k)} \ldots G_{j-1}^{(k)}$ which can be replaced by a single Householder transformation $H^{(k)}$, such that ${H^{(k)}}^T A_0^{(k)} H^{(k)}$ has the same entries annihilated. Moreover we obtain the same result when transforming the symmetric matrix into a tridiagonal one, which costs $4n^3/3 + O(n^2)$ with Householders, $2n^3 + O(n^2)$ with Givens transformations and then transforming the tridiagonal matrix to a semiseparable matrix by using Givens transformations, which costs an extra $9n^2 + O(n)$. This means that we have, in comparison with the tridiagonalization, an extra cost of $9n^2 + O(n)$. When first reducing matrices to tridiagonal form, intermediate matrices look like:

$$\left(\begin{array}{ccccc|cccc}
\times & \times & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\times & \times & \times & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \times & \times & \times & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \times & \times & \boxtimes & \boxtimes & \boxtimes & 0 & 0 \\
0 & 0 & 0 & \boxtimes & \boxtimes & \boxtimes & \boxtimes & 0 & 0 \\
\hline
0 & 0 & 0 & \boxtimes & \boxtimes & \boxtimes & \boxtimes & 0 & 0 \\
0 & 0 & 0 & \boxtimes & \boxtimes & \boxtimes & \boxtimes & \otimes & \boxtimes \\
0 & 0 & 0 & 0 & 0 & 0 & \otimes & \boxtimes & \boxtimes \\
0 & 0 & 0 & 0 & 0 & 0 & \boxtimes & \boxtimes & \boxtimes
\end{array}\right).$$

Because the algorithm from Theorem 49, and the algorithm which first tridiagonalizes a symmetric matrix, and afterwards makes a semiseparable matrix from it are essentially the same, one might doubt the usefulness of the reduction to semiseparable form. However, the algorithm of Theorem 49 has some advantages which will become clear in the next chapter.

Once the semiseparable matrix has been computed, several algorithms can be considered to compute its eigenvalues (see, for instance, [**35, 38, 69, 130**] and the implicit $QR$-algorithm, as considered in this thesis, see Part 3, Chapter 8).

The proposed algorithm itself, however, while running to completion, gives already information on the spectrum of the initial matrix. In particular, depending on the distribution of the eigenvalues, often, after few steps of the algorithm, the largest eigenvalues are already approximated with high accuracy. This important feature of the algorithm is analyzed in Chapter 5.

## 2. An orthogonal similarity reduction of a matrix to a Hessenberg-like

It is well-known how any square matrix can be transformed into a Hessenberg matrix by orthogonal similarity transformations. As the inverse of a Hessenberg matrix is a Hessenberg-like matrix, one might wonder if it is possible to transform a matrix into a similar Hessenberg-like one. By using an algorithm similar to that one described in the previous section, it is possible to do so for every matrix.

THEOREM 50. *Let $A$ be an $n \times n$ matrix. There exists an orthogonal matrix $U$ such that the following equation is satisfied:*

$$U^T AU = Z,$$

*for which $Z$ is a Hessenberg-like matrix.*

PROOF. In Theorem 49, it was unimportant if one first annihilated the elements in the last row or in the last column, because the matrix was symmetric. Here, however our matrix is not necessarily symmetric anymore. Therefore the transformations for annihilating elements in the upper or lower triangular part, are not necessarily the same anymore. To obtain a Hessenberg-like matrix, the transformation matrix $U$ should be composed of Householder and Givens transformations, constructed, to annihilate elements in the lower triangular part of $A$. Applying then the transformation $U^T AU$ results in a matrix whose lower triangular part is semiseparable. Hence the matrix is a Hessenberg-like matrix. $\square$

Note that the algorithm to reduce an arbitrary matrix, by means of orthogonal transformations into a similar Hessenberg-like matrix also requires $O(n^3)$ operations. It is obvious that the previous theorem can also be formulated for a lower Hessenberg-like matrix.

## 3. An orthogonal reduction of a matrix to upper (lower) triangular semiseparable form

Similar to the Householder bidiagonalization [**90**], the algorithm given here makes use of orthogonal transformations, i.e., Givens and Householder transformations, to reduce the matrix $A$ into an upper triangular semiseparable matrix. The algorithm can be retrieved from the constructive proof of the following theorem.

THEOREM 51. *Let $A \in \mathbb{R}^{m \times n}$, $m \geq n$. There exist two orthogonal matrices $U \in \mathbb{R}^{m \times m}$ and $V \in \mathbb{R}^{n \times n}$ such that*

$$U^T A V = \begin{pmatrix} S_u \\ 0 \end{pmatrix},$$

*where $S_u$ is an upper triangular semiseparable matrix.*

The case $n \geq m$ is formulated in Theorem 52.

PROOF. The proof given here is a constructive one. We prove the existence of such a transformation by reducing the matrix $A$ to the appropriate form by using Givens and Householder transformations.

The proof is given by finite induction. The proof is outlined for a matrix $A \in \mathbb{R}^{m \times n}$, with $m = 6$ and $n = 5$, as this case illustrates the general case. The side on which the operations are performed plays a very important role. In fact, the first phase of the algorithm constructs two sequences of matrices $A^{(l)}$ and $A^{(l)}_{k,j}$, with $l = 1, \ldots, n$, $k = 0, \ldots, l+1$, and $j, = 0, \ldots, l$, starting from $A^{(1)} = A$, according to one of the following rules:

$$U^{(l)}_{k+1} A^{(l)}_{k,j} = A^{(l)}_{k+1,j}, \quad A^{(l)}_{k,j} V^{(l)}_{j+1} = A^{(l)}_{k,j+1}.$$

Furthermore, one lets $A^{(l)}_{0,0} = A^{(l)}$ and $A^{(l+1)} = A^{(l)}_{l+1,l}$. The applied transformations to the left and to the right of $A^{(l)}_{k,j}$ are Givens and/or Householder transformations.

Step 1. In this first step of the algorithm three orthogonal matrices ${U^{(1)}_1}^T$, ${U^{(1)}_2}^T$ and $V^{(1)}_1$ are to be found, such that the matrix $A^{(2)} = {U^{(1)}_2}^T {U^{(1)}_1}^T A^{(1)} V^{(1)}_1$, with $A^{(1)} = A$, has the following properties: the first two rows of the matrix $A^{(2)}$ satisfy already the semiseparable structure and the first column of $A^{(2)}$ is zero below the first element. A Householder transformation $V^{(1)}_1$ is applied to the right of $A^{(1)} = A^{(1)}_{0,0}$ in order to annihilate all the elements in the first row except for the first one. The elements denoted with $\otimes$ will be annihilated, and the ones denoted with $\boxtimes$ mark the part of the matrix having already a semiseparable structure:

$$\left(\begin{array}{ccccc} \times & \otimes & \otimes & \otimes & \otimes \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \end{array}\right) \xrightarrow{A_{0,0}^{(1)}V_1^{(1)}} \left(\begin{array}{ccccc} \times & 0 & 0 & 0 & 0 \\ \times & \times & \times & \times & \times \\ \otimes & \times & \times & \times & \times \\ \otimes & \times & \times & \times & \times \\ \otimes & \times & \times & \times & \times \\ \otimes & \times & \times & \times & \times \end{array}\right)$$

$$\Updownarrow$$

$$A_{0,0}^{(1)} = A^{(1)} \quad \xrightarrow{A_{0,0}^{(1)}V_1^{(1)}} \quad A_{0,1}^{(1)}.$$

A Householder transformation ${U_1^{(1)}}^T$ is now applied to the left of $A_{0,1}^{(1)}$ in order to annihilate all the elements in the first column except the first two ones, $A_{1,1}^{(1)} = {U_1^{(1)}}^T A_{0,1}^{(1)}$, followed by a Givens transformation ${U_2^{(1)}}^T$ applied to the left of $A_{1,1}^{(1)}$ in order to annihilate the second element in the first column. As a consequence, the first two rows of $A_{2,1}^{(1)}$ have already a semiseparable structure:

$$\left(\begin{array}{ccccc} \times & 0 & 0 & 0 & 0 \\ \otimes & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \end{array}\right) \xrightarrow{{U_2^{(1)}}^T A_{1,1}^{(1)}} \left(\begin{array}{ccccc} \boxtimes & \boxtimes & \boxtimes & \boxtimes & \boxtimes \\ 0 & \boxtimes & \boxtimes & \boxtimes & \boxtimes \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \end{array}\right)$$

$$\Updownarrow$$

$$A_{1,1}^{(1)} \quad \xrightarrow{{U_2^{(1)}}^T A_{1,1}^{(1)}} \quad A_{2,1}^{(1)}.$$

Then we put $A^{(2)} = A_{2,1}^{(1)}$.

Step $k$ By induction, for $k > 1$. The first $k$ rows of $A^{(k)}$ have a semiseparable structure and the first $k-1$ columns are already in an upper triangular form. In fact, the upper left $k$ by $k$ block is already an upper triangular semiseparable matrix. Without loss of generality, let us assume $k = 3$. This means that $A^{(3)}$ has the following structure:

$$A^{(3)} = \left(\begin{array}{ccccc} \boxtimes & \boxtimes & \boxtimes & \boxtimes & \boxtimes \\ 0 & \boxtimes & \boxtimes & \boxtimes & \boxtimes \\ 0 & 0 & \boxtimes & \boxtimes & \boxtimes \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & \times & \times & \times \end{array}\right).$$

The aim of this step is to make the upper triangular semiseparable structure in the first 4 rows and the first 3 columns of the matrix. To this end, a Householder transformation $V_1^{(3)}$ is applied to the right of $A^{(3)}$, chosen in order to annihilate the last two elements of the first row of $A^{(3)}$. Note that because of the dependency between the first three rows, $V_1^{(3)}$ annihilates the last two entries of the second and third row, too. Furthermore, a Householder transformation is performed to the left of the matrix $A_{0,1}^{(3)}$ to annihilate the last two elements in column 3:

$$
\left(\begin{array}{ccccc}
\boxtimes & \boxtimes & \boxtimes & 0 & 0 \\
0 & \boxtimes & \boxtimes & 0 & 0 \\
0 & 0 & \boxtimes & 0 & 0 \\
0 & 0 & \times & \times & \times \\
0 & 0 & \otimes & \times & \times \\
0 & 0 & \otimes & \times & \times
\end{array}\right)
\xrightarrow{U_1^{(3)^T} A_{0,1}^{(3)}}
\left(\begin{array}{ccccc}
\boxtimes & \boxtimes & \boxtimes & 0 & 0 \\
0 & \boxtimes & \boxtimes & 0 & 0 \\
0 & 0 & \boxtimes & 0 & 0 \\
0 & 0 & \times & \times & \times \\
0 & 0 & 0 & \times & \times \\
0 & 0 & 0 & \times & \times
\end{array}\right)
$$

$$\Updownarrow$$

$$A_{0,1}^{(3)} = A_{0,0}^{(3)} V_1^{(3)} \quad \xrightarrow{U_1^{(3)^T} A_{0,1}^{(3)}} \quad A_{1,1}^{(3)}.$$

The Givens transformation $U_2^{(3)^T}$ is now applied to the left of the matrix $A_{1,1}^{(3)}$, annihilating the element marked with a circle:

$$
\left(\begin{array}{ccccc}
\times & \boxtimes & \boxtimes & 0 & 0 \\
0 & \boxtimes & \boxtimes & 0 & 0 \\
0 & 0 & \boxtimes & 0 & 0 \\
0 & 0 & \otimes & \times & \times \\
0 & 0 & 0 & \times & \times \\
0 & 0 & 0 & \times & \times
\end{array}\right)
\xrightarrow{U_2^{(3)^T} A_{1,1}^{(3)}}
\left(\begin{array}{ccccc}
\times & \boxtimes & \boxtimes & 0 & 0 \\
0 & \boxtimes & \boxtimes & 0 & 0 \\
0 & 0 & \boxtimes & \boxtimes & \boxtimes \\
0 & 0 & 0 & \boxtimes & \boxtimes \\
0 & 0 & 0 & \times & \times \\
0 & 0 & 0 & \times & \times
\end{array}\right)
$$

$$\Updownarrow$$

$$A_{1,1}^{(3)} \quad \xrightarrow{U_2^{(3)^T} A_{1,1}^{(3)}} \quad A_{2,1}^{(3)}.$$

Dependency is now created between the fourth and the third row. Nevertheless, as it can be seen in the figure above, the upper part does not satisfy the semiseparable structure, yet. A chasing technique is used in order to chase the nonsemiseparable structure upwards and away, by means of Givens transformations. Applying $V_2^{(3)}$ to the right to annihilate the entry $(2,3)$ of $A_{2,1}^{(3)}$, a nonzero element is introduced in the third row on the second column (i.e. in position $(3,2)$). Because of the semiseparable structure, this operation introduces two zeros in the third column, namely in the first and second row. Annihilating the element just created in the third row with a Givens transformation to the left, the semiseparable structure holds between the second and the third row:

$$\begin{pmatrix} \times & \boxtimes & 0 & 0 & 0 \\ 0 & \boxtimes & 0 & 0 & 0 \\ 0 & \otimes & \times & \boxtimes & \boxtimes \\ 0 & 0 & 0 & \boxtimes & \boxtimes \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times & \times \end{pmatrix} \xrightarrow{U_3^{(3)^T} A_{2,2}^{(3)}} \begin{pmatrix} \times & \boxtimes & 0 & 0 & 0 \\ 0 & \boxtimes & \boxtimes & \boxtimes & \boxtimes \\ 0 & 0 & \boxtimes & \boxtimes & \boxtimes \\ 0 & 0 & 0 & \boxtimes & \boxtimes \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times & \times \end{pmatrix}$$

$$\Updownarrow$$

$$A_{2,2}^{(3)} = A_{2,1}^{(3)} V_2^{(3)} \quad \xrightarrow{U_3^{(3)^T} A_{2,2}^{(3)}} \quad A_{3,2}^{(3)}.$$

This up-chasing of the semiseparable structure can be repeated to create a complete upper semiseparable part starting from row 4 to row 1.

$$\begin{pmatrix} \times & 0 & 0 & 0 & 0 \\ \otimes & \times & \boxtimes & \boxtimes & \boxtimes \\ 0 & 0 & \boxtimes & \boxtimes & \boxtimes \\ 0 & 0 & 0 & \boxtimes & \boxtimes \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times & \times \end{pmatrix} \xrightarrow{U_4^{(3)^T} A_{3,3}^{(3)}} \begin{pmatrix} \boxtimes & \boxtimes & \boxtimes & \boxtimes & \boxtimes \\ 0 & \boxtimes & \boxtimes & \boxtimes & \boxtimes \\ 0 & 0 & \boxtimes & \boxtimes & \boxtimes \\ 0 & 0 & 0 & \boxtimes & \boxtimes \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times & \times \end{pmatrix}$$

$$\Updownarrow$$

$$A_{3,3}^{(3)} = A_{3,2}^{(3)} V_3^{(3)} \quad \xrightarrow{U_4^{(3)^T} A_{3,3}^{(3)}} \quad A_{4,3}^{(3)}.$$

Then we put $A^{(4)} = A_{4,3}^{(3)}$. This proves the induction step.

Step $n+1$ This step is only performed if $m > n$. Finally, a Householder transformation has to be performed to the left to have a complete upper triangular semiseparable structure. In fact, suppose, the matrix has already the semiseparable structure in the first $n$ rows, then one single Householder transformation is needed to annihilate all the elements in the $n$th column below the $n$th row:

$$\begin{pmatrix} \boxtimes & \boxtimes & \boxtimes & \boxtimes & \boxtimes \\ 0 & \boxtimes & \boxtimes & \boxtimes & \boxtimes \\ 0 & 0 & \boxtimes & \boxtimes & \boxtimes \\ 0 & 0 & 0 & \boxtimes & \boxtimes \\ 0 & 0 & 0 & 0 & \boxtimes \\ 0 & 0 & 0 & 0 & \times \end{pmatrix} \xrightarrow{U_1^{(5)^T} A_{0,0}^{(5)}} \begin{pmatrix} \boxtimes & \boxtimes & \boxtimes & \boxtimes & \boxtimes \\ 0 & \boxtimes & \boxtimes & \boxtimes & \boxtimes \\ 0 & 0 & \boxtimes & \boxtimes & \boxtimes \\ 0 & 0 & 0 & \boxtimes & \boxtimes \\ 0 & 0 & 0 & 0 & \boxtimes \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\Updownarrow$$

$$A^{(5)} \quad \xrightarrow{U_1^{(5)^T} A_{0,0}^{(5)}} \quad A_{1,0}^{(5)}.$$

After the latter Householder transformation, the desired upper triangular semiseparable structure is created and the theorem is proved.

$\square$

The reduction just described is obtained by applying Givens and Householder transformations to $A$. Note that the computational complexity of applying the

Householder transformations is the same as of the standard procedure that reduces matrices into bidiagonal form using Householder transformations. This complexity is $4mn^2 - 4/3n^3$ (see [**91**]). At step $k$ of the reduction procedure, described above, $k$ Givens transformations $U_j^{(k)}, j = 2, 3, \ldots, k+1$ are applied to the left and $k-1$ Givens transformations $V_j^{(k)}$, $j = 2, 3, \ldots, k$ are applied to the right of the almost upper triangular semiseparable part of $A_{1,1}^{(k)}$. The purpose of these Givens transformations is to chase the bulge on the subdiagonal upwards while maintaining the upper triangular semiseparable structure in the upper part of the matrix. Applying the $k-1$ Givens transformations $V_j^{(k)}$ on the first $(k+1)$ columns of $A_{1,1}^{(k)}$ requires only $O(k)$ flops using the Givens-vector representation of the upper left upper triangular semiseparable matrix. Applying the $k$ Givens transformations $U_j^{(k)}$ on the upper left part of the matrix $A_{1,1}^{(k)}$ requires also only $O(k)$ flops using the Givens-vector representation of the upper triangular semiseparable matrix. Because the upper right part of the matrix $A_{1,1}^{(k)}$ can be written as a $(k+1) \times (n-k-1)$ matrix of rank 1, applying the Givens transformations $U_j^{(k)}$ on this part of $A_{1,1}^{(k)}$ also requires only $O(k)$ flops. Hence applying the Givens transformations during the whole reduction requires $O(n^2)$ flops.

A generalization of the previous result is formulated in the following theorem.

THEOREM 52. *Let $A \in \mathbb{R}^{m \times n}$. There exist two orthogonal matrices $U \in \mathbb{R}^{m \times m}$ and $V \in \mathbb{R}^{n \times n}$ such that*

- *for $m \geq n$:*
$$U^T A V = \begin{pmatrix} S_l \\ 0 \end{pmatrix},$$
*where $S_l$ is a lower triangular semiseparable matrix,*
- *for $m \leq n$:*
$$U^T A V = \begin{pmatrix} S_u & 0 \end{pmatrix},$$
*where $S_u$ is an upper triangular semiseparable matrix,*
- *for $m \leq n$:*
$$U^T A V = \begin{pmatrix} S_l & 0 \end{pmatrix},$$
*where $S_l$ is a lower triangular semiseparable matrix.*

PROOF. Using the transpose operation and permuting rows or columns, these results can easily be obtained from Theorem 51. □

## Conclusions

In this chapter we proposed two new basic algorithms. An orthogonal similarity transformation to transform symmetric matrices into symmetric semiseparable ones was constructed and secondly an orthogonal transformation to reduce a matrix into upper triangular semiseparable form was designed. We showed that the first transformation also can be used to transform nonsymmetric matrices into Hessenberg-like matrices.

Allthough both algorithms are slightly slower than the corresponding reductions to tridiagonal (resp. Hessenberg) and bidiagonal form, we will show in the next chapter that these reductions have some advantages, with respect to the tridiagonal approach.

CHAPTER 5

# Convergence properties of the different reduction algorithms

In the previous section three types of reductions to semiseparable form were designed, namely to symmetric semiseparable form, to Hessenberg-like form and to upper triangular semiseparable form. It was stated there that the orthogonal similarity reduction to a similar semiseparable matrix could also be achieved by first transforming the matrix to a similar tridiagonal one and then transforming the tridiagonal matrix into a similar semiseparable one. It is a known result that the already reduced part in the tridiagonal reduction contains the Lanczos-Ritz values. This same behavior is observed in the reduction to semiseparable form. We will prove this statement in this chapter. Secondly there was also a remark, which stated that the reduction, although it costs an extra $9n^2 + O(n)$ with respect to the tridiagonalization, was still interesting because there was some extra convergence property. While performing the reduction to semiseparable form, some kind of nested subspace iteration is performed. Also this behavior will be investigated in detail in this chapter. Both of the convergence properties, namely the Lanczos-Ritz values behavior, and the subspace iteration appear in all three reduction algorithms as will be explained.

In the first section we deduce a general theorem, providing necessary and sufficient conditions, for an orthogonal similarity transformation to obtain the Arnoldi-Ritz values in the already reduced part of the matrix. The theorem provided is very simple, only two conditions have to be placed on the reduction algorithm, but it is completely general. This means, that using this theorem the convergence properties of the transformations to tridiagonal, semiseparable, Hessenberg and Hessenberg-like can be proved in one line.

In the second section we will investigate in detail the convergence behavior of the orthogonal similarity reduction of a symmetric matrix to semiseparable form. As a consequence of the theorem provided in the first section we can immediately see that the semiseparable matrix has the Lanczos-Ritz values as eigenvalues. But, as already stated before, there is more: the reduction algorithm also performs some kind of nested subspace iteration. We will prove that the chasing technique, to chase the semiseparable structure downwards in the reduction algorithm, performs in fact a step of the $QR$-algorithm (in fact it is a step of $QL$) on the already semiseparable part of the matrix. This nested subspace iteration interacts with the Lanczos behavior and will therefore be investigated in more detail in the last subsection of this section.

In the third section, briefly some results will be mentioned concerning the orthogonal similarity transformation of a matrix into a similar Hessenberg-like. Briefly, because the results are straightforward knowing the convergence properties of the symmetric case.

The orthogonal reduction of an arbitrary matrix to an upper triangular semiseparable matrix is the subject of the fourth section. Although the transformation performed is not a similarity transformation, it is closely related to the orthogonal similarity transformation of a symmetric matrix to semiseparable form. A theorem will be provided giving directly the link between these two transformations. Using this theorem one can directly translate all the convergence properties of Section 2 towards this transformation.

## 1. The Arnoldi(Lanczos)-Ritz values in orthogonal similarity reductions

It is well-known that while reducing a symmetric matrix into a similar tridiagonal one, the intermediate tridiagonal matrices contain the Lanczos-Ritz values as eigenvalues. Or for a Hessenberg matrix they contain the so-called Arnoldi-Ritz values. More information can be found in the following books [**42, 45, 91, 140, 155, 169**] and the references therein.

In this section we provide necessary and sufficient conditions stating, whether partially reduced matrices during an orthogonal similarity reduction will contain the Arnoldi-Ritz values. It will also be shown that the reductions to a similar tridiagonal, Hessenberg, semiseparable and Hessenberg-like matrix are reductions satisfying the desired properties.

**1.1. Ritz values and Arnoldi(Lanczos)-Ritz values.** We will briefly introduce the notion "Ritz values". Suppose, we have a matrix $A^{(0)} = A$, which is transformed via an orthogonal similarity transformation into the matrix $A^{(1)} = Q_0^T A^{(0)} Q_0$. We remark that the transformation $Q_0$ is not essential. Quite often the orthogonal matrix $Q_0$ is equal to the identity matrix, e.g., in the reduction to semiseparable or tridiagonal form the matrix $Q_0$ equals the identity matrix $I$. It is however perfectly possible to perform a first transformation $Q_0$ on the matrix $A^{(0)}$. This will not affect the reduction algorithms, but it will affect the convergence behavior of the reduction, as we will show in this subsection. The other orthogonal transformations $Q_k$, $k \geq 1$, are constructed by the reduction algorithms:

$$A^{(2)} = Q_1^T A^{(1)} Q_1.$$

Let us denote the orthogonal transformation to go from $A^{(k)}$ to $A^{(k+1)}$ as $Q_k$, and we denote with $Q_{0:k}$ the orthogonal matrix equal to $Q_0 Q_1 \ldots Q_k$. This means that

$$\begin{aligned} A^{(k+1)} &= Q_k^T A^{(k)} Q_k \\ &= Q_k^T Q_{k-1}^T \ldots Q_1^T Q_0^T A Q_0 Q_1 \ldots Q_{k-1} Q_k \\ &= Q_{0:k}^T A Q_{0:k}. \end{aligned}$$

The matrix $A^{(k+1)}$ is of the following form:

$$\left(\begin{array}{c|c} R_{k+1} & \times \\ \hline \times & A_{k+1} \end{array}\right)$$

where $R_{k+1}$ stands for that part of the matrix of dimension $(k+1) \times (k+1)$ which is already transformed to the appropriate form, e.g., tridiagonal, semiseparable, Hessenberg, etc. The matrix $A_{k+1}$ is of dimension $(n-k-1) \times (n-k-1)$. Let us partition the matrix $Q_{0:k}$ as follows:

$$Q_{0:k} = \left( \overleftarrow{Q}_{0:k} | \overrightarrow{Q}_{0:k} \right) \quad \text{with} \quad \begin{cases} \overleftarrow{Q}_{0:k} \in \mathbb{R}^{n \times (k+1)} \\ \overrightarrow{Q}_{0:k} \in \mathbb{R}^{n \times (n-k-1)}. \end{cases}$$

This means,

$$A \left( \overleftarrow{Q}_{0:k} | \overrightarrow{Q}_{0:k} \right) = \left( \overleftarrow{Q}_{0:k} | \overrightarrow{Q}_{0:k} \right) \left( \begin{array}{c|c} R_{k+1} & \times \\ \hline \times & A_{k+1} \end{array} \right).$$

The eigenvalues of $R_{k+1}$ are called the Ritz values of $A$ with respect to the subspace spanned by the columns of $\overleftarrow{Q}_{0:k}$ (see e.g. [**45**]).

Suppose we have now the Krylov subspace of dimension $k$ with initial vector $v$:

$$\mathcal{K}_k(A, v) = \langle v, Av, \ldots, A^{k-1}v \rangle.$$

Where $\langle x, y, z \rangle$ denotes the vector space spanned by the vectors $x, y$ and $z$. If the columns of the matrix $\overleftarrow{Q}_{0:k}$ form an orthonormal basis of the Krylov subspace $\mathcal{K}_{k+1}(A, v)$, then we say that the eigenvalues of $R_{k+1}$ are called the Arnoldi-Ritz values of $A$ with respect to the initial vector $v$. If the matrix $A$ is symmetric, one often calls the Ritz values the Lanczos-Ritz values.

**1.2. Necessary conditions to obtain the Arnoldi(Lanczos)-Ritz values as eigenvalues in the already reduced block of the matrix.** In this subsection, we investigate the properties of orthogonal similarity transformations, where the eigenvalues in the already reduced block of the matrix are the Arnoldi-Ritz values, with respect to the starting vector $v$, where $v/\|v\| = \pm Q_0 e_1$.

Suppose that our similarity reduction of the matrix into another matrix has the following form after step $k-1$ (with $k = 1, 2, \ldots, n-1$):

$$\begin{pmatrix} R_k & \times \\ \times & \times \end{pmatrix} = Q_{0:k-1}^T A Q_{0:k-1}.$$

This means that we start with this matrix at step $k$ of the reduction: with $R_k$ a square matrix of dimension $k$, which has as eigenvalues the Arnoldi-Ritz values. Moreover we have the following properties for the orthogonal matrix $Q_{0:k-1}$:

(1) The columns of $\overleftarrow{Q}_{0:k-1}$ form an orthogonal basis for $\mathcal{K}_k(A, v)$.
(2) The columns of $\overrightarrow{Q}_{0:k-1}$ form an orthogonal basis for the orthogonal complement of $\mathcal{K}_k(A, v)$.

After the next step in the transformation we have that the block $R_{k+1}$ has the Arnoldi-Ritz values as eigenvalues with respect to $\mathcal{K}_{k+1}(A, v)$. This results in two easy conditions, similar to the ones described above. After step $k$, in the beginning of step $k+1$ we have:

(1) The columns of $\overleftarrow{Q}_{0:k}$ form an orthogonal basis for $\mathcal{K}_k(A, v) + \langle A^k v \rangle$.
(2) The columns of $\overrightarrow{Q}_{0:k}$ form an orthogonal basis for the orthogonal complement of $\mathcal{K}_{k+1}(A, v)$.

We have the following equalities:

$$\begin{aligned} A &= Q_{0:k-1}A^{(k)}Q_{0:k-1}^T \\ &= Q_{0:k}A^{(k+1)}Q_{0:k}^T. \end{aligned}$$

This means that the transformation to go from matrix $A^{(k)}$ to matrix $A^{(k+1)}$ can also be written in the following form:

$$Q_{0:k}^T Q_{0:k-1} A^{(k)} Q_{0:k-1}^T Q_{0:k} = A^{(k+1)}.$$

Using the fact that $Q_k$ denotes the orthogonal matrix to go from matrix $A^{(k)}$ to matrix $A^{(k+1)}$, we get:

$$\begin{aligned} Q_k^T &= Q_{0:k}^T Q_{0:k-1} \\ &= \begin{pmatrix} \overleftarrow{Q}_{0:k}^T \\ \overrightarrow{Q}_{0:k}^T \end{pmatrix} \left( \overleftarrow{Q}_{0:k-1} | \overrightarrow{Q}_{0:k-1} \right) \\ &= \begin{pmatrix} (Q_k)_{11}^T & (Q_k)_{12}^T \\ (Q_k)_{21}^T & (Q_k)_{22}^T \end{pmatrix} \end{aligned}$$

where the $(Q_k)_{11}^T, (Q_k)_{12}^T, (Q_k)_{21}^T$ and $(Q_k)_{22}^T$ denote a partitioning of the matrix $Q_k^T$. These blocks have the following dimensions: $(Q_k)_{11}^T \in \mathbb{R}^{(k+1)\times k}$, $(Q_k)_{12}^T \in \mathbb{R}^{(k+1)\times(n-k)}$, $(Q_k)_{21}^T \in \mathbb{R}^{(n-k-1)\times k}$ and $(Q_k)_{22}^T \in \mathbb{R}^{(n-k-1)\times(n-k)}$. It can be seen rather easily, by combining the properties of the matrices $Q_{0:k-1}$ and $Q_{0:k}$ from above, that the block $(Q_k)_{21}^T$ has to be zero. This zero block in the matrix $Q_k$ is the first necessary condition.

To obtain a second condition, we will investigate the structure of an intermediate matrix matrix $\tilde{A}^{(k)}$ satisfying

$$\begin{aligned} \tilde{A}^{(k)} &= Q_k^T A^{(k)} \\ &= Q_k^T Q_{0:k-1}^T A Q_{0:k-1}, \end{aligned}$$

which can be rewritten as:

$$Q_{0:k}\tilde{A}^{(k)} = AQ_{0:k-1}. \tag{19}$$

Rewriting equation (19) gives us:

$$A\left( \overleftarrow{Q}_{0:k-1} | \overrightarrow{Q}_{0:k-1} \right) = \left( \overleftarrow{Q}_{0:k} | \overrightarrow{Q}_{0:k} \right) \tilde{A}^{(k)}.$$

Because the columns of $A\overleftarrow{Q}_{0:k-1}$ belong to the Krylov subspace: $\mathcal{K}_{k+1}(A, v)$, which is spanned by the columns of $\overleftarrow{Q}_{0:k}$, we have that $\tilde{A}^{(k)}$ has a zero block of dimension $(n-k-1)\times k$ in the lower left corner. This provides us a second condition.

The two conditions presented here, namely the condition on $\tilde{A}^{(k)}$ and the condition on $Q_k$, are necessary to have the desired convergence properties in the reduction. In the next section we will prove that they are also sufficient. We will formulate this as a theorem:

THEOREM 53. *Suppose, we apply a similarity transformation on the matrix $A$ (as described in Section 1.1), such that the already reduced part $R_k$ in the matrix has the Arnoldi-Ritz values in each step of the reduction algorithm. Then we have the following two properties:*

- *The matrix $Q_k^T$, which is the orthogonal matrix to transform $A^{(k)}$ into the matrix $A^{(k+1)} = Q_k^T A^{(k)} Q_k$ has a zero block of dimension $(n-k-1)\times k$ in the lower left corner.*
- *The matrix $\tilde{A}^{(k)} = Q_k^T A^{(k)}$ has a zero block of dimension $(n-k-1)\times k$ in the lower left corner.*

**1.3. Sufficient conditions to obtain the convergence behavior.** We prove that the properties from Theorem 53 connected to the matrices $Q_k$ and $\tilde{A}^{(k)}$ are sufficient to have the Arnoldi-Ritz values as eigenvalues in the blocks $A_k$.

THEOREM 54. *Suppose, we apply a similarity transformation on the matrix $A$ (as described in Section 1.1), such that we have for $A^{(0)} = A$:*

$$Q_0 e_1 = \pm\frac{v}{\|v\|} \text{ and } Q_0^T A^{(0)} Q_0 = A^{(1)}.$$

*Suppose that at step $k$ of the reduction algorithm we have the following two properties:*
- *the matrix $Q_k^T$, which is the orthogonal matrix to transform $A^{(k)}$ into the matrix $A^{(k+1)} = Q_k^T A^{(k)} Q_k$ has a zero block of dimension $(n-k-1)\times k$ in the lower left corner;*
- *the matrix $\tilde{A}^{(k)} = Q_k^T A^{(k)}$ has a zero block of dimension $(n-k-1)\times k$ in the lower left corner.*

*Then we have that for the matrix $A^{(k)}$ partitioned as*

$$A^{(k)} = \left(\begin{array}{c|c} R_k & \times \\ \hline \times & A_k \end{array}\right),$$

*that the matrix $R_k$ of dimension $k\times k$ has the Arnoldi-Ritz values with respect to the Krylov space $\mathcal{K}_k(A,v)$.*

PROOF. We will prove the theorem by induction on $k$.

Step 1 The theorem is true for $k=1$, because $Q_0^T A Q_0$ contains clearly the Arnoldi-Ritz value in the upper left $1\times 1$ block.

Step $k$ Suppose the theorem is true for $A^{(1)}, A^{(2)}, \ldots, A^{(k-1)}$ This means that the columns of $\overleftarrow{Q}_{0:k-1}$ span the Krylov subspace $\mathcal{K}_k(A,v)$. Then we will prove now that the conditions are sufficient to have that the columns of $\overleftarrow{Q}_{0:k}$ span the Krylov subspace of $\mathcal{K}_{k+1}(A,v)$. We have the following equalities

$$\begin{aligned} \tilde{A}^{(k)} &= Q_k^T A^{(k)} \\ &= Q_k^T Q_{0:k-1}^T A Q_{0:k-1} \\ A Q_{0:k-1} &= Q_{0:k} \tilde{A}^{(k)} \\ &= \left(\overleftarrow{Q}_{0:k} | \overrightarrow{Q}_{0:k}\right) \tilde{A}^{(k)}. \end{aligned}$$

Hence, we have already that the columns of $A\overleftarrow{Q}_{0:k-1}$ are part of the space spanned by the columns of $\overleftarrow{Q}_{0:k}$. Note that the columns of $A\overleftarrow{Q}_{0:k-1}$ span the same space as $A\mathcal{K}_k(A,v)$. We have the following relation:

$$A\mathcal{K}_k(A,v) \subseteq \mathrm{Range}(\overleftarrow{Q}_{0:k}). \tag{20}$$

With Range$(A)$ we denote the vector space spanned by the columns of the matrix $A$. Since $Q_k^T$ has a zero block in the lower left position, we have that:

$$\begin{aligned} Q_{0:k} &= Q_{0:k-1}Q_k \\ Q_{0:k}Q_k^T &= Q_{0:k-1}. \end{aligned}$$

Hence,

$$\left( \overleftarrow{Q}_{0:k} | \overrightarrow{Q}_{0:k} \right) Q_k^T = \left( \overleftarrow{Q}_{0:k-1} | \overrightarrow{Q}_{0:k-1} \right).$$

Using the zero structure of the matrix $Q_k^T$ we have:

$$\text{Range}(\overleftarrow{Q}_{0:k-1}) = \mathcal{K}_k(A,v) \quad \subseteq \quad \text{Range}(\overleftarrow{Q}_{0:k}).$$

When we combine this, with equation (20) we get:

$$\text{Range}(\overleftarrow{Q}_{0:k}) = \mathcal{K}_{k+1}(A,v).$$

This proves the theorem for $A^{(k)}$.

□

**1.4. Some general remarks.** When we take a closer look at the matrix equation:

$$\begin{aligned} Q_k^T &= Q_{0:k}^T Q_{0:k-1} \\ &= \begin{pmatrix} \overleftarrow{Q}_{0:k}^T \\ \overrightarrow{Q}_{0:k}^T \end{pmatrix} \left( \overleftarrow{Q}_{0:k-1} | \overrightarrow{Q}_{0:k-1} \right), \end{aligned}$$

we can see that the matrix $Q_k^T$ has the upper right $(k+1) \times (n-k)$ block of rank less than or equal to 1. The upper right $(k+1) \times (n-k)$ block corresponds to the product $\overleftarrow{Q}_{0:k}^T \overrightarrow{Q}_{0:k-1}$. The columns of the matrix $\overleftarrow{Q}_{0:k}^T$ span the Krylov subspace $\mathcal{K}_{k+1}(A,v) = \mathcal{K}_k(A,v) + \langle A^k v \rangle$ and the columns of $\overrightarrow{Q}_{0:k-1}$ span the space orthogonal to $\mathcal{K}_k(A,v)$, which leads directly to the fact that the product $\overleftarrow{Q}_{0:k}^T \overrightarrow{Q}_{0:k-1}$, has rank less than or equal to 1.

The reader can easily verify that the similarity reductions of a symmetric matrix into a similar tridiagonal and the similarity reduction of a matrix into a similar Hessenberg, perfectly fit in this scheme. Moreover one can derive that the vector $v$ equals $e_1$, if of course the initial transformation $Q_0$ equals the identity matrix.

## 2. Convergence properties of an orthogonal similarity reduction of a symmetric matrix to semiseparable form

The similarity reduction of a symmetric matrix into a similar semiseparable one (as described in Section 1 of Chapter 4), has two different types of convergence behavior inside of the reduction algorithm. First we will prove that the intermediate, already reduced matrices have the Lanczos-Ritz values as eigenvalues. In the second subsection we will prove that the algorithm also performs some kind of nested subspace iteration on the matrix. The third subsection will investigate the interaction between these two convergence behaviors.

**2.1. The Lanczos-Ritz values in an orthogonal similarity reduction of a matrix into semiseparable form.** Let us partition the matrix $A_0^{(k)}$ (as in the proof of Theorem 49) as follows:

$$A_0^{(k)} = A^{(k)} = \left(\begin{array}{c|c} A_k & u_k v_k^T \\ \hline v_k u_k^T & S_k \end{array}\right),$$

with $A_k \in \mathbb{R}^{(n-k)\times(n-k)}$, $u_k \in \mathbb{R}^{(n-k)\times 1}$, $v_k \in \mathbb{R}^{k\times 1}$and $S_k \in \mathbb{R}^{k\times k}$. From the semiseparable structure, we know that $[v_k, S_k e_1]$ has rank less than or equal to one.

We see that the matrix $A^{(k)}$ has the lower right block already of semiseparable form. To apply the theorem as described in Section 1.3, we need the upper left part to be semiseparable. It is clear that we can arrange this.

THEOREM 55. *Let $A \in \mathbb{R}^{n\times n}$. Then there exists an orthogonal similarity reduction to semiseparable form, such that the intermediate matrices $A^{(k)}$ have the upper left $k \times k$ block of semiseparable form.*

PROOF. Apply the reduction of Theorem 49 to the matrix $A$. The intermediate matrices $A^{(k)}$ arising from the reduction in Theorem 49 have the lower right block semiseparable. Denote with $Q_{0:k-1}$ an intermediate orthogonal matrix such that the following equation is satisfied:

$$A^{(k)} = Q_{0:k-1}^T A Q_{0:k-1}.$$

This means that for $J$ equal to the counteridentity matrix we have:

$$\begin{aligned} \tilde{A}^{(k)} &= JA^{(k)}J \\ &= JQ_{0:k-1}{}^T A Q_{0:k-1} J \\ &= \tilde{Q}_{0:k-1}^T A \tilde{Q}_{0:k-1}. \end{aligned}$$

and $\tilde{A}^{(k)}$, has the upper left block of semiseparable form. □

Note that this algorithm is very easy to implement. Instead of using the counteridentity matrix $J$, we start the reduction on the first row and annihilate all the elements except for the first one, by orthogonal transformations performed on the right. Applying then the corresponding transformations on the left of the matrix the reader can easily verify, that the upper left block of the matrix is semiseparable and has as eigenvalues the Lanczos-Ritz values.

**2.2. subspace iteration inside of the orthogonal similarity reduction of a matrix into semiseparable form.** In Section 2.1 we showed that the intermediate semiseparable matrices have as eigenvalues the Lanczos-Ritz values. This behavior is completely the same as the one observed in an orthogonal similarity transformation to reduce a symmetric matrix into a similar tridiagonal one. One might therefore doubt the usefulness of this reduction, because it costs $9n^2 + O(n)$ more, and has not yet any advantage over the tridiagonal approach. In this section we will prove that the extra cost of $9n^2+O(n)$ provides an extra interesting property for this reduction.

At each step of the algorithm introduced in Theorem 49, one more row (column) is added by means of orthogonal transformations to the set of the rows (columns) of the matrix already proportional to each other. In this section, using arguments similar to those considered in [**183, 184, 185, 187, 195**], we show that this algorithm can be interpreted as a kind of nested subspace iteration method [**91**], where the size of the vector subspace is increased by one and a change of coordinate system is made at each step of the algorithm. As a consequence, depending on the gaps between the eigenvalues, the semiseparable part of the matrix will converge to a block diagonal matrix, and the eigenvalues of these blocks converge to the largest eigenvalues in absolute value of the original symmetric matrix.

Given a matrix $A$ and an initial subspace $\mathsf{S}^{(0)}$, the subspace iteration method [**91**] can be described as follows

$$\mathsf{S}^{(k)} = A\mathsf{S}^{(k-1)}, \quad k = 1, 2, 3, \ldots$$

Under weak assumptions on $A$ and $\mathsf{S}^{(0)}$, the $\mathsf{S}^{(k)}$ converge to an invariant subspace (more details on these assumptions will be investigated in the next subsection, because the subspace iteration will interact with the Lanczos convergence behavior). We will see that the reduction algorithm from a symmetric to a semiseparable matrix can be interpreted as such a kind of subspace iteration, where the dimension of the subspace grows by one at each step of the algorithm. Let $A^{(1)} = Q_0^T A Q_0$. For the reduction algorithm as presented in Section 1 in Chapter 4 we have $Q_0 = I$. Suppose we have only performed the first orthogonal similarity transformations such that the rows (columns) $n$ and $n-1$ are already proportional (Let us capture all the necessary Givens and Householder transformations to go from matrix $A^{(k)}$ to matrix $A^{(k+1)}$ in one orthogonal matrix $Q_k$):

$$A^{(2)} = Q_1^T A^{(1)} Q_1, \tag{21}$$

where $A^{(2)}$ has the semiseparable structure in the rows (columns) $n$ and $n-1$ and $Q_1 = [q_1^{(1)}, \ldots, q_n^{(1)}]$. From (21), we can write

$$\begin{aligned} A^{(1)} &= Q_1 \left(A^{(2)} Q_1^T\right) \\ &= Q_1 \begin{pmatrix} \times & \cdots & \times & 0 \\ \vdots & \ddots & \vdots & \vdots \\ \times & \cdots & \times & 0 \\ \times & \cdots & \times & \times \end{pmatrix} \\ &= Q_1 L_1. \end{aligned} \tag{22}$$

Let $e_1, \ldots, e_n$ be the standard basis vectors of $\mathbb{R}^n$. From (22), because of the structure of $L_1$, it can clearly be seen that:

$$A^{(1)}\langle e_n \rangle = \langle q_n^{(1)} \rangle.$$

This means that the last column of $A^{(1)}$ and $q_n^{(1)}$ span the same one–dimensional space. In fact one subspace iteration step is performed on the vector $e_n$. The first step of the algorithm is completed when the following orthogonal transformation is performed,

$$A^{(2)} = Q_1^T A^{(1)} Q_1.$$

The latter transformation can be interpreted as a change of coordinate system: $A^{(1)}$ and $A^{(2)}$ represent the same linear transformation with respect to different coordinate systems. Let $y \in \mathbb{R}^n$. Then $y$ is represented in the new system by $Q_1^T y$. This means that for the vector $q_n^{(1)}$ we get $Q_1^T q_n^{(1)} = e_n$. Summarizing, this means that one step of subspace iteration on the subspace $\langle e_n \rangle$ is performed, resulting in a new subspace $\langle q_n^{(1)} \rangle$, and then, by means of a coordinate transformation, it is transformed back into the subspace $\langle e_n \rangle$. So, instead of working with a fixed matrix and changing subspaces, we work with fixed subspaces and changing matrices. Therefore, denoting by $z^{(k)}$ the eigenvector corresponding to the largest eigenvalue in absolute value $\lambda$ of $A^{(k)}$, $k = 1, 2, \ldots$, we can say that, if $z^{(k)}$ has a nonzero last component and if the largest eigenvalue is unique, the sequence $\left\{z^{(k)}\right\}$ converges to $e_n$, and, consequently, the lower right element of $A^{(k)}$ converges to $\lambda$. Note that the last assumption of the nonzero component, will play an important role in the next section, where the interaction between the Lanczos behavior and the subspace iteration is investigated.

The second step can be interpreted in a completely analogous way. Suppose we have already the semiseparable structure in the last two rows (columns). Then we perform the following similarity transformation on $A^{(2)}$,

$$A^{(3)} = Q_2^T A^{(2)} Q_2, \tag{23}$$

in order to make the rows (columns) $n$ up to $n-2$ dependent. Using (23), $A^{(2)}$ can be written as follows,

$$\begin{aligned} A^{(2)} &= Q_2 \begin{pmatrix} \times & \cdots & \times & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \times & \cdots & \times & 0 & 0 \\ \times & \cdots & \times & \times & 0 \\ \times & \cdots & \times & \times & \times \end{pmatrix} \\ &= Q_2 L_2. \end{aligned}$$

Considering the subspace $\langle e_{n-1}, e_n \rangle$ and using the same notation as above, we have,

$$A^{(2)} \langle e_{n-1}, e_n \rangle = \langle q_{n-1}^{(2)}, q_n^{(2)} \rangle.$$

This means that the second step of the algorithm is a step of subspace iteration performed on a larger subspace. For every new dependency that is created in the symmetric matrix $A^{(k)}$, the dimension of the subspace is increased by one.

One can also see the algorithm as performing in each iteration step a $QL$ step without shift on the semiseparable bottom-right submatrix. Let us partition the matrix $A_0^{(k)}$ (as in the proof of Theorem 49) as follows:

$$A_0^{(k)} = A^{(k)} = \left( \begin{array}{c|c} A_k & u_k v_k^T \\ \hline v_k u_k^T & S_k \end{array} \right),$$

with $A_k \in \mathbb{R}^{(n-k)\times(n-k)}$, $u_k \in \mathbb{R}^{(n-k)\times 1}$, $v_k \in \mathbb{R}^{k\times 1}$and $S_k \in \mathbb{R}^{k\times k}$. From the semiseparable structure, we know that $[v_k, S_k e_1]$ has rank one. In the first part of the $k$th iteration step, an $(n-k)\times(n-k)$ orthogonal matrix $Q_{k,1}$ is constructed such that

$$Q_{k,1}^T u_k = [0, 0, \ldots, 0, \eta]^T$$

with $\eta = \pm\|u_k\|$. Hence, our matrix $A^{(k)}$ is transformed into the following orthogonally similar matrix

$$
\left(\begin{array}{c|c} Q_{k,1}^T & 0 \\ \hline 0 & I_k \end{array}\right)\left(\begin{array}{c|c} A_k & u_k v_k^T \\ \hline v_k u_k^T & S_k \end{array}\right)\left(\begin{array}{c|c} Q_{k,1} & 0 \\ \hline 0 & I_k \end{array}\right) \tag{24}
$$
$$
= \left(\begin{array}{c|c} \tilde{A}_k & \eta e_{n-k} v_k^T \\ \hline \eta v_k e_{n-k}^T & S_k \end{array}\right).
$$

Let us partition the matrix in a slightly different way, adding one more column and row to the semiseparable part. Let

$$
\tilde{A}_k = \begin{pmatrix} A_{k+1} & a \\ a^T & \alpha \end{pmatrix},
$$

then we can define the semiseparable matrix $\tilde{S}_k$ as follows

$$
\tilde{S}_{k+1} = \begin{pmatrix} \alpha & \eta v_k^T \\ \eta v_k & S_k \end{pmatrix}.
$$

In the second part of the $k$th iteration step, we perform a $QL$ step on the semiseparable matrix $\tilde{S}_{k+1}$ of order $k+1$, i.e., we construct an orthogonal matrix $Q_{k,2}$ such that

$$
\tilde{S}_{k+1} = Q_{k,2} L_k,
$$

with $L_k$ lower triangular. Hence, the matrix of (24) is transformed into the orthogonal similar matrix

$$
\left(\begin{array}{c|c} I_{n-k-1} & 0 \\ \hline 0 & Q_{k,2}^T \end{array}\right)\left(\begin{array}{c|c} A_{k+1} & a e_1^T \\ \hline e_1 a^T & \tilde{S}_k \end{array}\right)\left(\begin{array}{c|c} I_{n-k-1} & 0 \\ \hline 0 & Q_{k,2} \end{array}\right)
$$
$$
= \left(\begin{array}{c|c} A_{k+1} & u_{k+1} v_{k+1}^T \\ \hline v_{k+1} u_{k+1}^T & S_{k+1} \end{array}\right).
$$

The execution of the two steps is illustrated in Figure 5.1. At the beginning of the iteration the last four rows and columns have already a semiseparable structure. In (1), a similarity transformation (either an Householder matrix or a product of Givens transformations) is considered in order to annihilate the entries indicated by $\otimes$. Due to the semiseparable structure, all the entries in the first four rows (colums) with column (row) indexes greater than 5 are annihilated, too. In steps (2), (3), (4) and (5), the Givens rotations are only applied to the left, transforming the $(5 \times 5)$ principal submatrix in the right–bottom corner to lower triangular form. To retrieve the semiseparable structure in the last 5 rows and columns, the transpose of the latter Givens rotation must be applied to the right.

This means that from step $i$, $i = 1, \ldots, n$, all the consecutive steps perform subspace iterations on the subspace of dimension $i$. From [**187**], we know that these consecutive iterations on subspaces tend to create block lower triangular matrices. Hence, for a symmetric matrix these are block diagonal. Furthermore, the process works for all the nested subspaces at the same time, and so the semiseparable part of the matrices $A^{(k)}$ generated by the proposed algorithm, becomes more and more

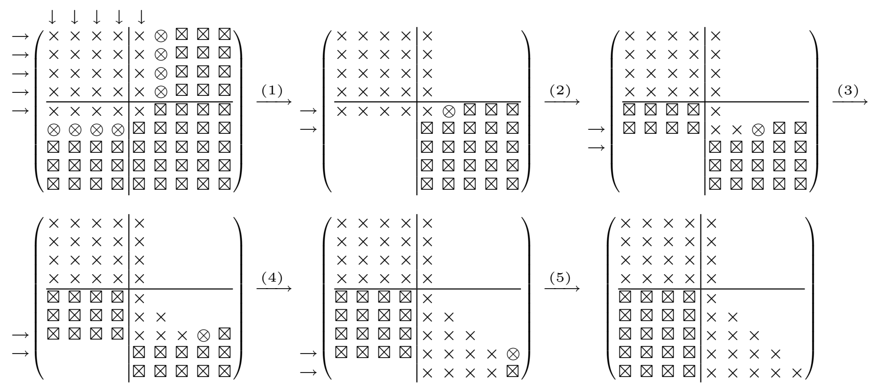

Figure 5.1: Description of $QL$ step without shift applied in each iteration of the semiseparable reduction.

block diagonal, and the blocks contain eigenvalues of the original matrix. This explains why in the numerical examples (see Chapter 6), the lower right block already gives a good estimate of the largest eigenvalues, since they are connected to a subspace on which the subspace iteration is performed most.

This insight also opens several new perspectives. In many problems, only the few largest eigenvalues (in absolute value) need to be computed [**22, 114, 181, 182, 192, 193**]. In such cases, the proposed algorithm gives the required information after only few steps, without running the algorithm to completion. Moreover, because the sequence of similar matrices generated at each step of the algorithm converges to a block diagonal matrix, the original problem can be divided into smaller independent subproblems. A direct application of this behavior can be found in the field of optical flow. In this field one wants to reduce the complexity of storing movies by not considering a movie as a sequence of images, but as pictures in which the pixels are moving. In this way one can associate a vector field to each pixel. Pixels which are moving have the largest component in the vector field. This means that moving objects in the image correspond to separate subspaces [**112**].

We know that at each step of the algorithm a step of $QL$ is performed on the semiseparable part of the matrix. This insight opens the perspective to a new research topic. Is it possible to incorporate a shift in this reduction, thereby performing a $QL$-step with shift on the already semiseparable part. This will however destroy the semiseparable structure, but it will reduce the matrix into a similar semiseparable plus diagonal matrix. Moreover it is possible to extract a complete diagonal matrix, not only a shift, from the semiseparable part and then perform the $QL$-step. In the following part we will prove that a $QL$-step applied on a semiseparable plus diagonal matrix is again a semiseparable plus diagonal matrix, for which the diagonal did not change. This last statement justifies our previous statements.

We finish this section with a theorem from [**187**] concerning the speed of convergence of subspace iterations.

Definition 56 (From [**187**, p. 8]). Denote with $\mathsf{S}$ and $\mathsf{T}$ two subspaces, then the distance $\mathrm{d}(\mathsf{S}, \mathsf{T})$ between these two subspaces is defined in the following way:

$$\mathrm{d}(\mathsf{S}, \mathsf{T}) = \sup_{s \in \mathsf{S}, \|s\|_2 = 1} \quad \inf_{t \in \mathsf{T}} \|s - t\|_2.$$

Using this definition, we can state the following convergence theorem:

Theorem 57 (From [**187**, Theorem 5.4]). *Let $A \in \mathbb{C}^{n \times n}$, and let $p$ be a polynomial of degree $\leq n$. Let $\lambda_1, \ldots, \lambda_n$ denote the eigenvalues of $A$, ordered so that $|p(\lambda_1)| \geq |p(\lambda_2)| \geq \cdots \geq |p(\lambda_n)|$. Suppose $k$ is a positive integer less than $n$ for which $|p(\lambda_k)| > |p(\lambda_{k+1})|$. Let $(p_i)$ be a sequence of polynomials of degree $\leq n$ such that $p_i \to p$ as $i \to \infty$ and $p_i(\lambda_j) \neq 0$ for $j = 1, \ldots, k$ and all $i$. Let $\rho = |p(\lambda_{k+1})|/|p(\lambda_k)|$. Let $\mathsf{T}$ and $\mathsf{U}$ be the invariant subspaces of $A$ associated with $\lambda_1, \ldots, \lambda_k$ and $\lambda_{k+1}, \ldots, \lambda_n$ respectively. Consider the nonstationary subspace iteration*

$$\mathsf{S}^{(i)} = p_i(A)\mathsf{S}^{(i-1)}$$

*where $\mathsf{S}^{(0)} = \mathsf{S}$ is a $k$-dimensional subspace of $\mathbb{C}^n$ satisfying $\mathsf{S} \cap \mathsf{U} = \{0\}$. Then for every $\hat{\rho}$ satisfying $\rho < \hat{\rho} < 1$ there exists a constant $\hat{C}$ such that*

$$\mathrm{d}\left(\mathsf{S}^{(i)}, \mathsf{T}\right) \leq \hat{C}\hat{\rho}^i, \quad i = 1, 2, 3, \ldots$$

In our case Theorem 57 can be applied with the polynomials $p_i(z) = p(z) = z$.

We translate the previous theorem to the matrix case, for which we apply the $QR$-algorithm without shift to the matrix $A^{(0)}$. With $A^{(k)} = Q_k R_k$ and $A^{(k+1)} = R_k Q_k$. We have that for

$$A^{(k)} = \begin{pmatrix} A_{11}^{(k)} & A_{12}^{(k)} \\ A_{21}^{(k)} & A_{22}^{(k)} \end{pmatrix},$$

there exists a $\hat{\rho}$, from Theorem 57, such that:

$$\|A_{21}^{(k)}\|_2 < C\hat{\rho}^k.$$

**2.3. The interaction between the subspace iteration and the Lanczos-Ritz values.** Taking a look at the numerical examples in Chapter 6, sometimes it seems that there is no sign of the convergence of the subspace iteration. The proofs stated above are correct, but there is an interaction with the Lanczos behavior of the algorithm, which will sometimes lead to a slower subspace iteration convergence than expected. This can only happen when, all the vectors $\langle e_{n-k}, e_{n-k+1}, \ldots, e_n \rangle$ have a small component in one or more directions of the eigenspace connected to the dominant eigenvalues. Before we show by examples, that this happens in practice, we first give a condition under which we are sure that the subspace iteration lets the matrix converge to one having a diagonal block containing the dominant eigenvalues. As soon as some of the Ritz values approximate all of the dominant eigenvalues, this convergence behavior appears. To show this we assume that the initial matrix $A$ has two clusters of eigenvalues, $\Lambda_1 = \{\lambda_{1,j} | j \in J\}$ and $\Lambda_2 = \{\lambda_{2,i} | i \in I\}$, with $\#J = n_1$ and $\#I = n_2$. Suppose also that there is a gap between cluster 2 and cluster 1,

$\min_{i\in I}|\lambda_{2,i}| \ggg \max_{j\in J}|\lambda_{1,i}|$. Using the following notations $\Delta_1 = \mathrm{diag}(\Lambda_1)$ and $\Delta_2 = \mathrm{diag}(\Lambda_2)$, we can write the matrix $A^{(k)}$ in the following way:

$$\begin{pmatrix} A_k & \times \\ \times & S_k \end{pmatrix} = \begin{pmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{pmatrix} \begin{pmatrix} \Delta_1 & 0 \\ 0 & \Delta_2 \end{pmatrix} \begin{pmatrix} V_{11}^T & V_{21}^T \\ V_{12}^T & V_{22}^T \end{pmatrix}, \tag{25}$$

with $S_k \in \mathbb{R}^{k\times k}$, $V_{11} \in \mathbb{R}^{(n-k)\times n_1}$, $V_{12} \in \mathbb{R}^{(n-k)\times n_2}$, $V_{21} \in \mathbb{R}^{k\times n_1}$ and $V_{22} \in \mathbb{R}^{k\times n_2}$. From equation (25), we have the following equation for $S_k$

$$S_k \quad = \quad V_{21}\Delta_1 V_{21}^T + V_{22}\Delta_2 V_{22}^T.$$

Let the eigenvalue decomposition of $S_k$ be denoted as:

$$S_k \quad = \quad \begin{pmatrix} V_1^{S_k} & V_2^{S_k} \end{pmatrix} \begin{pmatrix} \Delta_1^{S_k} & 0 \\ 0 & \Delta_2^{S_k} \end{pmatrix} \begin{pmatrix} V_1^{S_k T} \\ V_2^{S_k T} \end{pmatrix} \tag{26}$$

where we assume that some of the eigenvalues of $S_k$ (in fact the Ritz values) approximate already the dominant eigenvalues $\Lambda_2$. Let us denote $\Delta_2^{S_k} = \mathrm{diag}\left(\Lambda_2^{S_k}\right)$ with $\Lambda_2^{S_k}$ a set of eigenvalues $\Lambda_2^{S_k} = \{\lambda_{2,i}^{S_k} | \forall i \in I\}$, then we assume that

$$\lambda_{2,i}^{S_k} \approx \lambda_{2,i} \quad \forall i \in I.$$

subspace iteration needs weak conditions before convergence can occur: $V_{22}$ has full rank and is far from being not of full rank. This corresponds to the demand that the last vectors $\{e_{n-k+1},\ldots,e_n\}$ projected on the invariant subspace connected to the dominant eigenvalues $\lambda_{2,i}$ have a large component in every direction of this subspace.

Via an Householder transformation we can transform equation (25) into

$$\left(\begin{array}{cc|c} \multicolumn{2}{c|}{\tilde{A}} & \begin{matrix} 0 \\ \tilde{v}_k^T \end{matrix} \\ \hline 0 & \tilde{v}_k & S_k \end{array}\right) = \begin{pmatrix} \tilde{V}_{11} & \tilde{V}_{12} \\ V_{21} & V_{22} \end{pmatrix} \begin{pmatrix} \Delta_1 & 0 \\ 0 & \Delta_2 \end{pmatrix} \begin{pmatrix} \tilde{V}_{11}^T & V_{21}^T \\ \tilde{V}_{12}^T & V_{22}^T \end{pmatrix}.$$

Then we can substitute $S_k$ by its eigenvalue decomposition (26):

$$\left(\begin{array}{cc|cc} \multicolumn{2}{c|}{\tilde{A}} & \multicolumn{2}{c}{\begin{matrix} 0 \\ \hat{v}_k^T \end{matrix}} \\ \hline 0 & \hat{v}_k & \begin{matrix} \Delta_1^{S_k} \\ 0 \end{matrix} & \begin{matrix} 0 \\ \Delta_2^{S_k} \end{matrix} \end{array}\right) = \begin{pmatrix} \tilde{V}_{11} & \tilde{V}_{12} \\ \tilde{V}_{21} & \tilde{V}_{22} \end{pmatrix} \begin{pmatrix} \Delta_1 & 0 \\ 0 & \Delta_2 \end{pmatrix} \begin{pmatrix} \tilde{V}_{11}^T & \tilde{V}_{21}^T \\ \tilde{V}_{12}^T & \tilde{V}_{22}^T \end{pmatrix}.$$

Note that $V_{22}$ is of full rank, if and only if $\tilde{V}_{22}$ is of full rank. We get

$$\Delta_2^{S_k} = \begin{pmatrix} P\tilde{V}_{21}\Delta_1 & P\tilde{V}_{22}\Delta_2 \end{pmatrix} \begin{pmatrix} \tilde{V}_{12}^T P^T \\ \tilde{V}_{22}^T P^T \end{pmatrix},$$

where $P$ is the projection matrix $(0\ I)$ of size $n_2^{S_k} \times n$ where $I$ is the identity matrix of size $n_2^{S_k} \times n_2^{S_k}$, with $n_2^{S_k}$ the dimension of $\Delta_2^{S_k}$. This means that:

$$\Delta_2^{S_k} - P\tilde{V}_{12}\Delta_1\tilde{V}_{12}^T P^T = P\tilde{V}_{22}\Delta_2\tilde{V}_{22}^T P^T. \tag{27}$$

Note that $P\tilde{V}_{22}$ is a square matrix.

Because there is a gap between the eigenvalues $\lambda_{1,j}, j \in J$ and $\lambda_{2,i}, i \in I$, there will be a comparable gap between $\lambda_{2,j}^{S_k}, j \in J$ and $\lambda_{1,i}, i \in I$. Hence the matrix at the left-hand side of equation (27) is far from singular. Therefore this is also true for the right-hand side, and $\tilde{V}_{22}$ is of full rank. Hence, $V_{22}$ has full rank.

In this paragraph, a brief explanation is given, why in certain situations the subspace iteration does not work immediately from the start. Suppose the size of the block $\Delta_2$ is 2. We can write the matrix $A^{(1)} = Q_0^T A Q_0$ in decomposed form:

$$A^{(1)} = \left(\begin{array}{cc} A_1 & v_n \\ v_n^T & \alpha \end{array}\right) = \left(\begin{array}{cc} V_{11} & V_{12} \\ V_{21} & V_{22} \end{array}\right)\left(\begin{array}{cc} \Delta_1 & 0 \\ 0 & \Delta_2 \end{array}\right)\left(\begin{array}{cc} V_{11}^T & V_{21}^T \\ V_{12}^T & V_{22}^T \end{array}\right). \tag{28}$$

Note that $\alpha$ equals $S_1$. Applying already the Householder transformation on the matrix $A$ gives us the following decomposition:

$$\begin{aligned} &\left(\begin{array}{cc} \tilde{H} & 0 \\ q^T & 0 \\ 0 & 1 \end{array}\right)\left(\begin{array}{cc} V_{11} & V_{12} \\ V_{21} & V_{22} \end{array}\right)\left(\begin{array}{cc} \Delta_1 & 0 \\ 0 & \Delta_2 \end{array}\right)\left(\begin{array}{cc} V_{11}^T & V_{21}^T \\ V_{12}^T & V_{22}^T \end{array}\right)\left(\begin{array}{ccc} \tilde{H}^T & q & 0 \\ 0 & 0 & 1 \end{array}\right) \\ &= \left(\begin{array}{ccc} \tilde{A} & a^T & 0 \\ a & \gamma & \beta \\ 0 & \beta & \alpha \end{array}\right) = \tilde{A}^{(1)}. \end{aligned} \tag{29}$$

The Ritz values are the eigenvalues of the two by two matrix:

$$\left(\begin{array}{cc} \gamma & \beta \\ \beta & \alpha \end{array}\right).$$

In fact, to complete the previous step a Givens transformation should still be performed on the matrix $\tilde{A}^{(1)}$, to make the last two rows and columns dependent, and thereby transforming the matrix into $A^{(2)}$. Note that this last Givens transformation does not change the eigenvalues of the bottom right submatrix.

We show now that the weak assumptions to let the subspace iteration converge, are not satisfied. We prove that

$$(e_{n-1} \;\; e_n)$$

does not have two large linearly independent components in the span of

$$\left(\begin{array}{cc} \tilde{H} & 0 \\ q^T & 0 \\ 0 & 1 \end{array}\right)\left(\begin{array}{c} V_{12} \\ V_{22} \end{array}\right).$$

We can write $(e_{n-1} \;\; e_n)$ as a linear combination of the eigenvectors:

$$(e_{n-1} \;\; e_n) = \left(\begin{array}{cc} \tilde{H} & 0 \\ q^T & 0 \\ 0 & 1 \end{array}\right)\left(\begin{array}{cc} V_{11} & V_{12} \\ V_{21} & V_{22} \end{array}\right)\left(\begin{array}{c} C_1 \\ C_2 \end{array}\right).$$

The coordinates $C_2$ of $(e_{n-1}, e_n)$ with respect to the dominant eigenvectors, are the following:

$$\begin{aligned} C_2 &= \begin{pmatrix} V_{12}^T & V_{22}^T \end{pmatrix} \begin{pmatrix} \tilde{H}^T & q & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} e_{n-1} & e_n \end{pmatrix} \\ &= \begin{pmatrix} V_{12}^T & V_{22}^T \end{pmatrix} \begin{pmatrix} q & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} V_{12}^T q & V_{22}^T \end{pmatrix}. \end{aligned}$$

Using equation (28) and (29), we get that

$$\begin{pmatrix} V_{11}^T & V_{21}^T \\ V_{12}^T & V_{22}^T \end{pmatrix} \begin{pmatrix} v_n \\ \alpha \end{pmatrix} = \begin{pmatrix} \Delta_1 & 0 \\ 0 & \Delta_2 \end{pmatrix} \begin{pmatrix} V_{21}^T \\ V_{22}^T \end{pmatrix}.$$

Hence,

$$V_{12}^T v_n + V_{22}^T \alpha = \Delta_2 V_{22}^T.$$

Because of the Householder transformation we know that $q = v_n / \|v_n\|$, hence

$$V_{12}^T q = (\Delta_2 - \alpha I) \frac{V_{22}^T}{\|v_n\|}.$$

Therefore,

$$\begin{aligned} C_2 &= \begin{pmatrix} V_{12}^T q & V_{22}^T \end{pmatrix} \\ &= \begin{pmatrix} (\Delta_2 - \alpha I) \frac{V_{22}^T}{\|v_n\|} & V_{22}^T \end{pmatrix}. \end{aligned}$$

This means that if $\Delta_2 V_{22}^T$ lies almost in the same direction as $V_{22}^T$ then $C_2$ is almost singular. If the two largest eigenvalues $\lambda_{2,1} \approx \lambda_{2,2}$ this is the case. Note however that when $\lambda_{2,1} \approx -\lambda_{2,2}$, the subspace iteration will work immediately, because they are both extreme and immediately located by the Lanczos procedure.

Briefly spoken we have the following interaction between the Lanczos-Ritz value behavior, and the subspace iteration: the subspace iteration, will start converging as soon as the Lanczos-Ritz values have converged close enough to the dominant eigenvalues.

## 3. Convergence properties of the similarity reduction to a Hessenberg-like matrix

The results presented in Section 2, are directly applicable here. The performed similarity transformation is exactly the same, and nothing changes in the previous proofs, except that the Ritz values are not named the Lanczos-Ritz values anymore but the Arnoldi-Ritz values.

## 4. Convergence properties of an orthogonal transformation of a matrix to upper triangular semiseparable form

The theorems provided in Section 1, and the proof of the subspace iteration as given in Section 2.2 cannot directly be applied to the reduction of a matrix to upper triangular semiseparable form, simply because the latter transformation is not a similarity transformation. However, it is possible to reformulate all the proofs and formulas above to provide the same type of convergence behavior for this reduction. Doing so would cost a lot of effort. It is more convenient to provide a theorem which directly links the reduction to an upper triangular semiseparable form with the similarity reduction to symmetric semiseparable form. Providing such a theorem would make the proof of the above convergence properties unnecessary, as they can be seen as a direct consequence of this theorem.

THEOREM 58. *Suppose a matrix $A \in \mathbb{R}^{m \times n}$ is given. We reduce this matrix using Theorem 51 to upper triangular semiseparable form. The intermediate matrices are of the following form:*

$$A^{(k)} = U_{0:k}^T A V_{0:k}.$$

*(Note that the matrices $U_{0:k}$ and $V_{0:k}$ are a combination of several orthogonal transformations from Theorem 51). The matrix $A^{(k)}$ has the upper $k$ rows already of upper triangular semiseparable form. Let us denote this $k \times n$ matrix with $S_k^u$.*

*If we define the matrix $\tilde{A} = AA^T$, then we have that the matrix*

$$U_{0:k}^T \tilde{A} U_{0:k} = \tilde{A}^{(k)},$$

*will have the upper left $k \times k$ block $\tilde{S}_k$ of semiseparable form. Moreover we have that*

$$\tilde{S}_k = S_k^u S_k^{uT}.$$

PROOF. The proof is straightforward, when one combines Theorem 49, Theorem 51 and the note after Theorem 55. One can clearly see that the matrix $U_{0:k}$ constructed as in Theorem 51 will annihilate exactly the same elements as the orthogonal matrices needed to reduce a symmetric matrix into a similar semiseparable one, which has the upper left block of semiseparable form. □

Using this theorem will immediately reveal the interesting convergence properties connected with the reduction of a matrix to upper triangular semiseparable form.

**4.1. The Lanczos-Ritz values appearing in a reduction to an upper triangular semiseparable matrix.** We know from the reduction algorithm as presented in Theorem 51, that the intermediate matrices $A^{(k)}$, have the upper $k \times n$ matrix $S_k^u$ of upper triangular semiseparable form. Using the relations provided in Theorem 58, we know that the singular values of this matrix $S_k^u$ are the square roots of the eigenvalues of the matrix $\tilde{S}_k = S_k^u S_k^{uT}$, and the eigenvalues of the matrix $\tilde{S}_k$ as can be seen when combining Theorem 58 and the results from Section 2.1 are the Lanczos-Ritz values.

**4.2. subspace iteration in a reduction to an upper triangular semiseparable matrix.** In a similar way as in Section 2.2 one can prove that on the already semiseparable part of the matrix $A^{(k)}$ some kind of nested subspace iteration is performed. The proof is a straightforward extension of the results presented in 2.2, and is therefore not included. The interested reader can however find this proof in [**179**]. As a consequence of this subspace iteration, the upper triangular blocks along the main diagonal in the part of the matrix already semiseparable, give information on the largest singular values of the matrix.

## Conclusions

In this chapter we investigated the convergence behavior of the reduction algorithms in detail. A general framework was provided, which enables us to classify similarity transformations. Two easy to check conditions should be placed on such a similarity transformation in order to have the Arnoldi-Ritz values in the already reduced part of the matrix. We showed that the tridiagonalization and the reduction to semiseparable form satisfy the desired properties, and therefore also have the predicted convergence behavior.

Moreover we indicated in this chapter that the reduction to semiseparable form has an additional convergence behavior with respect to the tridiagonalization. It was proven that during the reduction to semiseparable form, some kind of nested subspace iteration is performed. This convergence behavior clearly interacts with the convergence towards the Arnoldi-Ritz values. The subspace iteration starts converging as soon as the Arnoldi-Ritz values approximate the dominant eigenvalues of the original matrix well enough.

CHAPTER 6

# Implementation of the algorithm and numerical experiments

In Chapter 4 of this thesis we proposed several algorithms to reduce matrices to semiseparable form. To calculate the computational complexity of these algorithms we always assumed that the part of the matrix already in semiseparable form, could be represented in an efficient way. This efficient representation, and the corresponding implementation will be given in this chapter. In Chapter 5 we investigated in detail the interaction between the subspace iteration and the Lanczos convergence behavior of the proposed reduction algorithms, in this chapter several numerical examples will illustrate these theoretical investigations.

The first three sections of this chapter give the mathematical details about the implementations of the different reduction algorithms. Three different types of reduction algorithms are considered: the reduction to symmetric semiseparable form; the reduction to Hessenberg-like form; the reduction to upper triangular semiseparable form. For each of these implementations the part of the matrix already in semiseparable form is represented using the Givens-vector representation.

In Section 4, four different experiments are performed to illustrate the interaction between the subspace iteration and the Lanczos convergence behavior. The experiments state the theoretical investigations of Chapter 5.

Section 5 illustrates the accuracy of the reduction to semiseparable form with respect to the accuracy of the reduction to tridiagonal form. Numerical experiments illustrate that the reduction to semiseparable form is more accurate for this example.

The last section of this chapter shows a numerical experiment in which the deflation possibilities of the reduction to semiseparable form are investigated. A matrix, which has three clusters of eigenvalues is created, and the reduction algorithm is applied to it. The results show, that while the reduction algorithm is still in progress, the three clusters are revealed by the algorithm. This feature is not shared by the reduction to tridiagonal form.

## 1. The orthogonal similarity reduction to symmetric semiseparable form

First we describe how to implement the orthogonal similarity reduction of a symmetric matrix into semiseparable form, using the Givens-vector representation of a semiseparable matrix. This representation gives information which can directly be used in the $QR$-algorithm applied to the resulting semiseparable matrix, as will be shown in Chapter 9. The implementation involves detailed shuffling of indices. Therefore only the mathematical details behind the implementation are given. The

MATLAB-files can be downloaded from `http://www.cs.kuleuven.ac.be/~marc/software`.

Suppose we are at the beginning of step $k = n - j$ in the reduction algorithm, this means that the rows $j+1, j+2, \ldots, n$ (columns $j+1, j+2, \ldots, n$) are already semiseparable. We can represent this semiseparable part by a row-vector $R_j \in \mathbb{R}^{1\times j}$, Givens transformations

$$\begin{pmatrix} c_i & -s_i \\ s_i & c_i \end{pmatrix}, \qquad i = j+1, j+2, \ldots, n-1$$

and a vector $[d_{j+1}, d_{j+2}, \ldots, d_n]$. At this point, the matrix $A^{(k)}$ similar to the original symmetric matrix $A$ can be divided into four blocks:

$$\left(\begin{array}{cc|ccc} A_k & c_{j+1}R_j^T & c_{j+2}s_{j+1}R_j^T & \cdots & s_{n-1}\ldots s_{j+1}R_j^T \\ c_{j+1}R_j & c_{j+1}d_{j+1} & c_{j+2}s_{j+1}d_{j+1} & \cdots & s_{n-1}\ldots s_{j+1}d_{j+1} \\ \hline c_{j+2}s_{j+1}R_j & c_{j+2}s_{j+1}d_{j+1} & c_{j+2}d_{j+2} & & \vdots \\ \vdots & \vdots & & \ddots & \\ s_{n-1}\ldots s_{j+1}R_j & s_{n-1}\ldots s_{j+1}d_{j+1} & \cdots & & d_n \end{array}\right) \tag{30}$$

with $A_k \in \mathbb{R}^{j\times j}$. In the actual implementation only the vector, the Givens transformations and the matrix

$$\begin{pmatrix} A_k & R_j^T \\ R_j & d_{j+1} \end{pmatrix}$$

are stored.

The first substeps in step $k$ of the method described in the proof of Theorem 49, eliminate the elements $1, 2, \ldots, j-1$ in row $j+1$ by multiplying $A_k$ to the right by the Givens transformations $G_1^{(k)}$, $G_2^{(k)}$, $\ldots$, $G_{j-1}^{(k)}$, i.e.,

$$R_j G_1^{(k)} G_2^{(k)} \cdots G_{j-1}^{(k)} = (0, \ldots, 0, \tilde{\alpha}_j) = \tilde{R}_{j+1}.$$

It is clear that we can obtain a similar result by applying a Householder transformation $H^{(k)}$ on the row $R_j$ such that the following equation is obtained:

$$R_j H^{(k)} = (0, \ldots, 0, \alpha_j) = \hat{R}_j,$$

with $|\alpha_j| = |\tilde{\alpha}_j|$. Performing the similarity Householder transformation on the matrix (30) transforms this matrix into the following one:

$$\begin{pmatrix} {H^{(k)}}^T A_k H^{(k)} & c_{j+1}\hat{R}_j^T & c_{j+2}s_{j+1}\hat{R}_j^T & \cdots & s_{n-1}\ldots s_{j+1}\hat{R}_j^T \\ c_{j+1}\hat{R}_j & c_{j+1}d_{j+1} & c_{j+2}s_{j+1}d_{j+1} & \cdots & s_{n-1}\ldots s_{j+1}d_{j+1} \\ c_{j+2}s_{j+1}\hat{R}_j & c_{j+2}s_{j+1}d_{j+1} & c_{j+2}d_{j+2} & & \vdots \\ \vdots & \vdots & & \ddots & \\ s_{n-1}\ldots s_{j+1}\hat{R}_j & s_{n-1}\ldots s_{j+1}d_{j+1} & \cdots & & d_n \end{pmatrix}. \tag{31}$$

From this point the Givens transformation $G_j^{(k)}$ can be calculated, such that the following equation is satisfied:

$$(\alpha_j, d_{j+1})\, G_j^{(k)} = (0, \alpha_{j+1}) \qquad \text{with} \qquad G_j^{(k)} = \begin{pmatrix} c_j & s_j \\ -s_j & c_j \end{pmatrix}.$$

The upper left part ${H^{(k)}}^T A_k H^{(k)}$ can be written as:

$$ {H^{(k)}}^T A_k H^{(k)} = \begin{pmatrix} A_{k-1} & R_{j-1} \\ R_{j-1} & \tilde{d}_j \end{pmatrix}. $$

We get the following matrix

$$ \begin{pmatrix} A_{k-1} & R_{j-1}^T & 0 & 0 & \cdots \\ R_{j-1} & \tilde{d}_j & c_{j+1}\alpha_j & c_{j+2}s_{j+1}\alpha_j & \cdots \\ 0 & c_{j+1}\alpha_j & c_{j+1}d_{j+1} & c_{j+2}s_{j+1}d_{j+1} & \cdots \\ 0 & c_{j+2}s_{j+1}\alpha_j & c_{j+2}s_{j+1}d_{j+1} & c_{j+2}d_{j+2} & \\ \vdots & \vdots & \vdots & & \ddots \end{pmatrix}, \tag{32} $$

after applying the similarity Givens transformation $G_j^{(k)}$ on the matrix (32) we get the following matrix:

(33)

$$ \begin{pmatrix} A_{k-1} & c_j R_{j-1}^T & s_j R_{j-1}^T & 0 & \cdots & 0 \\ c_j R_{j-1} & c_j \hat{d}_j & s_j \hat{d}_j & 0 & \cdots & 0 \\ s_j R_{j-1} & s_j \hat{d}_j & c_{j+1}\alpha_{j+1} & c_{j+2}s_{j+1}\alpha_{j+1} & \cdots & s_{n-1}\ldots s_{j+1}\alpha_{j+1} \\ 0 & 0 & c_{j+2}s_{j+1}\alpha_{j+1} & c_{j+2}d_{j+2} & & \\ \vdots & \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & s_{n-1}\ldots s_{j+1}\alpha_{j+1} & & \cdots & d_n \end{pmatrix}, $$

with $\hat{d}_j = \tilde{d}_j c_j + s_j c_{j+1}\alpha_j$. The Givens transformation ${G_j^{(k)}}^T$ and the element $\hat{d}_j$ can already be stored as part of the representation of the new semiseparable part which has to be formed at the bottom of the matrix. The process explained here can now be repeated to make all rows $j$, $j+1$ up to $n$ semiseparable, because the matrix (33) has essentially the same structure as the matrix (31). One can clearly see that the Givens transformations $G_j^{(k)}$ calculated here can immediately be stored to represent the lower right semiseparable block of the matrix.

NOTE. The latter step, i.e. the reduction of the lower right part to the semiseparable structure, can also be applied directly on a semiseparable matrix. This corresponds to performing a $QR$-step (in fact a $QL$-step) without shift on this semiseparable matrix (see Chapter 5, Section 2.2).

## 2. An orthogonal similarity reduction to a Hessenberg-like matrix

The implementation of this part is an easy, almost trivial extension of the implementation given in Section 1. Now attention is focussed on the lower triangular part. And all the Givens and Householder transformations are determined by the lower triangular part of the matrix.

We will describe the structure of an intermediate matrix in the reduction, when we represent this matrix with the Givens-vector representation.

Suppose we are at step $k$ of the reduction, with $j = n - k$. This means that we have a semiseparable part of dimension $k$ and a nonsemiseparable part of dimension

$j$. The intermediate matrix $A^{(k)}$has the following form:

$$
\left(\begin{array}{cc|cccc}
A_k & S_j^T & B_k & & & \\
c_{j+1}R_j & c_{j+1}d_{j+1} & & & & \\
\hline
c_{j+2}s_{j+1}R_j & c_{j+2}s_{j+1}d_{j+1} & c_{j+2}d_{j+2} & a_{12} & \cdots & a_{1,m-2} \\
\vdots & & & \ddots & & \vdots \\
 & & & & \ddots & \\
 & & & & c_{n-1}d_{n-1} & a_{m-2,m-2} \\
s_{n-1}\dots s_{j+1}R_j & & & \cdots & s_{n-1}d_{n-1} & d_n
\end{array}\right)
$$

with $S_j, R_j$ as row vectors of length $j$. The matrix $B_k$ is a $(j+1)\times(k-1)$ block. The entries $a_{ij}$ also have no special structure in general. One can see immediately that the structure of this matrix is completely similar to the one of equation (30), except the fact that symmetry is lost. Therefore we do not go into the details of this implementation anymore.

## 3. An orthogonal reduction to upper triangular semiseparable form

In this section only the reduction toards an upper triangular semiseparable form is described. All the other algorithms (as described in Theorem 52) can be deduced in an analogous way. Again we give some explicit formulas of the matrices. Based on these formulas the reader should be able to construct the algorithm.

Suppose we are at step $k$ of the algorithm, and we want to reduce an $m\times n$ matrix $A$ into upper triangular semiseparable form. (Note that for this implementation we start at the top of the matrix, while the reduction in the previous section started at the end.) This means that our matrix has a part of dimension $k\times n$ already of upper triangular semiseparable form, while a $j=n-k$ dimensional part is still not reduced. At the beginning of this step our matrix $A^{(k)}$ has the following form:

$$
\left(\begin{array}{cccc|cc}
d_n & s_{n-1}d_{n-1} & \cdots & & \cdots & s_{n-1}\cdots s_{j+1}R_j \\
0 & c_{n-1}d_{n-1} & & & & \vdots \\
\vdots & & \ddots & & & \\
0 & \cdots & 0 & c_{j+2}d_{j+2} & c_{j+2}s_{j+1}d_{j+1} & c_{j+2}s_{j+1}R_j \\
\hline
0 & \cdots & & 0 & c_{j+1}d_{j+1} & c_{j+1}R_j \\
\vdots & & & \vdots & & \\
0 & \cdots & & 0 & S_j^T & A_k
\end{array}\right) \tag{34}
$$

The matrix is partitioned in such a way that the upper left block is of dimension $(k-1)\times(k-1)$ and the lower right block is of dimension $(m-k+1)\times(n-k+1)$. The row vectors $S_j$ and $R_j$ are of length $(m-k)$ and $(n-k)$ respectively. The elements $d_i, c_i, s_i$ are defined in the same way as in the previous section. Equation (34) shows that the matrix $A^{(k)}$ has already the upper triangular part of the correct semiseparable form. In this step one wants to add one more row to the semiseparable structure, such that in the beginning of step $(k+1)$ we have a $(k+1)\times(k+1)$ upper triangular semiseparable matrix in the upper left $(k+1)\times(k+1)$ block. The first step consists of applying a Householder transformation or a sequence of Givens transformations to annihilate all the elements of the vector $R_j$ except for the first one. We choose to annihilate the elements with a Householder transformation $H_r^{(k)}$ (The $r$ denotes that we perform this transformation on the right of the matrix):

$$R_j H_r^{(k)} = (\alpha_j, 0, \ldots, 0) = \hat{R}_j.$$

Applying this transformation onto the matrix $A^{(k)}$ we get:

$$\left(\begin{array}{cccc|cc}
d_n & s_{n-1}d_{n-1} & \cdots & & \cdots & s_{n-1}\cdots s_{j+1}\hat{R}_j \\
0 & c_{n-1}d_{n-1} & & & & \vdots \\
\vdots & & \ddots & & & \\
0 & \cdots & 0 & c_{j+2}d_{j+2} & c_{j+2}s_{j+1}d_{j+1} & c_{j+2}s_{j+1}\hat{R}_j \\
\hline
0 & \cdots & & 0 & c_{j+1}d_{j+1} & c_{j+1}\hat{R}_j \\
\vdots & & & \vdots & & \\
0 & \cdots & & 0 & S_j^T & A_k H_r^{(k)}
\end{array}\right).$$

At this stage we are ready to annihilate the element $\alpha_j$ by applying a Givens transformation such that

$$(d_{j+1}, \alpha_j)\, G_{r,j}^{(k)} = (\alpha_{j+1}, 0)\,.$$

(Again the $r$ denotes that the Givens transformation is performed on the right-side of the matrix.) Note that the same result can be achieved when applying a Householder transformation $\tilde{H}_r^{(k)}$ such that

$$(d_{j+1}, R_j)\, \tilde{H}_r^{(k)} = (\tilde{\alpha}_{j+1}, 0, \ldots, 0)\,,$$

with $|\tilde{\alpha}_{j+1}| = |\alpha_{j+1}|$. After applying the Givens transformation $G_{r,j}^{(k)}$, our matrix has the following structure:

$$\left(\begin{array}{cccc|cc}
d_n & s_{n-1}d_{n-1} & \cdots & & s_{n-1}\cdots s_{j+1}\alpha_{j+1} & 0 \\
0 & c_{n-1}d_{n-1} & & & & \vdots \\
\vdots & & \ddots & & & \\
0 & \cdots & 0 & c_{j+2}d_{j+2} & c_{j+2}s_{j+1}\alpha_{j+1} & 0 \\
\hline
0 & \cdots & & 0 & c_{j+1}\alpha_{j+1} & 0 \\
\vdots & & & \vdots & & \\
0 & \cdots & & 0 & \hat{S}_j^T & \hat{A}_k
\end{array}\right). \tag{35}$$

The vector $S_j$ and the matrix $A_k H_r^{(k)}$ also did change after applying the Givens transformation. The new vector and matrix are denoted as $\hat{S}_j$ and $\hat{A}_k$. The next step consists of applying another Householder transformation on the left of the matrix (35) such that:

$$H_l^{(k)} S_j^T = (\beta_{j+1}, 0, \ldots, 0)^T\,.$$

(The $l$ denotes that the transformation will be performed on the left side of the matrix.) Applying this transformation on the matrix and rewriting $H_l^{(k)}\hat{A}_k$ as

$$H_l^{(k)}\hat{A}^{(j)} = \begin{pmatrix} d_j & R_{j-1} \\ S_{j-1} & A_{k-1} \end{pmatrix}$$

gives us

$$
\left(\begin{array}{cccc|ccc}
d_n & s_{n-1} & \dots & & s_{n-1}\cdots s_{j+1}\alpha_{j+1} & 0 & 0\\
0 & c_{n-1}d_{n-1} & & & \vdots & \vdots & \vdots\\
\vdots & & \ddots & & & & \\
0 & \dots & 0 & c_{j+2}d_{j+2} & c_{j+2}s_{j+1}\alpha_{j+1} & 0 & 0\\
\hline
0 & \dots & & 0 & c_{j+1}\alpha_{j+1} & 0 & 0\\
\vdots & & & \vdots & \beta_{j+1} & d_j & R_{j-1}\\
0 & \dots & & 0 & 0 & S_{j-1}^T & A_{k-1}
\end{array}\right). \tag{36}
$$

In the next step we will add one row to the semiseparable structure. Apply the Givens transformation $G_{l,j}^{(k)}$:

$$
G_{l,j}^{(k)} = \begin{pmatrix} c_j & s_j \\ -s_j & c_j \end{pmatrix}
$$

on the rows $k$ and $k+1$ such that $G_{l,j}^{(k)}\,(c_{j+1}\alpha_{j+1}, \beta_{j+1}) = \left(\tilde{d}_{j+1}, 0\right)^T$. This will give us the following matrix:

$$
\left(\begin{array}{cccc|ccc}
d_n & s_{n-1} & \dots & & s_{n-1}\cdots s_{j+1}\alpha_{j+1} & & 0\\
0 & c_{n-1}d_{n-1} & & & \vdots & & \vdots\\
\vdots & & \ddots & & & & \\
0 & \dots & 0 & c_{j+2}d_{j+2} & c_{j+2}s_{j+1}\alpha_{j+1} & & 0\\
\hline
0 & \dots & & 0 & \tilde{d}_{j+1} & s_j d_j & s_j R_{j-1}\\
\vdots & & & \vdots & 0 & c_j d_j & c_j R_{j-1}\\
 & & & & \vdots & & \\
0 & \dots & & 0 & 0 & S_{j-1}^T & A^{(j-1)}
\end{array}\right)
$$

Applying now the Givens transformation $G_{r,j+1}^{(k)}$ to the columns $k$ and $k+1$, in order to annihilate the upper part of column $k$, we get a matrix which is essentially the same as the matrix (36). One can continue and chase the complete structure upwards.

## 4. Numerical experiments: The interaction between the Lanczos and the subspace convergence behavior

The following three sections are based on the algorithm which transforms any symmetric matrix into a similar semiseparable one. Because the three reduction algorithms have essentially the same convergence behavior, there is no restriction in showing only numerical examples for the first algorithm. In the first experiment we want to illustrate the Lanczos convergence behavior of the orthogonal similarity reduction of a symmetric matrix to semiseparable form. Experiments two up to four illustrate the Lanczos behavior and the subspace iteration convergence. In Experiment 2 an example is created such that the subspace iteration has a large delay, and is not visible. After a clear delay in Experiment 3 the subspace iteration starts its convergence behavior. In Experiment 4 the example is created in such a way, that there is no delay in the convergence behavior of the subspace iteration.

In each experiment, we obtain a symmetric matrix $A$ by transforming the diagonal matrix $\Delta$ containing the prescribed eigenvalues by an orthogonal similarity transformation $A = Q^T \Delta Q$. The orthogonal matrix used is taken as the $Q$-factor in the $QR$ factorization of a random matrix, built by the MATLAB command $rand(n)$ where $n$ is the dimension of the matrix.

**4.1. Experiment 1.** We choose the eigenvalues as $\lambda_i = i$ for $i = 1, 2, \ldots, 200$. (These are equidistant eigenvalues.) In Figure 6.1 on the $y$-axis the eigenvalues are located. In each step of the algorithm ($x$-axis), a cross is placed if a Ritz value approximates a real eigenvalue up to 8 correct digits. This behavior is the same as the one described in [**125**, Section 4.2, Fig. 4.1], for equally spaced eigenvalues.

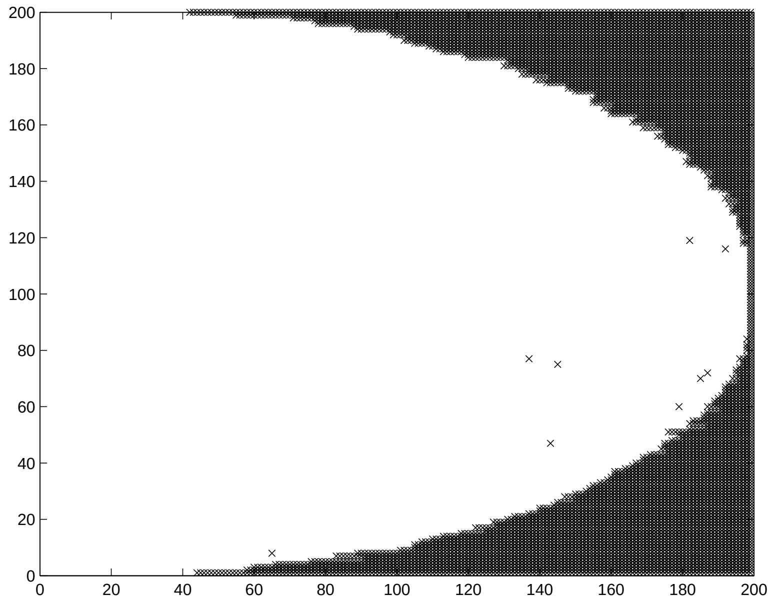


FIGURE 6.1: Equally spaced eigenvalues $1 : 200$

**4.2. Experiment 2.** We choose two clusters of equidistant eigenvalues, namely $[1 : 100, 1000 : 1099]$, each cluster has the same number of eigenvalues. The following convergence behavior of the Ritz values is computed (see Figure 6.2 left). Note however that the gap between the two intervals, does not appear in the following figure. For each $i = 1 : 199$ the norm of the block $S(i+1 : n, 1 : i)$ of the resulting semiseparable matrix is plotted. One would expect a small value for $i = 100$, because of the subspace iteration, but as explained in Section 2.2 this is not the case (see Figure 6.2 right).

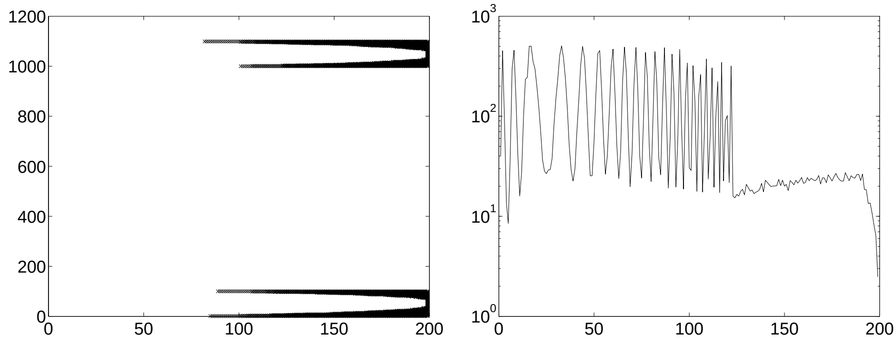


FIGURE 6.2: Equally spaced eigenvalues in two clusters $1 : 100$ and $1000 : 1099$

**4.3. Experiment 3.** In the previous experiment there was no sign of the influence of subspace iteration. In this experiment however we will clearly see the effect of the subspace iteration. For this example again two clusters of eigenvalues were chosen $[1 : 100]$ and $[1000 : 1009]$, one expects a clear view of the convergence of the subspace iteration in this case. Because the Lanczos-Ritz values approximate the extreme eigenvalues, it will at least take 20 steps before the 10 dominant eigenvalues are approximated. After these steps one can expect to see the convergence of the subspace iteration. The first figure (left of Figure 6.3) shows for each step $j = 1, 2, \ldots, n-1$ in the algorithm the norms of the blocks $S(i : n, 1 : i-1)$ for $i = n-j : n$, the lines correspond to one particular submatrix, i.e., the norm of this submatrix is shown after every step in the algorithm. The second figure (right of Figure 6.3) is constructed in an analogous way as in Experiment 4.2. In Figure 6.4 it can be seen in which step the Ritz values approximate the most extreme eigenvalues well enough, this is also the point from which the convergence behavior starts in Figure 6.3.

It is clearly seen in Figure 6.3 that the subspace iteration starts with a small delay (as soon as the Lanczos-Ritz values approximate the dominant eigenvalues well enough the convergence behavior starts).

**4.4. Experiment 4.** The previous experiment showed that the subspace iteration started working after a delay. In this experiment the largest eigenvalues in absolute value, have opposite signs, such that they will be located fast by the Lanczos algorithm and therefore the subspace iteration convergence will show up without a delay. The eigenvalues are located in three clusters $[-1004 : -1000, -100 : 100, 1000 : 1004]$. The Lanczos-Ritz values will converge fast to the dominant eigenvalues, and therefore the subspace iteration convergence will start fast. Figure 6.5 shows the fast convergence after few steps of the iteration and also the Lanczos convergence behavior.

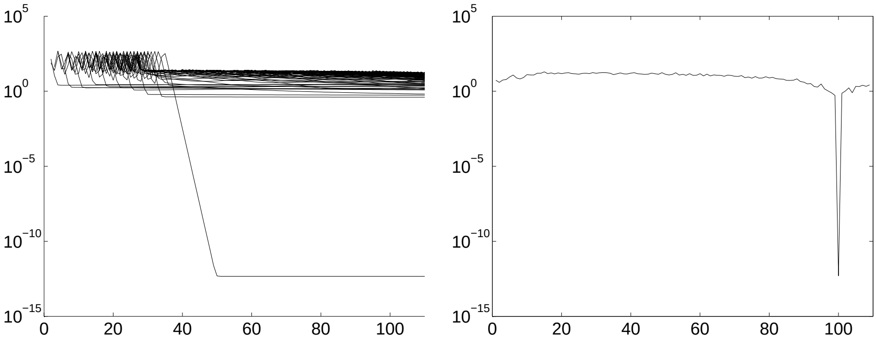


FIGURE 6.3: Equally spaced eigenvalues in two clusters 1 : 100 and 1000 : 1009

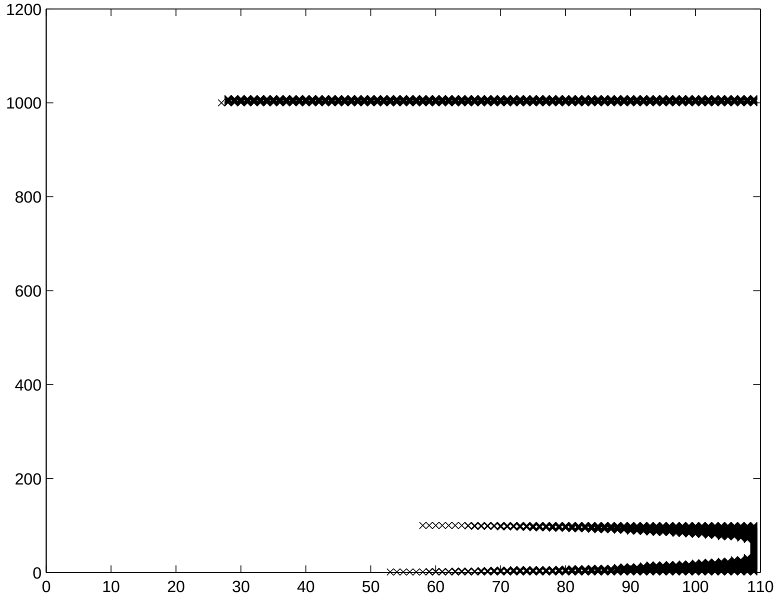


Figure 6.4: Lanczos behavior of equally spaced eigenvalues in two clusters 1 : 100 and 1000 : 1009

## 5. Numerical experiments: The accuracy of the reduction algorithm

In the following experiment a nongraded matrix with a large condition number ($\approx 10^{40}$) is transformed accurately into a semiseparable matrix. The following problems are taken from [**47**]. We consider some of the eigenvalue problems which the traditional $QR$-algorithm cannot solve. We take the following matrix: $A = DPD$,

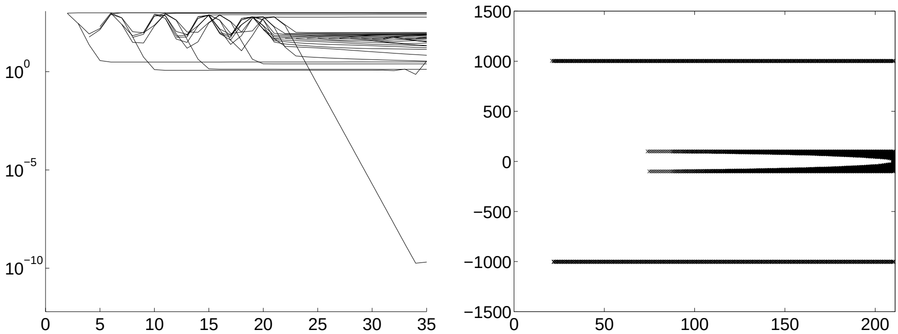


Figure 6.5: Equally spaced eigenvalues in three clusters $[-1004 : -1000, -100 : 100, 1000 : 1004]$

where $D = \mathrm{diag}(10^{20}, 10^{10}, 1)$,

$$A = \begin{pmatrix} 10^{40} & 10^{29} & 10^{19} \\ 10^{29} & 10^{20} & 10^{9} \\ 10^{19} & 10^{9} & 1 \end{pmatrix} \text{ and } P = \begin{pmatrix} 1 & 0.1 & 0.1 \\ 0.1 & 1 & 0.1 \\ 0.1 & 0.1 & 1 \end{pmatrix}.$$

The eigenvalues of the matrix $A$ are the following:

$\Lambda = [1.000000000000000 \cdot 10^{40}, 9.900000000000000 \cdot 10^{19}, 9.818181818181818 \cdot 10^{-1}]$.

The eigenvalues computed by the routine $eig(\cdot)$ in MATLAB are as follows:

$$[-3.85544 \cdot 10^{23}, 9.90002 \cdot 10^{-01}, 1.000000000000000 \cdot 10^{40}].$$

One can see that the eigenvalue solver of MATLAB, was only able to calculate one eigenvalue correctly. When we reduce the matrix $A$ to semiseparable form we get the following matrix:

$$\begin{pmatrix} 1.188000000000 \cdot 10^{2} & 1.080000000000 \cdot 10^{11} & 9.720000000000 \\ 1.080000000000 \cdot 10^{11} & 9.900000000000 \cdot 10^{19} & 8.910000000000 \cdot 10^{9} \\ 9.720000000000 & 8.910000000000 \cdot 10^{9} & 1.000000000000 \cdot 10^{40} \end{pmatrix}.$$

One eigenvalue already converged, and computing the eigenvalues of the upper left 2 by 2 block, we get the following eigenvalues:

$$\left[9.818181818181789 \cdot 10^{-1}, 9.900000000000000 \cdot 10^{19}\right].$$

So all the eigenvalues can be computed when first using the reduction to semiseparable form, and this up to at least 14 correct significant decimal digits.

Calculating the eigenvalues of the matrix $JAJ$ ($J$ is the counteridentity matrix, sometimes also called the exchange matrix) with

$$J = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix},$$

both algorithms give accurate results. The eigenvalue solver of Matlab gives the following result:

$$\left[9.818181818181818 \cdot 10^{-1}, 9.900000000000000 \cdot 10^{19}, 1.000000000000000 \cdot 10^{40}\right].$$

and the results based on the construction of the semiseparable matrix are exactly the same up to sixteen digits.

## 6. Numerical experiments: Deflation possibilities

In the following experiment we want to illustrate the interplay between the Lanczos convergence behavior and the nested subspace iteration of the new method. We want to focus attention on the deflation possibilities of the new algorithm.

Let us construct a symmetric matrix having the eigenvalues as shown in Figure 6.6. There are three clusters of eigenvalues where the relative gap between the

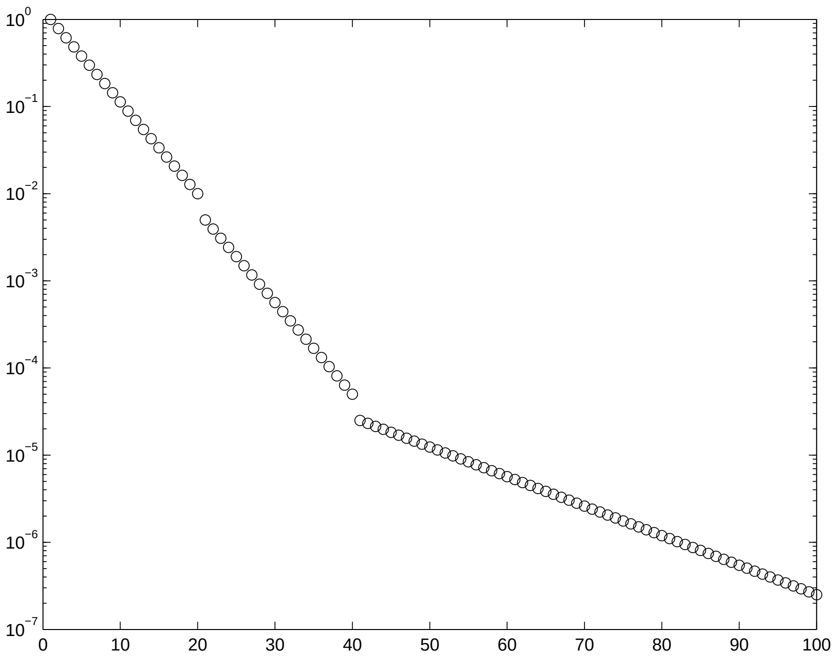


Figure 6.6: Eigenvalues of the symmetric matrix: 3 clusters

first and second cluster is equal to 0.5 and the same as the relative gap between the second and third cluster. The first cluster has 20 eigenvalues geometrically distributed between 1 and $10^{-2}$, the second cluster 20 eigenvalues between $5 \cdot 10^{-3}$ and $5 \cdot 10^{-5}$ and the third 60 eigenvalues between $2.5 \cdot 10^{-5}$ and $2.5 \cdot 10^{-7}$.

Let us first look at Figure 6.7 showing the behavior of the Ritz values for this example. Note that the same behavior is also obtained when using the classical tridiagonalization approach. However, the tridiagonal matrix obtained using this

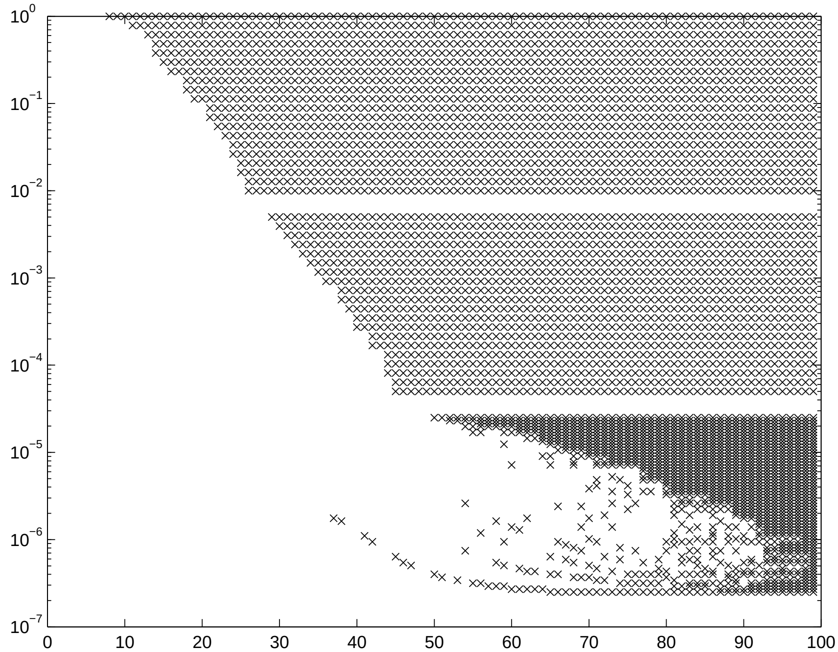


FIGURE 6.7: Behavior of the Ritz values

approach does not give a clear indication of the possible clusters or of deflation possibilities, i.e., no subdiagonal element becomes very small in magnitude. One might however chose a different initial transformation $Q_0$ (see Chapter 5, Section 1) to increase the Lanczos convergence behavior towards the last cluster, but one will not be able to recognize two clusters in the tridiagonal matrix. The magnitude of the subdiagonal elements is plotted in Figure 6.8. During the execution of the algorithm to obtain a similar semiseparable matrix, there will be a clear point where a diagonal block corresponding to the first cluster can be deflated and another point where a diagonal block corresponding to the second cluster can be deflated. Looking at Figure 6.7, we see that the eigenvalues of the first cluster are approximated well by the Lanczos-Ritz values from step 25 on. This means that at that moment the subspace iteration where the subspace has dimension 20 will converge towards the subspace corresponding to the first cluster. The convergence factor will be 0.5. This can clearly be seen in Figure 6.9. This figure shows for each step $j = 1, 2, \ldots, n-1$ in the algorithm the norms of the subdiagonal blocks $S(i : n, 1 : i-1)$ for $i = n-j : n$. Each line in the plot indicates the change of the norm at each step of the algorithm. The line indicated by "o" corresponds to the norm of the subdiagonal block $S(81 : 100, 1 : 80)$. For the eigenvalues of the second cluster a similar convergence behavior occurs around step 45. The line indicated by "*" corresponds to the norm of the subdiagonal block $S(61 : 100, 1 : 60)$. The parallel lines in the figure having a smaller

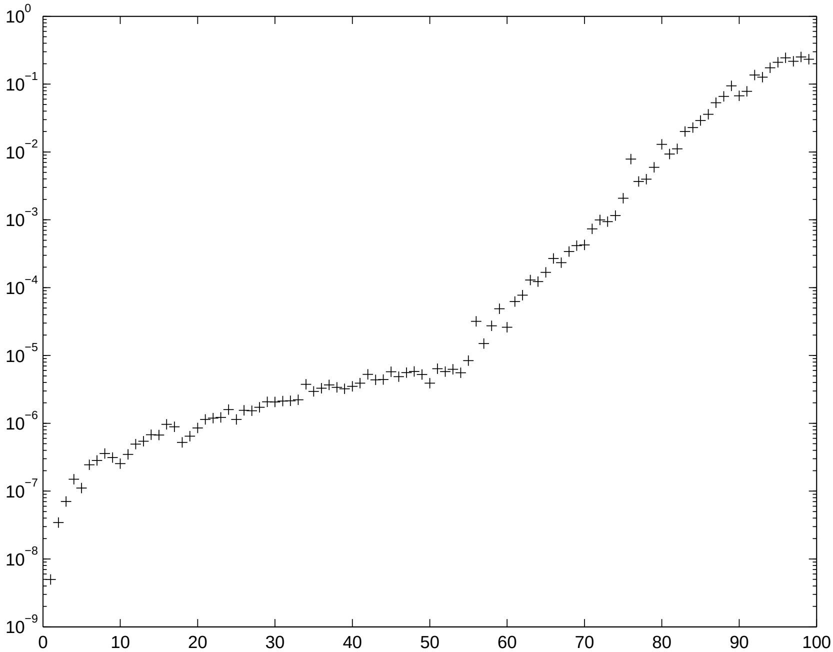


FIGURE 6.8: Magnitude of subdiagonal elements of similar tridiagonal matrix

slope correspond to the subspace convergence behavior inside each of the clusters. Hence, at step 68 a first diagonal $20 \times 20$ block can be deflated while at step 80 the next diagonal $20 \times 20$ block can be deflated. This example shows clearly the influence of the convergence of the Lanczos-Ritz values on the convergence behavior of the nested subspace iteration.

## Conclusions

In this chapter first of all the mathematical details behind the implementation of the three basic algorithms of this part were given. The reduction algorithms were implemented in an efficient way by representing the already reduced semiseparable part by the Givens-vector representation.

Furthermore, different experiments were performed showing some of the features of the reduction algorithms. First the interaction between the two convergence behaviors as explained in the previous chapter was illustrated. In Section 5 we investigated the accuracy of the reduction algorithm with respect to some difficult matrices. The experiments revealed that the reduction to semiseparable form seems to be more accurate than the reduction to tridiagonal form. In the final section of this chapter we investigated the deflation possibilities of the reduction algorithm. We showed that the subspace iteration allows us to deflate at more places as in the corresponding tridiagonal case.

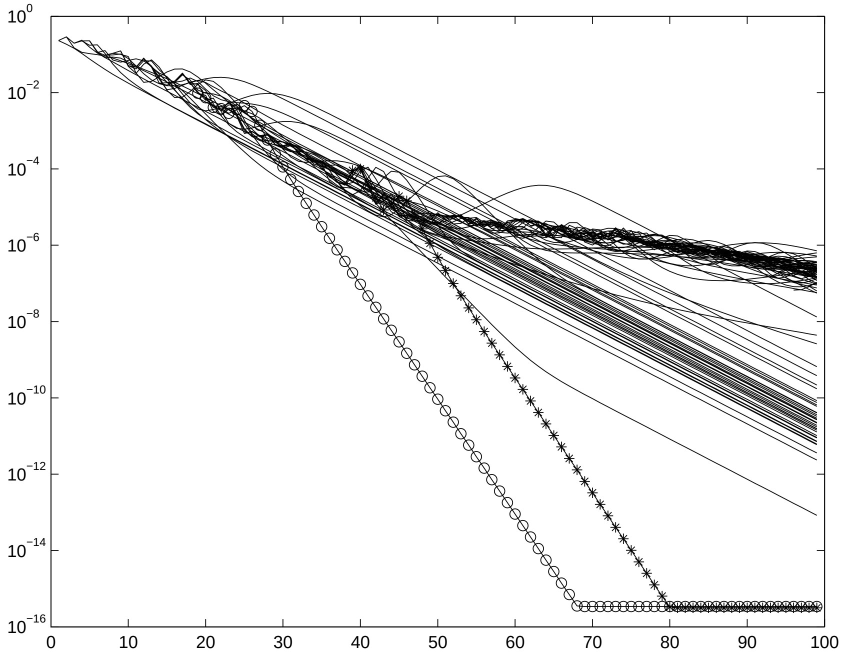


FIGURE 6.9: Norms of subdiagonal matrices during the semiseparable reduction

# Part 3

# QR-algorithms

CHAPTER 7

# Theoretical results needed to design implicit $QR$-algorithms for semiseparable matrices

In Chapter 8 of this thesis we will design different types of implicit $QR$-algorithms for matrices of semiseparable form. Before we are able to construct these algorithms some theoretical results are needed. In this chapter we will provide answers to the following problems for Hessenberg-like matrices. How can we calculate the $QR$-factorization of a Hessenberg-like plus diagonal matrix? What is the structure of an unreduced Hessenberg-like matrix and how can we transform a matrix to the unreduced form? Is the Hessenberg-like structure maintained after a step of $QR$ with shift? Does some kind of analogue of the implicit $Q$-theorem exist for Hessenberg-like matrices? The results in this chapter are applied to Hessenberg-like matrices, because this is the most general structure. In the following chapter, we will adapt, if it is necessary, the theorems to the other cases, namely the symmetric case and the upper triangular semiseparable case.

In a first section of this chapter we will investigate "a type" of $QR$-factorization of Hessenberg-like plus diagonal matrices. We say a type of because we will show in the next section, that the $QR$-factorization is not necessarily unique. The factorization we propose is based on the paper [**178**]. In the factorization presented in this paper, the orthogonal matrix $Q$ is constructed as a product of $2n-2$ Givens transformations, with $n$ the dimension of the matrix. Suppose we have a Hessenberg-like plus diagonal matrix $Z+D$. It will be shown that a first sequence of $n-1$ Givens transformations $Q_1$ will transform the matrix $Z$ into an upper triangular matrix $Q_1^T Z$. These Givens transformations will transform the matrix $D$ into a Hessenberg matrix. Our resulting matrix $Q_1^T(Z+D)$ will therefore be Hessenberg. The second sequence of Givens transformations will reduce the Hessenberg matrix to upper triangular form.

The concept of irreducibility is an important topic for the construction of implicit $QR$-algorithms. We define what is meant with an unreduced Hessenberg-like matrix. It will be shown later on that this unreducedness demand plays a very important role in the implicit $Q$-theorem and in the construction of the implicit $QR$-algorithm. Moreover, using the definition of an unreduced Hessenberg-like matrix, it is rather easy to prove the essential uniqueness of the $QR$-factorization for Hessenberg-like matrices. With essentially unique we mean that two $QR$-factorizations only differ for the sign of the elements.

In the third section we will investigate the structure of a Hessenberg-like matrix after having applied a step of the $QR$-algorithm with shift. First we show that the choice of the $QR$-factorization is essential to obtain again a Hessenberg-like matrix after a $QR$-step. This fact is illustrated by an example in the Hessenberg and in the Hessenberg-like case. Using the $QR$-factorization for Hessenberg-like matrices as explained in Section 1, we will prove the following theorems: The structure of a Hessenberg-like matrix is maintained after a step of $QR$ without shift; The weakly lower triangular structure of a matrix is maintained after a $QR$-step with shift; The Hessenberg-like structure is maintained after a $QR$-step with shift; The diagonal in a Hessenberg-like plus diagonal matrix is maintained after applying a step of $QR$ with shift. Note that for these theorems, no assumptions concerning the singularity of the matrices are made.

In the second section we defined unreduced Hessenberg-like matrices and in Section 5 we will prove an implicit $Q$-theorem for Hessenberg-like matrices. This implicit $Q$-theorem strongly depends on the unreducedness of the Hessenberg-like matrix. Therefore we will provide in Section 4 an easy way of transforming a Hessenberg-like matrix to unreduced form. The algorithm provided performs in fact a special $QR$-step without shift on the Hessenberg-like matrix. This results in an algorithm, revealing immediately the singularities of the corresponding Hessenberg-like matrix.

In Section 5 of this chapter we will provide an implicit $Q$-theorem for Hessenberg-like matrices. The theorem is quite similar to the theorem for Hessenberg matrices. It will prove to be very powerful in the next chapter when we start with the construction of the different $QR$-algorithms.

## 1. The $QR$-factorization of Hessenberg-like plus diagonal matrices

Before $QR$-algorithms can be introduced it is important to retrieve more information about the $QR$-factorization of Hessenberg-like plus diagonal matrices. We consider the class of Hessenberg-like plus diagonal matrices instead of the Hessenberg-like class, because we will deduce implicit $QR$-algorithms with a shift.

The $QR$-factorization we will propose here, is based on the $QR$-factorization of semiseparable plus diagonal matrices as described in [**178**]. First we will explain the algorithm for computing the $QR$-factorization of Hessenberg-like matrices. Then we will specify in more detail the $QR$-factorization of Hessenberg-like plus diagonal matrices. The computation of the $QR$-factorization of a Hessenberg-like matrix $Z$ is straightforward. Due to the structure of the matrix involved, an application of $n-1$ Givens rotations, $G_i^T, i = 1,\ldots,n-1$, applied from bottom to top ($G_1^T$ acts on the last two rows of the matrix, annihilating all the elements in the last row of $Z$ below the main diagonal, $G_2^T$ acts on the rows $n-2$ and $n-1$ of $G_1^T Z$, annihilating all the elements in row $n-1$ below the main diagonal, $G_{n-1}^T$ acts on the first two rows of $G_{n-2}^T \cdots G_1^T Z$, annihilating the element in position $(2,1)$) will reduce the Hessenberg-like matrix $Z$ into an upper triangular matrix $\tilde{R}$:

$$G_{n-1}^T \ldots G_2^T G_1^T Z = \tilde{R}.$$

NOTE. These Givens transformations are the transpose of the Givens transformations stored in the Givens-vector representation of the Hessenberg-like matrix.

Let us now calculate the $QR$-factorization of a Hessenberg-like plus diagonal matrix: $Z + D$. The $Q$ factor of this reduction consists of $2n - 2$ Givens transformations. The first $n - 1$ Givens transformations are performed on the rows of the Hessenberg-like matrix from bottom to top. These Givens transformations are exactly the same as the ones described in the previous paragraph. These Givens transformations transform the Hessenberg-like matrix into an upper triangular matrix. When taking into consideration the diagonal, one can see that these $n - 1$ Givens transformations, transform the diagonal part into a Hessenberg matrix.

$$G_{n-1}^T \dots G_2^T G_1^T (Z + D) = \tilde{R} + \tilde{H} = H.$$

Recombining the upper triangular matrix $\tilde{R}$ and the Hessenberg matrix $\tilde{H}$ gives another Hessenberg matrix $H$. This matrix will then be made upper triangular by another $n - 1$ Givens transformations $G_i^T, i = n, \dots, 2n - 2$, from top to bottom.

$$\begin{aligned} & G_{2n-2}^T \dots G_2^T G_1^T (Z + D) \\ = \;& G_{2n-2}^T \dots G_n^T \left(\tilde{R} + \tilde{H}\right) \\ = \;& R. \end{aligned}$$

NOTE. If the Givens transformation is not uniquely determined, this means that the Givens transformation has to be calculated based on the vector $(0, 0)^T$, then we choose the Givens transformation equal to the identity matrix.

When applying this technique to a semiseparable plus diagonal matrix, more general theorems concerning the structure, e.g., the structure of $R$, can be derived. These theorems can be found in [**178**]. In Chapter 9, an order $O(n)$ algorithm will be derived for implementing the $QR$-factorization of semiseparable plus diagonal matrices. Moreover this will be used to solve systems of equations with the coefficient matrix of semiseparable plus diagonal form. For the purpose of this chapter no more information concerning the structure of the $R$ and $Q$ factors is desired, only concerning the unicity of the $QR$-factorization.

## 2. Unreduced Hessenberg-like matrices

Unreduced Hessenberg matrices are very important for the development of an implicit $QR$-algorithm for Hessenberg matrices. It is essential for the application of an implicit $QR$-algorithm that the corresponding Hessenberg matrix is unreduced, otherwise the algorithm would break down. Another feature of an unreduced Hessenberg matrix $H$ is that it has an essentially unique $QR$-factorization. Moreover also $H - \kappa I$, with $\kappa$ a shift has an essentially unique $QR$-factorization. This is straightforward because of the zero structure below the subdiagonal. Only the last column can be dependent of the first $n-1$ columns. Essentially unique means, only the sign of the columns of the orthogonal matrices $Q$ can differ, and the signs of the elements in $R$. Because the previous $n - 1$ columns are linearly independent, the first $n-1$ columns of the matrix $Q$ in the $QR$-factorization of $H = QR$, are linearly independent. $Q$ is an orthogonal matrix, and the first $n - 1$ columns are essentially unique. As the dimension is $n$ and we already have $n - 1$ orthogonal columns, the last column is uniquely defined, orthogonal to the first $n - 1$ columns. This means that the $QR$-factorization is essentially unique. Summarizing, this means that the $QR$-factorization of an unreduced Hessenberg matrix $H$ has the following form:

$$H = Q \left( \begin{array}{c|c} R & w \\ \hline 0 & \alpha \end{array} \right),$$

for which $R$ is an upper triangular matrix of dimension $(n-1) \times (n-1)$, $w$ is a column vector of length $(n-1)$, and $\alpha$ is an element in $\mathbb{R}$. We have that $\alpha = 0$ if and only if $H$ is singular, this means that the last column of $H$ depends on the previous $n-1$.

Let us define now an unreduced Hessenberg-like matrix, in the following way:

DEFINITION 59. An Hessenberg-like matrix $Z$ is said to be unreduced if

(1) all the blocks $Z(i+1:n, 1:i)$ (for $i = 1, \ldots, n-1$) have rank equal to 1; this corresponds to the fact that there are no zero blocks below the diagonal;
(2) all the blocks $Z(i:n, 1:i+1)$ (for $i = 1, \ldots, n-1$) have rank strictly higher than 1, this means that on the superdiagonal, no elements are includable in the semiseparable structure.

NOTE. If an Hessenberg-like matrix $Z$ is unreduced it is also nonsingular. This can be seen by calculating the $QR$-factorization of $Z$ as described in Section 1 of this chapter. Because none of the elements above the diagonal is includable in the semiseparable structure below the diagonal, all the diagonal elements of the upper triangular matrix $R$ in the $QR$-factorization of $Z$ will be different from zero, implying the nonsingularity of the Hessenberg-like matrix $Z$.

When these demands are placed on the Hessenberg-like matrix, and they can be checked rather easily, we have the following theorem connected to the $QR$-factorization:

THEOREM 60. *Suppose $Z$ to be an unreduced Hessenberg-like matrix. Then the matrix $Z - \kappa I$, with $\kappa$ as a shift has an essentially unique $QR$-factorization.*

PROOF. If $\kappa = 0$ the theorem is true because $Z$ is nonsingular. Suppose $\kappa \neq 0$. Because the Hessenberg-like matrix $Z$ is unreduced, the first $n-1$ Givens transformations $G_i$ of Section 1 of this chapter are nontrivial. Applying them to the left of the matrix $Z - \kappa I$ results therefore in an unreducible Hessenberg matrix. As this Hessenberg matrix has the first $n-1$ columns independent of each other also the matrix $Z - \kappa I$ has the first $n-1$ columns independent. Hence, the $QR$-factorization of $Z - \kappa I$ is essentially unique. □

NOTE. An unreduced Hessenberg-like matrix $Z$ is always nonsingular, and has therefore, always an essentially unique $QR$-factorization. If one however admits that the block $Z(n-1:n, 1:n)$ is of rank 1, the matrix $Z$ also will have an essentially unique $QR$-factorization.

NOTE. The unreduced Hessenberg-like matrix $Z$ is always nonsingular, but the matrix $Z - \kappa I$ can be singular.

This demand of unreducedness will play an important role in the proof of the implicit $Q$-theorem for semiseparable matrices. Before we will formulate and prove this theorem we investigate another, also very interesting property of Hessenberg-like matrices. We will prove that the application of each step of the $QR$-algorithm, where the orthogonal matrix $Q$ is designed as in Section 1, will maintain the structure of the Hessenberg-like matrices.

## 3. A $QR$-step maintains the Hessenberg-like structure

In this section we will prove that a $QR$-step applied to a Hessenberg-like matrix is again a Hessenberg-like matrix. Mathematically spoken this means that for a Hessenberg-like matrix $Z$, with $\kappa$ a shift and

$$Z - \kappa I = QR$$

the $QR$-decomposition of $Z - \kappa I$ (with the $QR$-decomposition as described in Section 1), the matrix $\tilde{Z}$ satisfying

$$\tilde{Z} \quad = \quad RQ + \kappa I$$

is again a Hessenberg-like matrix. Because we want to place this in a general framework, we derive a theorem proving this, without putting any constraints on the Hessenberg-like matrix. However, we do need to place demands on the type of $QR$-factorization used. This is illustrated by the following example for Hessenberg matrices.

Example 26. For Hessenberg matrices, the type of $QR$-factorization is important to prove that the Hessenberg structure is maintained under one step of the $QR$-algorithm, as the example will show. Suppose we have a Hessenberg matrix of the following form:

$$H = \begin{pmatrix} \times & \times & \times & \times & \times \\ \otimes & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & \otimes & \times \end{pmatrix}. \tag{37}$$

We perform a sequence of Givens transformations on the Hessenberg matrix such that it becomes upper triangular $G_2^T G_1^T H = R$, for which $G_1$ and $G_2$ are Givens transformations and $R$ is an upper triangular matrix. In fact only the two elements marked with $\otimes$ in matrix (37) need to be annihilated. Therefore $G_1^T$ performs a Givens transformation on rows 3 and 5 to annihilate the element in position $(5, 4)$ and $G_2^T$ annihilates the element in position $(2, 1)$ by performing a Givens transformation on the first and second row. We can calculate now the matrix $RG_1G_2$. This corresponds to performing one step of $QR$ without shift to the Hessenberg matrix $H$. The matrix $RG_1G_2$ is of the following form:

$$RG_1G_2 = \begin{pmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & \times & 0 & \times \end{pmatrix}.$$

One can clearly see that the resulting matrix is not Hessenberg anymore. Therefore some assumptions have to be placed on the type of $QR$-factorization such that the resulting matrix after a step of the $QR$-algorithm is again Hessenberg. The $QR$-factorization presented in [**91**], consists of annihilating all the subdiagonal elements by a Givens transformation on the subdiagonal element and the diagonal element above it. When we use this type of $QR$-factorization to compute the $QR$-steps, we will always have again a Hessenberg matrix.

The type of $QR$-factorization we will use to compute the decomposition for Hessenberg-like matrices, is the one as presented in Section 1 of this chapter. The $Q$ factor of the $QR$-decomposition consists of two sequences of $n-1$ Givens transformations, let us define $Q_1 = G_1 \dots G_{n-1}$ and $Q_2 = G_n \dots G_{2n-2}$. This means that the $QR$-decomposition of a Hessenberg-like matrix minus a shift matrix has the following form:

$$Q_2^T Q_1^T (Z - \kappa I) \quad = \quad R$$

leading to

$$(Z - \kappa I) \quad = \quad Q_1 Q_2 R.$$

Completing the step of the $QR$-algorithm leads to the following equations

$$\begin{aligned} \tilde{Z} &= RQ_1Q_2 + \kappa I \\ &= Q_2^T Q_1^T Z Q_1 Q_2. \end{aligned}$$

NOTE. If the matrix $R$ is invertible, which is not always the case, then we have also the following equation:

$$\tilde{Z} = RZR^{-1}.$$

Using this equation one can easily prove that the Hessenberg-like structure is maintained. (See [**19**]). Moreover if the matrix $Z$ is invertible, one can directly switch, via inversion, to the Hessenberg case, which provides us with another easy proof for maintaining the Hessenberg-like structure.

In the remaining part of this section we will first prove that the Hessenberg-like structure is maintained when a step of $QR$ without shift is performed. Then we will prove theorems for $QR$ with shift applied on matrices having weakly lower triangular rank 1, Hessenberg-like matrices and finally for Hessenberg-like plus diagonal matrices.

THEOREM 61. *Suppose we have a Hessenberg-like matrix $Z$ and a step of the $QR$-algorithm (with the $QR$-decomposition as in Section 1) without shift is performed on the matrix $Z$:*

$$\begin{cases} Z &= QR \\ \tilde{Z} &= RQ. \end{cases}$$

*Then the matrix $Z$ will be a Hessenberg-like matrix.*

PROOF. Because no shift is involved only the first $n-1$ Givens transformations need to be performed on the Hessenberg-like matrix $Z$ to retrieve the upper triangular matrix $R$.

$$\begin{aligned} R &= Q_1^T Z \\ &= G_{n-1}^T \dots G_1^T Z. \end{aligned}$$

The Givens transformation $G_k^T$ performs an operation on rows $(n-k)$ and $(n-k+1)$. Therefore one can clearly see that the transformations $RG_1 \dots G_{n-1}$ will create a matrix $\tilde{Z}$ whose lower triangular rank is maximum 1. As an illustration we perform the transformations on a matrix of dimension 4, the elements marked with $\boxtimes$ denote the part of the matrix satisfying the semiseparable structure.

$$
\begin{array}{cccc}
R \; = & \left(\begin{array}{cccc} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times \end{array}\right) & \xrightarrow{RG_1} & \left(\begin{array}{cccc} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & 0 & \boxtimes & \times \\ 0 & 0 & \boxtimes & \boxtimes \end{array}\right) \\
\xrightarrow{RG_1G_2} & \left(\begin{array}{cccc} \times & \times & \times & \times \\ 0 & \boxtimes & \times & \times \\ 0 & \boxtimes & \boxtimes & \times \\ 0 & \boxtimes & \boxtimes & \boxtimes \end{array}\right) & \xrightarrow{RG_1G_2G_3} & \left(\begin{array}{cccc} \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes \end{array}\right)
\end{array}
$$

$\square$

Using this theorem, and the knowledge about the Givens transformations involved in the $QR$-factorization we can easily formulate a theorem which states that the structure of a matrix with weakly lower triangular rank equal to one is maintained under one step of the $QR$-algorithm with shift. (Note that for matrices $A$ with weakly lower triangular rank 1 the shift is not essential, because the matrix $A$ plus a diagonal has again weakly lower triangular rank 1.)

THEOREM 62. *Suppose we have a matrix $A$ whose weakly lower triangular rank is* 1 *and a step of $QR$ with shift (with the $QR$-decomposition as in Section 1) is performed on this matrix:*

$$
\begin{cases} A - \kappa I & = & QR \\ \tilde{A} & = & RQ + \kappa I. \end{cases}
$$

*The matrix $\tilde{A}$ will have weakly lower triangular rank equal to one.*

PROOF. The transformation $Q$ consists of two sequences of Givens transformations $Q_1$ and $Q_2$: $Q = Q_1 Q_2$. The first sequence of transformations $Q_1^T$ will make use of the structured rank of the matrix $A$ to reduce $A$ to a Hessenberg matrix. This transformation $Q_1^T$ will also transform the shift matrix $\kappa I$ into a Hessenberg matrix. We have

$$
Q_1^T A - \kappa Q_1^T I = H_1 + H_2 = H
$$

for which $H_1, H_2$ and $H$ are all Hessenberg matrices. The second sequence of Givens transformations $Q_2^T$ will be applied from top to bottom, to transform $H$ into an upper triangular matrix $R$:

$$
Q_2^T Q_1^T (A - \kappa I) = Q_2^T H = R.
$$

The matrix $\tilde{A}$ satisfies now the following equation:

$$
\tilde{A} = RQ_1Q_2 + \kappa I.
$$

Because of the structure of the orthogonal matrices $Q_1$ and $Q_2$ which consist of a sequence of Givens transformations in a specific order we have that $RQ_1$ is a matrix with lower triangular rank equal to 1. Performing the transformation $Q_2$ to the right of $RQ_1$, $RQ_1Q_2$ will transform the lower triangular rank to a weakly lower triangular rank of 1. Therefore our matrix $\tilde{A}$ will have weakly lower triangular rank equal to 1. □

The theorem above already explains that for a Hessenberg-like plus diagonal matrix one again gets a Hessenberg-like plus diagonal matrix after a $QR$-step. In Theorem 64 we will even prove that the diagonal term of the Hessenberg-like plus diagonal matrix is maintained.

We will prove now that a $QR$-step with shift applied on a Hessenberg-like matrix is again such a Hessenberg-like matrix.

THEOREM 63. *Suppose we have a Hessenberg-like matrix $Z$, and we apply one step of the shifted $QR$-algorithm (with the $QR$-decomposition as in Section 1) on this matrix:*

$$\left\{ \begin{array}{rcl} Z - \kappa I & = & QR \\ \tilde{Z} & = & RQ + \kappa I. \end{array} \right.$$

*Then the matrix $\tilde{Z}$ will be a Hessenberg-like matrix.*

PROOF. Without loss of generality, we may assume that the element in the lower left corner is different from zero, otherwise there are zero blocks below the diagonal (a theorem similar to Theorem 24 can be derived to state this fact), and we can split the matrix into different blocks. Suppose we perform now a $QR$-step on the Hessenberg-like matrix $Z$, then we have also the following two equalities:

$$Q_1^T Z = R_1 \tag{38}$$

$$Q_2^T (R_1 - \kappa Q_1^T I) = R_2 \tag{39}$$

with $R_1$ and $R_2$ upper triangular, and $\kappa$ as a shift. Assume $\kappa$ to be different from zero, otherwise we can refer to Theorem 61. Using the equations (38) and (39) we have that

$$Q_1^T(-\kappa I) = H_1$$

where $H_1$ is a Hessenberg matrix, which has all the subdiagonal elements different from zero. All the subdiagonal elements have to be different from zero because all the Givens transformations $G_k$ such that $Q_1 = G_1 \dots G_{n-1}$ are different from a diagonal matrix. This is straightforward because we assumed that the element in the lower left corner is different from zero.

This means that the Hessenberg matrix $H_1$ and therefore also $R_1 + H_1$ is unreduced and has a unique $QR$-decomposition:

$$Q_2^T(R_1 + H_1) = R,$$

where $R$ has two possible structures: all the diagonal elements different from zero or all the diagonal elements different from zero, except for the last one. We have to distinguish between the two cases.

**Case 1:** Suppose all the diagonal elements are different from zero, this means that the matrix $R$ is invertible, this gives us:

$$\begin{array}{rcl} \tilde{Z} & = & RQ + \kappa I \\ & = & RZR^{-1}. \end{array}$$

We can see that the multiplication with the upper triangular matrices will not change the rank of the submatrices below the diagonal. Therefore $\tilde{Z}$ is again a Hessenberg-like matrix.

**Case 2:** Suppose now that our matrix $R$ has the lower right element equal to zero. We partition the matrices $Q$ and $R$ in the following way:

$$\begin{aligned} R &= \left(\begin{array}{c|c} \tilde{R} & w \\ \hline 0 & 0 \end{array}\right) \\ Q &= \left(\begin{array}{c|c} \tilde{Q} & q \end{array}\right), \end{aligned}$$

with $\tilde{R}$ an $(n-1) \times (n-1)$ matrix, $w$ a column vector of length $n-1$, $\tilde{Q}$ a matrix of dimension $n \times (n-1)$ and $q$ a column vector of length $n$. This gives us the following equalities:

$$\begin{aligned} Z - \kappa I &= \left(\begin{array}{c|c} \tilde{Q} & q \end{array}\right) \left(\begin{array}{c|c} \tilde{R} & w \\ \hline 0 & 0 \end{array}\right) \\ &= \left(\begin{array}{c|c} \tilde{Q}\tilde{R} & \tilde{Q}w \end{array}\right). \end{aligned}$$

Denote with $P$ an $n \times (n-1)$ projection matrix of the following form $(I_{n-1} \;\; 0)^T$, then we get (because $\tilde{R}$ is invertible):

$$\begin{aligned} (Z - \kappa I)P &= \tilde{Q}\tilde{R} \\ (Z - \kappa I)P\tilde{R}^{-1} &= \tilde{Q} \\ (Z - \kappa I)\begin{pmatrix} \tilde{R}^{-1} \\ 0 \end{pmatrix} &= \tilde{Q}. \end{aligned}$$

Using these equalities we get:

$$\begin{aligned} \tilde{Z} &= RQ + \kappa I_n \\ &= \left(\begin{array}{c|c} \tilde{R} & w \\ \hline 0 & 0 \end{array}\right) Q + \kappa I_n \\ &= \left(\begin{array}{c|c} \tilde{R} & w \\ \hline 0 & 0 \end{array}\right) \left(\begin{array}{c|c} (Z - \kappa I)\begin{pmatrix} \tilde{R}^{-1} \\ 0 \end{pmatrix} & q \end{array}\right) + \kappa I_n \\ &= \left(\begin{array}{c|c} \hat{Z} - \kappa I_{n-1} & z \\ \hline 0 & 0 \end{array}\right) + \kappa I \\ &= \left(\begin{array}{c|c} \hat{Z} & z \\ \hline 0 & \kappa \end{array}\right) \end{aligned}$$

where $\hat{Z}$ is Hessenberg-like, and therefore also $\tilde{Z}$ is a Hessenberg-like matrix.

$\square$

NOTE. It is clear from the previous proof that the matrix $\tilde{Z}$ immediately reveals the eigenvalue $\kappa$. This $\kappa$ is called the perfect shift.

NOTE. In the previous theorem we spoke about maintaining of structure of Hessenberg-like matrices after a step of $QR$. The proof can however be adapted very easily for generator representable Hessenberg-like matrices. After a step of $QR$ performed on a generator representable Hessenberg-like matrix we get again a generator representable Hessenberg-like matrix, or we will have convergence to one eigenvalue in the lower right corner, while the remaining part in the upper left corner will again be a generator representable Hessenberg-like matrix.

NOTE. Once more we state that it is important that the sequence of Givens transformations are performed in this special order, otherwise it is possible to construct a $QR$-factorization which will generally not give again a Hessenberg-like matrix as the following example shows.

EXAMPLE 27. Suppose we have the following Hessenberg-like matrix:

$$Z = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}.$$

Annihilating the complete last row with a Givens transformation applied on the second and fourth row, and then annihilating the first element of the second row, one gets a $QR$-factorization of the matrix $Z$. Applying the transformations again on the right side of the matrix, will not give as result a Hessenberg-like matrix, because there will be a block of rank 2 in the lower triangular part.

Theorem 62 stated that the structure of a Hessenberg-like matrix plus diagonal is maintained under the $QR$-algorithm with shift. Moreover, the diagonal term of the resulting Hessenberg-like plus diagonal matrix is the same as the original Hessenberg-like plus diagonal matrix.

THEOREM 64. *Suppose $Z$ is a Hessenberg-like matrix and $D$ is a diagonal matrix. Applying a step of $QR$ (with the $QR$-decomposition as in Section 1) with shift on the matrix $Z + D$ results in a Hessenberg-like plus diagonal matrix, which can always be written as a Hessenberg-like plus diagonal matrix with the same diagonal $D$: $\tilde{Z} + D$.*

PROOF. We will assume that we perform a $QR$-step without shift on the Hessenberg-like plus diagonal matrix $Z + D$. This is essentially the same as the proof with shift, because the shift can be included in the diagonal $D$. We have

$$\begin{cases} Q_1^T Z &= R_1 \\ Q_2^T(R_1 + Q_1^T D) &= R, \end{cases}$$

with $R_1$ and $R$ two upper triangular matrices. Let us denote with $H$ the Hessenberg matrix $R_1 + Q_1^T D$. We will distinguish between different cases:

- Suppose $R$ is invertible. This is the same as stating that $Z + D$ is nonsingular. We know that the matrix after one $QR$-step without shift has the following form:

  $$\begin{aligned} R(Z+D)R^{-1} &= RZR^{-1} + RDR^{-1} \\ &= \hat{Z} + \hat{R}, \end{aligned}$$

  where $\hat{Z}$ is clearly a Hessenberg-like matrix, and $\hat{R}$ an upper triangular matrix with as diagonal elements, the diagonal elements of $D$. This can clearly be rewritten as:

  $$\tilde{Z} + D,$$

  which proves one case of the theorem.

- Suppose the Hessenberg matrix $H$ to be unreduced. This means that

$$Q_2^T H = \left(\begin{array}{c|c} \tilde{R} & w \\ \hline 0 & \alpha \end{array}\right),$$

with $\tilde{R}$ invertible. If $\alpha$ is different from zero, the matrix $R$ is invertible, and we are in case 1 of the proof. So we assume $\alpha$ to be zero. Similar to the proof of Theorem 63 we can write:

$$\begin{aligned} RQ &= \left(\begin{array}{c|c} \tilde{R} & w \\ \hline 0 & 0 \end{array}\right) Q \\ &= \left(\begin{array}{c|c} \tilde{R} & w \\ \hline 0 & 0 \end{array}\right) \left(\begin{array}{c|c} (Z+D)\begin{pmatrix} \tilde{R}^{-1} \\ 0 \end{pmatrix} & q \end{array}\right) \\ &= \left(\begin{array}{c|c} \hat{Z}+\hat{D} & z \\ \hline 0 & 0 \end{array}\right), \end{aligned}$$

where $\hat{Z}$ is a Hessenberg-like matrix of dimension $(n-1)\times(n-1)$, if the dimension of $Z$ is $n$ and $\hat{D}$ is a diagonal matrix with as elements the first $n-1$ elements of the matrix $D$. Denoting the last element of $D$ with $d_n$, we can rewrite this as:

$$\begin{aligned} \left(\begin{array}{c|c} \hat{Z}+\hat{D} & z \\ \hline 0 & 0 \end{array}\right) &= \left(\begin{array}{c|c} \hat{Z} & z \\ \hline 0 & -d_n \end{array}\right) + D \\ &= \tilde{Z} + D \end{aligned}$$

for which $\tilde{Z}$ is a Hessenberg-like and $D$ the diagonal, which proves the theorem if $H$ is unreduced.

- No conditions are put on the matrices. We know however that the matrix $H$ is a Hessenberg matrix. This matrix can always be divided in several blocks which are unreduced. For all these different blocks we can apply case 2 and we know that they maintain the Hessenberg-like plus diagonal structure. Recombining the blocks proves the theorem.

□

## 4. The reduction to unreduced Hessenberg-like form

The implicit $QR$-algorithm for Hessenberg matrices is based on the unreducedness of the corresponding Hessenberg matrix. For Hessenberg matrices it is straightforward how to split the matrix into two or more Hessenberg matrices which are unreduced (See Section 7.5.1 in [**91**]).

In this section we will reduce a Hessenberg-like matrix to unreduced form as presented in Definition 59. The first demand, that there are no zero blocks in the lower left corner of the matrix can be satisfied by dividing the matrix into different blocks, in a complete similar way as for the Hessenberg case. This corresponds to deflation of the matrix.

The second demand, the fact that the semiseparable structure does not extend above the diagonal is not solved in such an easy way. It can be seen that matrices having this special structure are singular. In the solution we propose, we will chase in fact the dependent rows completely to the lower right where they will form zero rows which can be removed. This chasing technique is achieved by performing a special $QR$-step without shift on the matrix. We will demonstrate this technique on a $5 \times 5$ matrix. The $\boxtimes$ denote the dependent elements. One can clearly see that the semiseparable structure of the matrix extends above the diagonal:

$$Z = \begin{pmatrix} \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \boxtimes \end{pmatrix}.$$

We start the annihilation with the traditional orthogonal matrix $Q_1 = G_1 G_2 G_3 G_4$. This results in the following matrix $\tilde{R} = G_4^T G_3^T G_2^T G_1^T Z$:

$$\tilde{R} = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \end{pmatrix}.$$

In normal circumstances one would apply the transformation $Q_1$ now on the right of the above matrix to complete one step of $QR$ without shift. Instead of applying now this transformation we continue and annihilate the elements marked with $\otimes$ with the transformations $G_5$ and $G_6$. $G_5$ is performed on rows 3 and 4, while $G_6$ is performed on rows 4 and 5:

$$\begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & \otimes & \times \\ 0 & 0 & 0 & 0 & \times \end{pmatrix} \xrightarrow{G_5^T \tilde{R}} \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \\ 0 & 0 & 0 & 0 & \otimes \end{pmatrix}$$

$$\xrightarrow{G_6^T G_5^T \tilde{R}} \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

We have now finished performing the transformations on the left side of the matrix. Denote $R = G_6^T G_5^T \tilde{R}$. To complete the $QR$-step applied to the matrix $Z$, we have to perform the transformations on the right-side of the matrix:

$$
R = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}
\quad \xrightarrow{RG_1} \quad
\begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \boxtimes & \times \\ 0 & 0 & 0 & \boxtimes & \boxtimes \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}
$$

$$
\xrightarrow{RG_1G_2} \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \boxtimes & \boxtimes & \times \\ 0 & 0 & \boxtimes & \boxtimes & \boxtimes \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}
\quad \xrightarrow{RG_1G_2G_3} \quad
\begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \boxtimes & \times & \times & \times \\ 0 & \boxtimes & \boxtimes & \boxtimes & \times \\ 0 & \boxtimes & \boxtimes & \boxtimes & \boxtimes \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}
$$

$$
\xrightarrow{RQ_1} \begin{pmatrix} \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \boxtimes \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}
\quad \xrightarrow{RQ_1G_5} \quad
\begin{pmatrix} \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \boxtimes \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}
$$

$$
\xrightarrow{RQ_1G_5} \left(\begin{array}{cccc|c} \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \times \\ \hline 0 & 0 & 0 & 0 & 0 \end{array}\right)
$$

It is clear that the last matrix can be deflated such that we get an unreduced matrix and one eigenvalue which has already converged.

The same technique can be applied for matrices with larger blocks crossing the diagonal or more blocks crossing the diagonal.

NOTE. Suppose we have a symmetric semiseparable matrix coming from the reduction algorithm as presented in Part 2. Then we know that this matrix can be written as a block diagonal matrix, for which all the blocks are generator representable semiseparable matrices. Moreover, for all these blocks condition (2) in Definition 59 is satisfied in a natural way. This can be seen by combining the knowledge that the reduction algorithm performs steps of $QR$ without shift, the fact that if condition (2) is not satisfied the matrix has to be singular and the note following Theorem 63. For these matrices the scheme as presented here does not need to be performed. The same is true for Hessenberg-like and upper triangular semiseparable matrices coming from one of these reduction algorithms.

## 5. The implicit $Q$-theorem

We will now prove an implicit $Q$-theorem for Hessenberg-like matrices. Combined with the theoretical results provided in the previous sections, we will be able to derive implicit $QR$-algorithms for semiseparable and Hessenberg-like matrices in the next chapter. The theorem is proved in a similar way as the implicit $Q$-theorem for Hessenberg matrices as formulated in Section 7.4.5 from [**91**]. First two definitions and some propositions are needed.

DEFINITION 65. Two matrices $Z_1$ and $Z_2$ are called essentially the same if there exists a matrix $W = \mathrm{diag}(\pm 1, \pm 1, \ldots, \pm 1)$ such that the following equation holds:

$$Z_1 = W Z_2 W^T.$$

Before proving the implicit $Q$-theorem, we define the so called unreduced number of a Hessenberg-like matrix.

DEFINITION 66. Suppose $Z$ to be a Hessenberg-like matrix, the unreduced number $k$ of $Z$ is the smallest integer such that one of the following two conditions is satisfied:

(1) The submatrix $S(k+1:n; 1:k) = 0$.
(2) The element $S(k+1, k+2)$ with $k < n-1$ is includable in the lower semiseparable structure.

If the matrix is unreduced, $k = n$.

PROPOSITION 67. *Suppose we have a Hessenberg-like matrix $Z$ which is not unreduced, and does not have any zero blocks below the diagonal, i.e., whose bottom left element is nonzero. Then this matrix can be transformed via similarity transformations to a Hessenberg-like matrix, for which the upper left $(n-l) \times (n-l)$ matrix is of unreduced form and the last $l$ rows equal to zero, where $l$ equals the nullity of the matrix. Moreover if $k$ is the unreduced number of the matrix $Z$ then the orthogonal transformation can be chosen in such a way that the upper left $k \times k$ block of this orthogonal transformation equals the identity matrix.*

PROOF. We will explicitely construct the transformation matrix. We illustrate this technique on a matrix of dimension $5 \times 5$ of the following form

$$Z = \begin{pmatrix} \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \boxtimes \end{pmatrix}.$$

We annihilate first the elements of the fifth and fourth row, with Givens transformations $G_1$ and $G_2$. This results in the matrix

$$G_2^T G_1^T Z = \begin{pmatrix} \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \times \\ 0 & 0 & 0 & 0 & \times \\ 0 & 0 & 0 & 0 & \otimes \end{pmatrix}.$$

The Givens transformation $G_3$ is constructed to annihilate the element marked with $\otimes$:

$$G_3^T G_2^T G_1^T Z \quad = \quad \begin{pmatrix} \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \times \\ 0 & 0 & 0 & 0 & \times \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Completing the similarity transformation by applying the transformations $G_1, G_2$ and $G_3$ on the right of $G_3^T G_2^T G_1^T Z$ gives us the following matrix:

$$\tilde{Z} = G_3^T G_2^T G_1^T Z G_1 G_2 G_3 = \left(\begin{array}{ccccc} \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & 0 \end{array}\right).$$

The matrix does not yet satisfy the desired structure, as we want the upper left $4 \times 4$ block to be of Hessenberg-like form. To do so, we remake the matrix semiseparable, by similar techniques as in Part 2. Let us perform the transformation $G_4$ on the right of the matrix $\tilde{Z}$ to annihilate the marked element. Completing this similarity transformation gives us:

$$\tilde{Z} = \left(\begin{array}{ccccc} \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times \\ 0 & 0 & \otimes & \times & \times \\ 0 & 0 & 0 & 0 & 0 \end{array}\right) \xrightarrow{\tilde{Z}G_4} \left(\begin{array}{ccccc} \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & 0 \end{array}\right)$$

$$\xrightarrow{G_4^T \tilde{Z} G_4} \left(\begin{array}{cccc|c} \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \times \\ \hline 0 & 0 & 0 & 0 & 0 \end{array}\right)$$

The resulting matrix has the upper left $4 \times 4$ block of unreduced Hessenberg-like form, while the last row is zero. The transformations involved did not change the upper left $2 \times 2$ block as desired. □

NOTE. The resulting matrices from Proposition 67 are sometimes called zero-tailed matrices (see [**16**]).

NOTE. Proposition 67 can be generalized to Hessenberg-like matrices with zero blocks below the diagonal. These matrices can be transformed to a Hessenberg-like matrix with zero rows on the bottom and all the Hessenberg-like blocks on the diagonal of unreduced form.

PROPOSITION 68. *Suppose $Z$ is an unreduced Hessenberg-like matrix. Then $Z$ can be written as the sum of a rank 1 matrix and a strictly upper triangular matrix, where the superdiagonal elements of this matrix are different from zero. We have that*

$$Z = uv^T + R,$$

*and $u_n$ and $v_1$ are different from zero.*

PROOF. Straightforward, because of the fact that the Hessenberg-like matrix is unreduced. □

PROPOSITION 69. *Suppose we have the following equality:*

$$WZ = XW, \tag{40}$$

*where $W$ is an orthogonal matrix with the first column equal to $e_1$ and $Z$ and $X$ are two Hessenberg-like matrices of the following form:*

$$Z = \begin{pmatrix} Z_1 & \times & \times \\ 0 & Z_2 & \times \\ 0 & 0 & 0 \end{pmatrix} \quad X = \begin{pmatrix} X_1 & \times & \times \\ 0 & X_2 & \times \\ 0 & 0 & 0 \end{pmatrix}$$

*both having $l$ zero rows at the bottom and the matrices $Z_1$, $Z_2$, $X_1$ and $X_2$ of unreduced form. If we denote the dimension of the upper left block of $Z$ with $n_{Z_1}$ and the dimension of the upper left block of $X$ with $n_{X_1}$. Then we have that $n_{X_1} = n_{Z_1}$ and $W$ has the lower left $(n - n_{X_1}) \times n_{X_1}$ block equal to zero.*

PROOF. Assume $n_{Z_1} \geq n_{X_1}$. When considering the first $n_{Z_1}$ columns of equation (40) we have

$$W(1:n, 1:n_{Z_1})\, Z_1 = V, \tag{41}$$

where $V$ is of dimension $n \times n_{Z_1}$, with the last $l$ rows of $V$ equal to zero. Because $Z_1$ is invertible we know that $W(1:n, 1:n_{Z_1}) = VZ_1^{-1}$ has the last $l$ rows equal to zero. We will prove by induction that all the columns $w_k$ with $1 \leq k \leq n_{X_1}$ have the components with index higher than $n_{X_1}$ equal to zero.

Step 1: $k = 1$. Because $We_1 = e_1$, we know already that this is true for $k = 1$. Let us write the matrices $Z_1$ and $X_1$ as (Using Proposition 68):

$$\begin{aligned} Z_1 &= u^{(1)}{v^{(1)}}^T + R^{(1)}, \\ X_1 &= u^{(2)}{v^{(2)}}^T + R^{(2)}, \end{aligned}$$

with $v_1^{(1)} = v_1^{(2)} = 1$. Multiplying equation (40) to the right with $e_1$ gives us

$$W \begin{pmatrix} u^{(1)} \\ 0 \\ \vdots \\ 0 \end{pmatrix} = \begin{pmatrix} u^{(2)} \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Note that $u^{(1)}$ is of length $n_{Z_1}$ and $u^{(2)}$ is of length $n_{X_1}$.

Step $k$: Suppose $2 \leq k \leq n_{X_1}$. We prove that $w_k$ has the components $n_{X_1}+1, \ldots, n$ equal to zero. We know by induction that this is true for the columns $w_i$ with $i < k$ . Multiply both sides of equation (40) with $e_k$, this gives us:

$$W \left( \begin{pmatrix} u^{(1)} \\ 0 \\ \vdots \\ 0 \end{pmatrix} v_k^{(1)} + \begin{pmatrix} r_{1,k}^{(1)} \\ \vdots \\ r_{k-1,k}^{(1)} \\ 0 \\ \vdots \\ 0 \end{pmatrix} \right) = Xw_k.$$

This can be rewritten as:

$$u^{(2)}v_k^{(1)} + W \begin{pmatrix} r_{1,k}^{(1)} \\ \vdots \\ r_{k-1,k}^{(1)} \\ 0 \\ \vdots \\ 0 \end{pmatrix} = Xw_k.$$

Because of the induction procedure, we know that the left-hand side of the former equation has the components below element $n_{X_1}$ equal to zero. The vector $w_k$ only has the first $n-l$ components different from zero. Because the matrices $X_1$ and $X_2$ are nonsingular we know that $w_k$ has only the first $n_{X_1}$ components different from zero, because $Xw_k$ can only have the first $n_{X_1}$ components different from zero. This proves the induction step.

This means that the matrix $W$ has in the lower left position an $(n-n_{X_1}) \times n_{X_1}$ block of zeros. A combination of equation (40) and the zero structure of $W$ leads to the fact that $WZ$ needs to have a zero block in the lower left position of dimension $(n-n_{X_1}) \times n_{X_1}$. Therefore the matrix $Z_1$ has to be of dimension $n_{X_1}$ which proves the proposition. □

NOTE. The proposition above can be formulated in a similar way for more zero blocks below the diagonal.

Using this property, we can prove the following implicit $Q$-theorem, for Hessenberg-like matrices.

THEOREM 70 (implicit $Q$-theorem for Hessenberg-like matrices ). *Suppose the following equations hold,*

$$Q_1^T A Q_1 = Z \tag{42}$$

$$Q_2^T A Q_2 = X \tag{43}$$

*with $Q_1e_1 = Q_2e_1$, where $Z$ and $X$ are two Hessenberg-like matrices, with unreduced numbers $k_1$ and $k_2$ respectively and $Q_1$ and $Q_2$ are orthogonal matrices. Let us denote $k = \min(k_1, k_2)$. Then we have that the first $k$ columns of $Q_1$ and $Q_2$ are the same, up to the sign, and the upper left $k \times k$ submatrices of $Z$ and of $X$ are essentially the same. More precisely, there exists a matrix $V = \mathrm{diag}(1, \pm 1, \pm 1, \ldots, \pm 1)$, of size $k \times k$, such that we have the following two equations:*

$$\begin{aligned} Q_1(1:k, 1:k) &= Q_2(1:k, 1:k)\,V, \\ Z(1:k, 1:k)\,V &= V\,X(1:k, 1:k). \end{aligned}$$

PROOF. Using Proposition 67 and the note following the proposition we can assume that the matrices $Z$ and $X$ in (42) and (43) are of the following form (for simplicity, we assume the number of blocks to be equal to two):

$$Z = \begin{pmatrix} Z_1 & \times & \times \\ 0 & Z_2 & \times \\ 0 & 0 & 0 \end{pmatrix} \quad X = \begin{pmatrix} X_1 & \times & \times \\ 0 & X_2 & \times \\ 0 & 0 & 0 \end{pmatrix},$$

with $Z_1$, $Z_2$, $X_1$ and $X_2$ unreduced Hessenberg-like matrices. This does not affect our statements from the theorem, as the performed transformations from Proposition 67 do not affect the upper left $k \times k$ block of the matrices $Z$ and/or $X$.

Denoting $W = Q_1^T Q_2$ and using the equations (42) and (43) the following equality holds:

$$ZW = WX. \tag{44}$$

Moreover, we know by Proposition 69 that the matrix $W$ has the lower left block of dimension $(n - n_1) \times n_1$ equal to zero, where $n_1$ is the dimension of the block $Z_1$.

If we can prove now that for the first $n_1$ columns of $W$ the following equality holds: $w_k = \pm e_k$, then we have that (as $n_1 \geq k$ and by Proposition 67) the theorem holds.

According to Proposition 68 we can write the upper rows of the matrix $Z$ in the following form:

$$Z(1:n_1; 1:n) = u^{(1)}\left({v^{(1)}}^T, {\tilde{v}^{(1)}}^T\right) + R^{(1)},$$

where $u^{(1)}, v^{(1)}$ have length $n_1$. The vector $\tilde{v}^{(1)}$ is of length $n - n_1$ and is chosen in such a way that the matrix $R^{(1)} \in \mathbb{R}^{n_1 \times n}$, has the last row equal to zero. Moreover the matrix $R^{(1)}$ has the left $n_1 \times n_1$ block strictly upper triangular. Also the left part of $X$ can be written as

$$X(1:n; 1:n_1) = \begin{pmatrix} u^{(2)} \\ \tilde{u}^{(2)} \end{pmatrix} {v^{(2)}}^T + R^{(2)}$$

with $u^{(2)}$ and $v^{(2)}$ of length $n_1$, $\tilde{u}^{(2)}$ is of length $n - n_1$ and $R^{(2)}$ a strictly upper triangular matrix of dimension $n \times n_1$. Both of the strictly upper triangular parts $R^{(1)}$ and $R^{(2)}$ have nonzero elements on the supdiagonals. The couples $u^{(1)}, v^{(1)}$ and $u^{(2)}, v^{(2)}$ are the generators of the semiseparable matrices $Z_1$ and $X_1$, respectively. They are normalized in such a way that $v_1^{(1)} = v_1^{(2)} = 1$. Denoting with $P$ the projection operator $P = (I_{n_1},\ 0)$ we can calculate the upper left $n_1 \times n_1$ block of the matrices in equation (44):

$$\begin{aligned} PZWP^T &= PWXP^T \\ \left(u^{(1)}\left({v^{(1)}}^T, {\tilde{v}^{(1)}}^T\right) + R^{(1)}\right) WP^T &= PW\left(\begin{pmatrix} u^{(2)} \\ \tilde{u}^{(2)} \end{pmatrix} {v^{(2)}}^T + R^{(2)}\right). \end{aligned} \tag{45}$$

Denoting the columns of $W$ as $(w_1, w_2, \ldots, w_n)$, the fact that $Q_1 e_1 = Q_2 e_1$ leads to the fact that $w_1 = e_1$. Hence, also the first row of $W$ equals $e_1^T$. Multiplying (45) to the right by $e_1$, gives

$$u^{(1)} = PW \begin{pmatrix} u^{(2)} \\ \tilde{u}^{(2)} \end{pmatrix}. \tag{46}$$

Because of the structure of $P$ and $W$, multiplying (46) to the left by $e_1^T$ gives us that $u_1^{(1)} = u_1^{(2)}$. Using equation (46), we will prove now by induction that for $i \leq n_1$ : $w_i = \pm e_i$ and

$$\left({v^{(1)}}^T, {\tilde{v}^{(1)}}^T\right) WP^T = \left({v^{(1)}}^T, {\tilde{v}^{(1)}}^T\right) (w_1,\ w_2, \ldots, w_{n_1}) = {v^{(2)}}^T, \tag{47}$$

which will prove the theorem.

Step 1 $l = 1$. This is a trivial step, because $v_1^{(1)} = v_1^{(2)} = 1$ and $w_1 = e_1$.

Step $l$ $1 < l \leq n_1$. By induction we have that $w_i = \pm e_i \quad \forall 1 \leq i \leq l-1$ and (47) holds for the first $(l-1)$ columns. This means that:

$$\left({v^{(1)}}^T, {\tilde{v}^{(1)}}^T\right)(w_1, w_2, \ldots, w_{l-1}) = \left(v_1^{(2)}, v_2^{(2)}, \ldots, v_{l-1}^{(2)}\right).$$

Taking (46) into account, (45) becomes

$$u^{(1)}\left(\left({v^{(1)}}^T, {\tilde{v}^{(1)}}^T\right)WP^T - {v^{(2)}}^T\right) = \left(PWR^{(2)} - R^{(1)}WP^T\right). \tag{48}$$

Multiplying (48) to the right by $e_l$, we have

$$u^{(1)}\left(\left({v^{(1)}}^T, {\tilde{v}^{(1)}}^T\right)WP^T - {v^{(2)}}^T\right)e_l = \left(PWR^{(2)} - R^{(1)}WP^T\right)e_l. \tag{49}$$

Because of the special structure of $P, W, R^{(1)}$ and $R^{(2)}$ the element in the $n_1$th position in the vector on the right-hand side of (49) is equal to zero.

We know that $u_{n_1}^{(1)}$ is different from zero, because of the unreducedness assumption. Therefore the following equation holds:

$$u_{n_1}^{(1)}\left(\left({v^{(1)}}^T, {\tilde{v}^{(1)}}^T\right)WP^T - {v^{(2)}}^T\right)e_l = 0.$$

This means that

$$\left({v^{(1)}}^T, {\tilde{v}^{(1)}}^T\right)w_l - v_l^{(2)} = 0.$$

Therefore equation (47) is already satisfied up to element $l$:

$$\left({v^{(1)}}^T, {\tilde{v}^{(1)}}^T\right)(w_1, w_2, \ldots, w_l) = \left(v_1^{(2)}, v_2^{(2)}, \ldots, v_l^{(2)}\right). \tag{50}$$

Using equation (50) together with equation (49) leads to the fact that the complete right-hand side of equation (49) has to be zero. This gives the following equation:

$$R^{(1)}WP^Te_l = PWR^{(2)}e_l$$

leading to

$$R^{(1)}We_l = \sum_{j=1}^{l-1} r_{j,l}^{(2)} w_j \tag{51}$$

with the $r_{j,l}^{(2)}$ as the elements of column $l$ of matrix $R^{(2)}$. Hence, the right-hand side can only have the first $l-1$ components different from zero. Because the superdiagonal elements of the left square block of $R^{(1)}$ are nonzero and because the first $n_1$ columns of $W$ have the last $(n-n_1)$ elements equal to zero, only the first $l$ elements of $w_l$ can be different from zero. This together with the fact that $W$ is orthogonal means that $w_l = \pm e_l$, which proves the induction step.

□

NOTE. In the definition of the unreduced number we assumed that also the element $Z(1,2)$ was not includable in the semiseparable structure. The reader can verify, that one can admit this element to be includable in the semiseparable structure, and the theorem and the proof still remain true for this newly defined unreduced number.

NOTE. This theorem can also be applied to the reduction algorithms which transform matrices to Hessenberg-like, symmetric semiseparable and upper triangular semiseparable form, thereby stating the uniqueness of these reduction algorithms in case the outcome is in unreduced form.

Finally we give some examples, connected with this theorem. (See also [**16**].) The first condition in Definition 66 is quite logical and we will not give any examples connected with this condition.

EXAMPLE 28. Suppose we have the matrices:

$$A = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} \quad Q_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \\ 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix}$$

and $Q_2{=}I$. Then we have that $X = A$ and

$$Z = \begin{pmatrix} 1 & \sqrt{2} & 0 \\ \sqrt{2} & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

One can see that the unreduced number equals 1 because the element $X(2,2)$ is includable in the lower semiseparable structure of the matrix $X$. Thus we know by the theorem that equality is only garantueed for the upper left element of the matrices $X$ and $Z$.

EXAMPLE 29. Suppose we have the matrices:

$$A = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} \quad Q_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

and $Q_2{=}I$. Then we have that $X = A$ and

$$Z = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

One can see that the unreduced number equals 0 because the element $X(1,2)$ is includable in the lower semiseparable structure of the matrix $X$. But using the definition of the unreduced number connected to the note following Theorem 70, we know that the unreduced number equals 1 and the equality holds for the upper left element of $Z$ and $X$.

## Conclusions

In this chapter we investigated theoretical results in order to design implicit $QR$-algorithms. We showed that a special kind of $QR$-factorization is needed in order to have the structure maintained after a $QR$-step. Using this $QR$-factorization, which consists of $2n-1$ Givens transformations, we proved that the structure of a Hessenberg-like matrix is maintained. Moreover we proved that for a Hessenberg-like plus diagonal matrix the structure and the diagonal are maintained after a step of this $QR$-algorithm. As in the tridiagonal case we defined an unreduced Hessenberg-like matrix and a method to transform a Hessenberg-like matrix to the unreduced form. Based on the unreduced number, an implicit $Q$-theorem for Hessenberg-like matrices was derived.

CHAPTER 8

# Implicit $QR$-algorithms for semiseparable matrices

In the previous chapter interesting theorems connected to $QR$-algorithms for Hessenberg-like matrices were proved. Starting with a Hessenberg-like matrix we know how to transform it to unreduced form. The structure of a Hessenberg-like matrix after an explicit step of the $QR$-algorithm is known and an implicit $Q$-theorem for Hessenberg-like matrices was proved. To construct an implicit algorithm only the implicit step of $QR$ is missing. In this chapter we will provide this step for Hessenberg-like matrices and for symmetric semiseparable matrices. Moreover, we will translate this implicit $QR$-step such that we can use it to calculate the singular values of an upper triangular semiseparable matrix. The results provided in this chapter can be combined in a straightforward way with the reduction algorithms presented in Part 2. Using this combination we can compute the eigenvalues of symmetric and unsymmetric matrices via intermediate semiseparable and Hessenberg-like matrices and we can also compute the singular values of arbitrary matrices via intermediate upper triangular semiseparable matrices.

In the first section an implicit $QR$-algorithm for semiseparable matrices will be designed. Brief discussions about the unreducedness and the type of shift are included. One should make use of the symmetry of the matrix when transforming it to unreduced form. The shift we will consider is the Wilkinson shift. The actual implicit $QR$-algorithm consists of two orthogonal transformations. In a first orthogonal transformation in fact a step of $QR$ without shift will be performed on the semiseparable matrix. The resulting matrix is again semiseparable. After this step a similarity Givens transformation will be applied on the first 2 columns and first 2 rows of the new semiseparable matrix. This will disturb the semiseparable structure. From now on we switch to the implicit approach to restore the semiseparable structure. We will prove that the combination of these two steps corresponds to performing one step of the explicit $QR$-algorithm.

In the second section we focus on the development of an implicit $QR$-algorithm for Hessenberg-like matrices. One important remark has to be made: we only consider Hessenberg-like matrices with real eigenvalues, otherwise a double shift strategy should be used, which is not included in our survey. The approach is similar to the symmetric case. First a step of $QR$ without shift is performed. Afterwards a similarity Givens transformation is applied. The resulting disturbed Hessenberg-like matrix will then be reduced back to Hessenberg-like form. Using the implicit $Q$-theorem one can easily prove that it corresponds to performing an explicit step of $QR$.

The last section focusses on the computation of singular values of upper triangular semiseparable matrices $S_u$. The implicit approach will be deduced by looking at the implicit $QR$-algorithm applied to the symmetric matrix $S = S_u^T S_u$. Before starting the design of the new method, a brief explanation of the $QR$-method for bidiagonal matrices is given. We will explain what the structure of an unreduced upper triangular semiseparable matrix is, and how we can transform it to this form. It will be shown that one step of the $QR$-method applied to the upper triangular semiseparable matrix $S_u$ corresponds to four main steps: transforming the upper triangular semiseparable matrix to lower triangular form; creating a disturbation in the lower triangular semiseparable matrix; transforming the matrix back to lower triangular semiseparable form; finally transforming the resulting semiseparable matrix back to upper triangular semiseparable form.

## 1. An implicit $QR$-algorithm for symmetric semiseparable matrices

In this section we will derive an implicit $QR$-algorithm for symmetric semiseparable matrices. We will make extensive use of the theorems and tools provided in the previous chapter for Hessenberg-like matrices. Because the class of matrices we are working with is symmetric, we will sometimes adapt a little bit the tools slightly, taking into account the symmetry.

Suppose we have a symmetric semiseparable matrix $S$, and a shift $\kappa$ then we want to calculate the matrix $\tilde{S}$ for which:

$$\left\{ \begin{array}{rcl} S - \kappa I & = & QR \\ \tilde{S} & = & RQ + \kappa I. \end{array} \right. \tag{52}$$

(The $QR$-factorization is defined as in Chapter 7 Section 1.) Performing the transformations of (52) explicitely leads to an explicit $QR$-algorithm for semiseparable matrices. This means, first calculating the $QR$-factorization and then performing the product $RQ$ Explicit $QR$-algorithms can be found in the literature, see, e.g. [**19**]. We can however also write the matrix $\tilde{S}$ as

$$\tilde{S} = Q^T S Q.$$

In this section we will calculate the matrix $\tilde{S}$ based on these formulas without explicitely forming the orthogonal matrix $Q$. Because $Q$ consists of a sequence of $2n-2$ Givens transformations, we will apply these Givens transformations one after the other on both sides of the matrix. In fact we will calculate a matrix $\hat{S}$ satisfying

$$\hat{S} = \hat{Q}^T S \hat{Q}$$

such that $Qe_1 = \hat{Q}e_1$. The matrix $\hat{Q}$ will be constructed implicitly, without considering the complete $QR$-factorization of the matrix $S$. Using the implicit $Q$-theorem (see Theorem 70) we know then that the matrices $\hat{S}$ and $\tilde{S}$ are essentially the same. Hence $\tilde{S}$ is the result of applying a $QR$-step on the matrix $S$, with shift $\kappa$. We will now consider the different steps to obtain an implicit $QR$-algorithm for symmetric semiseparable matrices.

**1.1. Unreduced symmetric semiseparable matrix.** In Section 2 of the previous chapter we have developed a method to transform a Hessenberg-like matrix to unreduced form. An unreduced symmetric semiseparable matrix satisfies exactly the same properties as defined in the previous chapter. However, because the matrix we are working with is symmetric, it is easier to reduce such a matrix to unreduced form. Suppose for example we have the following symmetric semiseparable matrix:

$$\left(\begin{array}{ccccc} \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \boxtimes \end{array}\right).$$

For which the $\boxtimes$ denote the elements satisfying the semiseparable structure corresponding to the lower triangular part. It can be seen that also the element $(2,3)$ is part of this semiseparable structure, therefore, this matrix is not in unreduced form. We have to distinguish between two cases now. If the elements in positions $(2,3)$ and $(3,3)$ are zero, then we have because of the symmetry and the semiseparable structure that the complete column 3 and row 3 are zero. By permuting the matrix, this zero row and column can be placed in the final column and row and they can be removed via deflation. Suppose however that the elements are different from zero, then we know, because of the semiseparable structure of the upper triangular part that the complete row 2 and row 3 are dependent of each other. To remove these dependent rows, one can easily perform a Givens transformation on row 2 and 3 which will annihilate the complete row 3. Performing the Givens transformation on the right, in order to complete the similarity transformation, will annihilate all the elements in column 3. One can permute this zero column and row to the last position and remove them via deflation. This technique can be continued until the symmetric semiseparable matrix we are working with is unreduced.

**1.2. The shift $\kappa$.** The choice of the shift is very important to achieve a fast convergence about one of the eigenvalues. The shifts we consider here are discussed in [**91**]. More references towards analysis of different shifts can also be found there. Suppose we have the following semiseparable matrix represented with the Givens-vector representation:

$$\left(\begin{array}{cccc} c_1 d_1 & & & \\ c_2 s_1 d_1 & c_2 d_2 & & \\ c_3 s_2 s_1 d_1 & c_3 s_2 d_2 \quad c_3 d_3 & & \\ \vdots & & \ddots & \\ c_{n-1} s_{n-2} \cdots s_1 d_1 & \ldots & c_{n-1} d_{n-1} & \\ s_{n-1} s_{n-2} \cdots s_1 d_1 & \ldots & s_{n-1} d_{n-1} & d_n \end{array}\right).$$

One can choose $d_n$ as a shift, the so-called Rayleigh shift, or one can consider as a shift the eigenvalue of

$$\left(\begin{array}{cc} c_{n-1} d_{n-1} & s_{n-1} d_{n-1} \\ s_{n-1} d_{n-1} & d_n \end{array}\right),$$

that is closest to $d_n$, the Wilkinson shift [**190**]. Using this shift in the tridiagonal case will give cubic convergence. The numerical results provided in Chapter 9 will experimentally prove the same rate of convergence for the symmetric semiseparable case.

**1.3. An implicit $QR$-step on a symmetric semiseparable matrix.** The first $n-1$ Givens transformations are in fact completely determined by the semiseparable matrix. To perform these transformations, the shift $\kappa$ is not yet needed. When applying these Givens transformations to the left of the semiseparable matrix $S^{(0)} = S$, this matrix becomes upper triangular:

$$G_{n-1}^T \dots G_1^T S = R.$$

Directly applying the Givens transformations $G_1 \dots G_n$ on the right of the matrix $R$, will construct a matrix $RG_1 \dots G_{n-1}$ whose lower triangular part is semiseparable. Because of symmetry reasons the resulting matrix $RG_1 \dots G_{n-1}$ is a symmetric semiseparable matrix. It can be seen that the application of the different Givens transformations can be done at the same time, i.e. instead of first applying all the transformations on the left, we apply them to left and to right at the same time. More details about this step can be found in the next chapter. There the implementation of the algorithm will be described, and there one can clearly see the behavior of the algorithm as explained above (Chapter 9, Section 2):

$$S^{(1)} = G_1^T S^{(0)} G_1$$

followed by

$$S^{(2)} = G_2^T S^{(1)} G_2$$

and so on.

As stated before, this step corresponds to applying a $QR$-step without shift to the semiseparable matrix.

The application of the second sequence of Givens transformations is the hardest step of the two and requires some theoretical results. To initialize this step the knowledge of the Givens transformation $G_n^T$ is crucial. $G_n^T$ is the Givens transformation which will start to reduce the Hessenberg matrix $G_{n-1}^T \dots G_1^T (S - \kappa I)$ to upper triangular form. The algorithm however did not calculate $G_{n-1}^T \dots G_1^T (S - \kappa I)$ but a semiseparable matrix $S^{(n-1)} = G_{n-1}^T \dots G_1^T S G_1 \dots G_{n-1}$. However, because we use the Givens-vector representation as mentioned in the first section, we know that the upper left element of the matrix $G_{n-1}^T \dots G_1^T S$ is the first element in the vector $d$ from the Givens vector representation of the matrix $S$. It can be seen that the elements in the upper left positions $(1,1)$ and $(2,1)$ of the matrix $G_{n-1}^T \dots G_1^T \kappa I$ equal

$$G_{n-1}^T \begin{pmatrix} \kappa \\ 0 \end{pmatrix}.$$

This means that the Givens transformation $G_n^T$ already can be applied to the matrix $S^{(n-1)}$, i.e., $S^{(n)} = G_n^T S^{(n-1)} G_n$. From this point we will work directly on the matrix $S^{(n)}$ and therefore we switch to the implicit approach.

The matrix $S^{(n-1)}$ was a semiseparable matrix and the output of one step of the implicit $QR$-algorithm also has to be a semiseparable matrix. However after having applied the similarity transformation $G_n^T S^{(n-1)} G_n$ the semiseparable structure is disturbed. The following sequence of Givens transformations which will be applied to the matrix $S^{(n)}$ will restore the semiseparable structure. Even more: the resulting matrix will essentially be the same as the matrix coming from one step of the $QR$-algorithm with shift $\kappa$. We will show that it is possible to rebuild a semiseparable matrix out of $S^{(n)}$ without changing the first row and column. To do so we use some kind of chasing technique, more information about general chasing techniques can be found in, e.g. [**186, 188**]. To prove the two main theorems, two propositions are needed.

PROPOSITION 71. *Suppose the following symmetric* $3 \times 3$ *matrix is given,*

$$A = \begin{pmatrix} x & a & d \\ a & b & e \\ d & e & f \end{pmatrix}, \tag{53}$$

*which is not yet semiseparable. Then there exists a Givens transformation*

$$G = \begin{pmatrix} 1 & 0 \\ 0 & \hat{G} \end{pmatrix} \quad \text{with} \quad \hat{G} = \frac{1}{\sqrt{1+t^2}} \begin{pmatrix} t & -1 \\ 1 & t \end{pmatrix}$$

*such that the following matrix*

$$G^T A G$$

*is a symmetric semiseparable matrix.*

PROOF. We can assume that $ae - bd$ is different from zero, otherwise, the matrix would already be semiseparable. The proof is constructive, i.e. the matrix $G$ will be constructed such that the matrix $G^T A G$ indeed is a semiseparable matrix. Calculating explicitly the product $G^T A G$ gives the following matrix (with $s = 1+t^2$):

$$\frac{1}{s} \begin{pmatrix} sx & \sqrt{s}(at+d) & \sqrt{s}(-a+dt) \\ \sqrt{s}(ta+d) & (tb+e)t + (te+f) & (-1)(tb+e) + (te+f)t \\ \sqrt{s}(-a+td) & (-b+te)t + (-e+tf) & (-1)(-b+te) + (-e+tf)t \end{pmatrix}.$$

When this matrix has to be semiseparable the determinant of the lower left 2 by 2 block has to be zero, i.e.

$$\big(ta+d\big)\big((-b+te)t + (-e+tf)\big) - \big((tb+e)t + (te+f)\big)\big(-a+td\big) = 0.$$

Solving this equation for $t$ gives the following result:

$$t = \frac{de - af}{ae - db},$$

which is properly defined because $ae - db \neq 0$. □

PROPOSITION 72. *Suppose the following symmetric* $4 \times 4$ *matrix is given,*

$$A = \begin{pmatrix} x & a & 0 & d \\ a & b & 0 & e \\ 0 & 0 & 0 & f \\ d & e & f & y \end{pmatrix}, \tag{54}$$

*which is not yet semiseparable. Then there exists a Givens transformation*

$$G = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \hat{G} & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{with} \quad \hat{G} = \frac{1}{\sqrt{1+t^2}} \begin{pmatrix} t & -1 \\ 1 & t \end{pmatrix}$$

*such that the upper* 3 *by* 3 *block of the following matrix*

$$G^T A G$$

*is a symmetric semiseparable matrix. And the lower left two by two block is of rank one.*

PROOF. The proof is straightforward. Calculating the product $G^T AG$, shows that the block

$$\begin{pmatrix} -a & -tb \\ d & te+f \end{pmatrix}$$

has to be of rank 1. This corresponds with $t = -af/(ae - bd)$. Where $(ae - bd)$ is different from zero because the matrix is not yet semiseparable. □

One might wonder how both these theorems can be used for larger matrices. This is shown in the following theorem:

THEOREM 73. *Suppose an unreduced symmetric semiseparable matrix $S$ is given. On this matrix a similarity Givens transformation $\tilde{G}$ is performed, involving the first two columns and rows:*

$$\tilde{S} = \tilde{G}^T S \tilde{G}.$$

*Then there exists a Givens transformation $G$:*

$$G = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \hat{G} & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \textit{with} \quad \hat{G} = \frac{1}{\sqrt{1+t^2}} \begin{pmatrix} t & -1 \\ 1 & t \end{pmatrix}$$

*such that the upperleft 3 by 3 block of the matrix $G^T \tilde{S} G$ is semiseparable, and the first two columns are dependent, except for the first element.*

PROOF. The proof is divided in different cases, thereby using both of the above propositions. Assume the Givens transformation $\tilde{G}$ to be different from the identity matrix, otherwise we take $G$ equal to the identity matrix.

Case 1 Suppose the upper left $3 \times 3$ submatrix $\tilde{S}$ is the same as in equation (53). In this case Proposition 71 is applied.

Case 2 Suppose the upper left $4 \times 4$ submatrix $\tilde{S}$ is the same as in equation (54). In this case Proposition 72 is applied.

The only thing which remains to prove is that these are the only two possible cases which will occur. Suppose the upper left $3 \times 3$ block of $\tilde{S}$ is of the following form:

$$\begin{pmatrix} x & a & d \\ a & b & e \\ d & e & f \end{pmatrix}. \tag{55}$$

It is obvious that the $a$ or $b$ has to be different from zero, otherwise the matrix $S$ would not have been in unreduced form, because the upper left $2 \times 2$ block would have been of rank 1. If $d$ and/or $e$ is different from zero a, not so easy, calculation reveals that $ae \neq bd$. This is the case because, if we assume that $ae = bd$, the original matrix $S = \tilde{G}\tilde{S}\tilde{G}^T$ has the first two columns of rank 1, which is in contradiction with the unreducedness. This means that we are in case 1.

Suppose the matrix $\tilde{S}$ has the upper left $4 \times 4$ block of the following form: (This corresponds to $d = e = 0$ in equation (55))

$$\begin{pmatrix} x & a & 0 & d \\ a & b & 0 & e \\ 0 & 0 & 0 & f \\ d & e & f & y \end{pmatrix}.$$

We will first prove that or $d$ or $e$ has to be different from zero. Assume now, $d = e = 0$, and $f \neq 0$. This would mean that the complete columns underneath $e$ and $d$ would be zero, because the original matrix $S$ is semiseparable. This is in contradiction with the unreducedness of the original matrix $S$, which can be written in this case as a block diagonal matrix. Suppose now $d = e = f = 0$, this would mean that we have the following upper 4 by 4 matrix:

$$\begin{pmatrix} x & a & 0 & 0 \\ a & b & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & y \end{pmatrix}.$$

For this matrix the element in position $(3, 4)$ can be included in the semiseparable structure of the original matrix $S$, this is in contradiction with the unreducedness assumption.

We know now, that $d$ and/or $e$ is different from zero and we can again assume that $ae \neq bd$. This means that we are in case 2 of the theorem.

□

Now we are ready to reduce a semiseparable matrix which is disturbed in the upper left two by two block, back to semiseparable form.

THEOREM 74. *Suppose we have an $n$ by $n$ unreduced symmetric semiseparable matrix $S$, which will be disturbed in the first two rows and columns by means of a similarity Givens transformation $G$. Then there exists an orthogonal transformation $U$ with $Ue_1 = e_1$ such that $U^T G^T S G U$ is again a symmetric semiseparable matrix.*

PROOF. In this theorem a 5 by 5 matrix is considered, and we will use the special Givens transformation from Theorem 73. Some more notation is needed to simplify the construction of $U$: Denote with $G^{(i)}$ the orthogonal transformation which performs a Givens transformation on the columns $i$ and $i + 1$ of the matrix $S$. To prove that the algorithm gives the desired result, several figures are included. Starting with the matrix $S$ each figure shows all the dependencies in the matrix. In the following figures, the blocks grouped by the full lines represent semiseparable parts in the matrix, and the elements grouped by the dashed lines represent rank 1 parts. These rank 1 parts are very important for the progress of the algorithm. The arrows point out the rows and columns on which we performe the Givens transformations to come to the desired result. The first figure (Figure 8.1), shows what happens with the matrix $S$ after the disturbing Givens transformation is applied.

The following step consists of calculating the Givens transformation from Theorem 73. For case 1 of the theorem, Proposition 71 is used and more explanation is needed how we get to the structured matrix on the right in Figure 8.3. For case 2

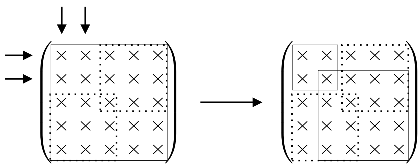

Figure 8.1: Applying the Givens transformation which disturbs the semiseparable structure.

of Theorem 73, we can skip the following comments, and we immediately arrive at the matrix on the right of Figure 8.3.

When applying the Givens transformation of the first kind, we take a closer look to see how the dependencies will change. First we apply the transformation ${G^{(2)}}^T$ only on the left of the matrix, to see how the dependencies will change (see Figure 8.2). There are now two rank 1 parts, the small 2 by 2 matrix and the larger 3 by 3 matrix. The small block has to be of rank 1 because the next Givens transformation $G^{(2)}$ applied to the right will not change the rank of this block, and after this Givens transformation that block is part of the semiseparable matrix, and therefore of rank 1. The 3 by 3 block remains of rank 1 after the Givens transformation is performed. This means that after applying the Givens transformation on the left we have a 2 by 4 matrix of rank 1 (see Figure 8.3), because the first three elements of column 3 of the matrices in Figure 8.3 are not all equal to zero. If they were all zero, we would have been in case 2 of Theorem 73.

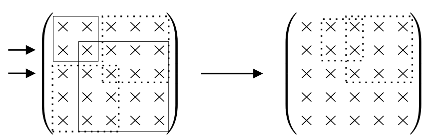

FIGURE 8.2: Applying the transformation ${G^{(2)}}^T$ to the left.

Applying the transformation $G^{(2)}$ to the right of the matrix, gives the matrix $S^{(2)} = {G^{(2)}}^T S(G^{(2)})$ which, because of symmetry reasons has the following structure:

The figure shows that the upper semiseparable part has increased, while the lower semiseparable part is reduced in dimension. Very important are the remaining rank 1 parts. The remaining dependency in these blocks make sure that the next Givens transformation indeed create a semiseparable matrix of dimension 4. The next Givens transformation $G^{(3)}$ is calculated by using Theorem 73 applied to the matrix $S^{(2)}$ without the first row and column. Applying the transformation $G^{(3)}$, this means calculating $S^{(3)} = {G^{(3)}}^T S(G^{(3)})$, will create a semiseparable block in the middle of the matrix. However because of the rank 1 parts, the complete upper left 4 by 4 block will become dependent. This is shown in the following Figure 8.4:

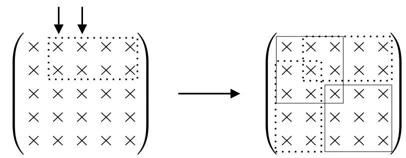

FIGURE 8.3: Applying the transformation $G^{(2)}$ to the right

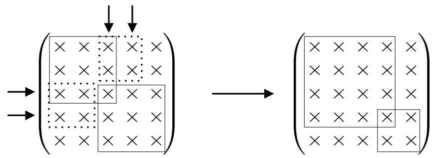

FIGURE 8.4: Applying the similarity transformation $G^{(3)}$

Before performing the final Givens transformation, one can also search the rank 1 blocks such that the last Givens transformation will transform the matrix into a complete semiseparable one.

Even though we proved the theorem here for a semiseparable matrix of dimension 5, one can easily see that the statements remain true, also for higher dimensions of the matrices. $\square$

This final theorem produces an algorithm to transform the semiseparable matrix with a disturbance in the upper left part back to an orthogonal similar semiseparable matrix. In the next subsection it will be proven that the constructed semiseparable matrix will be essentially the same as the semiseparable matrix coming directly from the $QR$-algorithm.

**1.4. Proof of the correctness of the implicit approach.** In this final part of Section 1 we will prove that the matrix arising from the theorem above is essentially the same as a matrix coming from an explicit step of $QR$ to an unreduced semiseparable matrix. First we will briefly combine all the essential information provided in this section to come to an implicit $QR$-algorithm for semiseparable matrices. Suppose we have a given semiseparable matrix $S$, for the unreducedness of the matrix $S$, there are two essential conditions:

An unreduced symmetric semiseparable matrix $S$ satisfies the following demands:

(1) all the blocks $S(i+1:n, 1:i)$ (for $i = 1, \dots, n-1$) have rank equal to 1, this means that there are no zero blocks below the diagonal;
(2) all the blocks $S(i:n, 1:i+1)$ (for $i = 1, \dots, n-1$) have rank strictly higher than 1, this means that on the superdiagonal, no elements are includable in the semiseparable structure.

While executing the algorithm, the second demand only needs to be checked once, because this corresponds to singularities, which cannot be created afterwards. Therefore, before performing a $QR$-step, we transform the matrix $S$ to unreduced form as described in Section 1.1. All the singularities are now removed, we have a matrix $S^{(0)}$. After a step of $QR$ with or without shift, it is possible that the matrix has almost a block diagonal structure, if this is the case, on should apply deflation and continue with the smaller blocks. More on the deflation criterion used, can be found in Chapter 9, Section 5.

A brief description of one step of the implicit $QR$-algorithm:

(1) Calculate the shift $\kappa$.
(2) Apply the first orthogonal transformation $Q_1$ to the matrix $S^{(0)}$: $S^{(1)} = Q_1^T S^{(0)} Q_1$.
(3) Apply deflation to the matrix $S^{(1)}$, if there are zero blocks created in the matrix $S^{(1)}$.
(4) Apply a step of implicit $QR$ to the blocks, deflated from the matrix $S^{(1)}$.
(5) Apply again deflation to resulting blocks from the previous step.

Because of the implicit $Q$-theorem 70, we know that we performed a step of the $QR$-algorithm similar to the original matrix, thereby justifying our implicit approach for performing a step of $QR$ on a semiseparable matrix $S$.

## 2. An implicit $QR$-algorithm for Hessenberg-like matrices

In the previous section we developed an implicit $QR$-algorithm for symmetric semiseparable matrices. This was useful because every symmetric matrix can be transformed into a similar semiseparable one. As nonsymmetric matrices can be transformed via similarity transformations into Hessenberg-like matrices, we will deduce in this section an implicit $QR$-algorithm this class of matrices.

**2.1. The shift $\kappa$.** For Hessenberg-like matrices one can also consider the Rayleigh and Wilkinson shift as for the symmetric case. However, one important remark has to be made. Our Hessenberg-like matrix is not necessarily symmetric anymore, and does not necessarily contain all real eigenvalues. It is perfectly possible to have complex conjugate eigenvalues in the semiseparable matrix. To obtain convergence to these eigenvalues, one can try the following scheme, as presented in [**91**, p. 355], which is called the double shift strategy or the performance of a Francis $QR$-step. We will only present this scheme for Hessenberg matrices. Suppose we have a Hessenberg matrix, for which the lower right two by two block contains two complex conjugate eigenvalues $z_1$ and $z_2$. The double shift strategy performs in fact two $QR$-steps with different shifts:

$$\begin{aligned} H - z_1 I &= Q_1 R_1 \\ H_1 &= RQ + z_1 I \\ H_1 - z_2 I &= Q_2 R_2 \\ H_2 &= R_2 Q_2 + z_2 I. \end{aligned}$$

The resulting Hessenberg matrix should be real but numerical rounding errors can prevent that. Therefore one switches to the implicit approach, in which two steps of $QR$ are applied simultaneously. In this section we will however assume that all our eigenvalues of the Hessenberg-like matrix are real and that we therefore not need this implicit double shift strategy.

**2.2. An implicit $QR$-step on the Hessenberg-like matrix.** Without loss of generality we assume our Hessenberg-like matrix $Z$ to be unreduced. We know that we have to perform two sequences of Givens transformations on the matrix $Z$. The first sequence of Givens transformations $Q_1$ can be performed immediately on both sides of the matrix at the same time. This can be seen from the proof of Theorem 61:

$$Z_1 = Q_1^T Z Q_1.$$

By Theorem 61 the matrix $Z$ is again a Hessenberg-like matrix. The second step, is performed in an implicit way, just like in the symmetric case. To start with the implicit approach, we need an initialization step. We know that $H_1$ is a Hessenberg matrix:

$$\begin{aligned} H_1 &= Q_1^T Z - Q_1^T \kappa I \\ &= R_1 - Q_1^T \kappa I. \end{aligned}$$

This makes it easy to calculate the first Givens $G_n^T$ transformation of the matrix $Q_2$. This transformation $G_n^T$ will annihilate the element $(2,1)$ in the matrix $H_1$. Switching to the implicit approach, we perform the transformation $G_n^T$ on the Hessenberg-like matrix $Z_1$:

$$Z_2 = G_n^T Z_1 G_n.$$

Note that the matrix $Z_2$ is not Hessenberg-like anymore, but has the structure disturbed in the upper left $2 \times 2$ block, with the remaining part of the matrix still of the correct form. We will now design a chasing technique which will remove the disturbance in the matrix.

**2.3. Chasing the disturbance in the matrix $Z_2$.** We will now derive three propositions, that determine Givens transformations, that can be used to reestablish the structure of the Hessenberg-like matrix.

PROPOSITION 75. *Suppose the following* $3 \times 3$ *matrix is given,*

$$A = \begin{pmatrix} a & g & h \\ b & c & i \\ d & e & f \end{pmatrix},$$

*for which* $be - dc \neq 0$. *Then there exists a Givens transformation*

$$G = \begin{pmatrix} 1 & 0 \\ 0 & \hat{G} \end{pmatrix} \quad \textit{with} \quad \hat{G} = \frac{1}{\sqrt{1+t^2}} \begin{pmatrix} t & -1 \\ 1 & t \end{pmatrix}$$

*such that the following matrix*

$$G^T A G$$

*has the lower left* $2 \times 2$ *block of rank* $1$.

PROOF. The proof is similar to the one of Theorem 71. A simple calculation reveals that the parameter $t$ of the Givens transformation

$$t = \frac{id - fb}{eb - dc},$$

is well defined because $eb - dc \neq 0$. $\square$

This is however not enough to restore the complete lower triangular semiseparable structure. Just like in the symmetric case, another type of Givens transformation is needed.

PROPOSITION 76. *Suppose the following* $4 \times 4$ *matrix is given,*

$$A = \begin{pmatrix} a & g & h & j \\ b & c & i & k \\ 0 & 0 & 0 & l \\ d & e & f & m \end{pmatrix},$$

*which is not yet Hessenberg-like. Then there exists a Givens transformation*

$$G = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \hat{G} & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \textit{with} \quad \hat{G} = \frac{1}{\sqrt{1+t^2}} \begin{pmatrix} t & -1 \\ 1 & t \end{pmatrix}$$

*such that the the upper left* 3 *by* 3 *block of the following matrix*

$$G^T A G$$

*is Hessenberg-like. And the lower left* $3 \times 2$ *block is of rank* 1*.*

PROOF. A straightforward calculation reveals:

$$t = \frac{id - bf}{eb - cd},$$

which is well defined. $\square$

In fact the structure given in Proposition 72 is not the most general. The following proposition will capture all the possible transformations.

PROPOSITION 77. *Suppose the following matrix is given,*

$$A = \begin{pmatrix} a & g & \dots & & & \\ b & c & i & \dots & & \\ 0 & 0 & 0 & l & \dots & \\ \vdots & & \vdots & \ddots & \ddots & \\ 0 & \dots & 0 & \dots & 0 & j \\ d & e & f & \dots & & m \end{pmatrix},$$

*which is not yet Hessenberg-like. Then there exists a Givens transformation*

$$G = \begin{pmatrix} 1 & 0 & 0 & \dots \\ 0 & \hat{G} & 0 & \\ 0 & 0 & 1 & \\ \vdots & & & \ddots \end{pmatrix} \quad \textit{with} \quad \hat{G} = \frac{1}{\sqrt{1+t^2}} \begin{pmatrix} t & -1 \\ 1 & t \end{pmatrix}$$

*such that the upper left* 3 *by* 3 *block of the following matrix*

$$G^T A G$$

*is of Hessenberg-like form. And the first two columns are dependent, except for the first element.*

PROOF. A calculation reveals:

$$t = \frac{id - bf}{eb - cd},$$

which is well defined. □

NOTE. This transformation is not necessary in the symmetric case. In the symmetric case only two different types of Givens transformations are needed (see proof of Theorem 73).

Using the propositions above, one can prove in a similar way as in Theorem 74 that one can reduce the disturbed Hessenberg-like matrix $Z_2$ to Hessenberg-like form. Moreover considering the special structure of the matrices involved one can prove, via the implicit $Q$-theorem that we applied a step of $QR$ to the original Hessenberg-like matrix $Z$.

## 3. An implicit $QR$-algorithm for computing the singular values

In this section an adaptation of the $QR$-algorithm of Section 1 will be made such that it becomes suitable for computing the singular values of matrices via upper triangular semiseparable matrices. The adaptations are achieved in a completely similar way as the translation of the $QR$-algorithm for tridiagonal matrices towards the $QR$-algorithm for bidiagonal matrices.

**3.1. The standard $QR$-method for the calculation of the singular values.** This section is based on [**91**, p. 448-460].

Let $A \in \mathbb{R}^{m \times n}$. Without loss of generality, assume that $m \geq n$. In a first phase the matrix $A$ is transformed into a bidiagonal one by orthogonal transformations $U_B$ and $V_B$ to the left and to the right, respectively, i.e.,

$$U_B^T A V_B = \left( \begin{array}{c} B \\ 0 \end{array} \right), \text{ with } B = \left( \begin{array}{ccccc} d_1 & f_1 & 0 & \cdots & 0 \\ 0 & d_2 & \ddots & \ddots & \vdots \\ & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & f_{n-1} \\ 0 & \cdots & & 0 & d_n \end{array} \right).$$

Without loss of generality we can assume that all the elements $f_i$ and $d_i$ in the matrix $B$ are different from zero [**91**, p. 454]. Knowing $B$, the $QR$-method is applied to it. The latter method generates a sequence of bidiagonal matrices converging to a block diagonal one.

Here a short description of one iteration of the $QR$-method is considered. More details can be found in [**91**, p. 452-456].

Let $\tilde{V}_1$ be the Givens rotation

$$\tilde{V}_1 = \left( \begin{array}{cc} c_1 & s_1 \\ -s_1 & c_1 \end{array} \right), \text{ such that } \tilde{V}_1^T \left( \begin{array}{c} d_1^2 - \kappa \\ d_1 f_1 \end{array} \right) = \left( \begin{array}{c} \times \\ 0 \end{array} \right),$$

where $\kappa$ is a shift. We consider here the Wilkinson shift of the matrix $B^T B$. Let $V_1 = \operatorname{diag}(\tilde{V}_1, I_{n-2})$, with $I_k, k \geq 1$, the identity matrix of order $k$. The matrix $B$

is multiplied to the right by $V_1$, introducing the bulge denoted by $\otimes$ in the matrix $BV_1$,

$$BV_1 = \begin{pmatrix} \times & \times & 0 & \cdots & & & \\ \otimes & \times & \times & 0 & \cdots & & \\ 0 & 0 & \times & \times & 0 & \cdots & \\ & & & \ddots & \ddots & & \end{pmatrix}.$$

The remaining part of the iteration is completed applying $2n-3$ Givens rotations, $U_1, \ldots, U_{n-1}$, $V_2, \ldots, V_{n-1}$ moving the bulge along the bidiagonal and eventually removing it,

$$U^T BV = (U_{n-1}^T \cdots U_1^T) B (V_1 V_2 \cdots V_{n-1}) = \begin{pmatrix} \times & \times & & & \\ & \times & \times & & \\ & & \times & \times & \\ & & & \times & \times \\ & & & & \times \end{pmatrix}.$$

Hence, the bidiagonal structure is preserved after one iteration of the $QR$-method. It can be seen that $Ve_1 = V_1 e_1$,where $e_1$ is the first vector of the canonical basis.

We will now relate the chasing algorithm presented above to the $QR$-factorization of the related tridiagonal matrix $B^T B$, because in the next part, we will describe a similar algorithm for upper triangular semiseparable matrices $S_u$ and we will base this algorithm on a $QR$-step applied to the semiseparable matrix $S = S_u^T S_u$.

Suppose we have the bidiagonal matrix $B$ and we want to compute its singular values. A naive way to do this consists of calculating the eigenvalues of the symmetric tridiagonal matrix $T = B^T B$. In fact this is not a good idea, because it will square the condition number. Let us apply now one step of the $QR$-algorithm to this tridiagonal matrix $T$.

$$\left\{ \begin{array}{rcl} B^T B - \kappa I & = & QR \\ \tilde{T} & = & RQ + \kappa I. \end{array} \right.$$

Let us compare this with the transformations applied to the matrix $B$ as described above, $B$ is transformed into another upper bidiagonal matrix $\tilde{B} = U^T BV$. This means that we have the following two equations:

$$\left\{ \begin{array}{rcl} \tilde{T} & = & Q^T (B^T B) Q \\ \tilde{B}^T \tilde{B} & = & V^T (B^T B) V. \end{array} \right.$$

We know that $Qe_1 = Ve_1$. Using the implicit $Q$-theorem for tridiagonal matrices, we get the following result.

THEOREM 78 (Implicit $Q$-theorem, from [**91**, p. 416]). *Suppose $Q = [q_1, \ldots, q_n]$ and $V = [v_1, \ldots, v_n]$ are orthogonal matrices with the property that both $Q^T AQ = T$ and $V^T AV = S$ are tridiagonal where $A \in \mathbb{R}^{n\times n}$ is symmetric. Let $k$ denote the small positive integer for which $t_{k+1,k} = 0$ with the convention that $k = n$ if $T$ is irreducible. If $v_1 = q_1$, then $v_i = \pm q_i$ and $|t_{i,i-1}| = |s_{i,i-1}|$ for $i = 2, \ldots, k$. Moreover, if $k < n$, then $s_{k+1,k} = 0$.*

We know that the matrices $\tilde{T}$ and $\tilde{B}^T \tilde{B}$ are essentially the same. This means that if $\tilde{T}$ converged to an eigenvalue, then $\tilde{B}$ converged to a singular value. Moreover, using the technique as described in the beginning of this section we can calculate the singular values, without explicitely forming the product $B^T B$, which would have squared the condition number of the matrix.

Summarizing, the $QR$-method for computing the singular values of a matrix $A \in \mathbb{R}^{m\times n}$ can be divided into two phases.

**Phase 1** Reduction of $A$ into a bidiagonal matrix $B$ by means of orthogonal transformations.

**Phase 2** The $QR$-method to compute the singular values is applied to $B$.

In the new method, the role of the bidiagonal matrices is played by upper triangular semiseparable matrices, which are the inverses of bidiagonal matrices, as proved in Theorem 17.

For this $QR$-method there are some important issues. First of all the involved bidiagonal matrix is unreduced. Secondly, the knowledge of the first column of the orthogonal factor of the $QR$-factorization of $B^T B - \kappa I$, is very important and will make sure that the implicit $Q$-theorem can be applied afterwards. Moreover, after each iteration of the $QR$-method the bidiagonal structure of the involved matrices is preserved.

In this section we will derive what is meant with an unreduced upper triangular semiseparable matrix. We will prove that one iteration of the implicit $QR$-method applied to an upper triangular semiseparable matrix $S_u$ is uniquely determined by the first column of the orthogonal factor $Q$ of the $QR$-factorization of

$$S_u{}^T S_u - \kappa I, \tag{56}$$

where $\kappa$ is a suitable shift. Exploiting the particular structure of matrix (56), we will show how the first column of $Q$ can be computed without explicitly computing $S_u{}^T S_u$. Finally we will make sure that the resulting matrix after one step of the implicit $QR$-method is again an upper triangular semiseparable matrix.

**3.2. Unreduced upper triangular semiseparable matrices and the shift $\kappa$.** We will use the Wilkinson shift, i.e., it is the eigenvalue of the lower right block of the matrix $S_u^T S_u$, that is closest to the element in the lower right position.

Let us define first what is meant by an unreduced upper triangular semiseparable matrix.

DEFINITION 79. Suppose $S_u$ is an upper triangular semiseparable matrix. $S_u$ is called unreduced if

(1) The ranks of the blocks $S_u(1 : i, i+1 : n)$, for $i = 1, \dots, n-1$ are 1. This means that there are no zero blocks above the diagonal.
(2) All the diagonal elements are different from zero.

NOTE. An unreduced lower triangular semiseparable matrix is defined in a similar way.

Assume that we have an upper triangular matrix $S_u$, and we want to transform it into an unreduced one. Condition 1, can be satisfied rather easily by dividing the matrix into different blocks. This means that the resulting block-matrices are generator representable. From now on, we will take each of these blocks seperately. Assume the generators of such an upper triangular block matrix to be $u$ and $v$. If condition 2 is not satisfied, i.e., a diagonal element $d_i = u_i v_i$ is zero, then either $u_i$ or $v_i$ equals zero. Assume $S_u$ of the following form:

$$S_u = \begin{pmatrix} u_1v_1 & u_2v_1 & u_3v_1 & \cdots & u_nv_1 \\ 0 & u_2v_2 & u_3v_2 & \cdots & u_nv_2 \\ & 0 & u_3v_3 & \cdots & u_nv_3 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & \cdots & & 0 & u_nv_n \end{pmatrix}. \tag{57}$$

Assume $u_i = 0$. This means that we have a matrix of the following form (let us take for example a matrix of size $5 \times 5$, with $u_3 = 0$):

$$\begin{pmatrix} \boxtimes & \boxtimes & 0 & \boxtimes & \boxtimes \\ 0 & \boxtimes & 0 & \boxtimes & \boxtimes \\ 0 & 0 & 0 & \boxtimes & \boxtimes \\ 0 & 0 & 0 & \boxtimes & \boxtimes \\ 0 & 0 & 0 & 0 & \boxtimes \end{pmatrix}.$$

We want to transform this matrix to unreduced form. Let us apply a Givens transformation $G^T$ on row 3 and 4 of the matrix, which will annihilate the complete row 4, because these rows are dependent. Thus we get the following matrix:

$$\begin{pmatrix} \boxtimes & \boxtimes & 0 & \boxtimes & \boxtimes \\ 0 & \boxtimes & 0 & \boxtimes & \boxtimes \\ 0 & 0 & 0 & \boxtimes & \boxtimes \\ 0 & 0 & 0 & \boxtimes & \boxtimes \\ 0 & 0 & 0 & 0 & \boxtimes \end{pmatrix} \xrightarrow{G^T S_u} \begin{pmatrix} \boxtimes & \boxtimes & 0 & \boxtimes & \boxtimes \\ 0 & \boxtimes & 0 & \boxtimes & \boxtimes \\ 0 & 0 & 0 & \boxtimes & \boxtimes \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \boxtimes \end{pmatrix}.$$

In the matrix on the right we can clearly remove the zero row and column, to get an unreduced upper triangular semiseparable matrix when all other diagonal elements are different from zero.

The results applied here can be generalized to $v_i = 0$, by annihilating a column via a Givens transformation on the right, also the extension towards more zeros on the diagonal is straightforward.

**3.3. The new method via upper triangular semiseparable matrices.** In this section we will apply all the necessary transformations to the upper triangular matrix $S_u$ such that the first column of the final orthogonal transformation applied to the right of $S_u$, will be the same as the first column of the matrix $Q$ in the $QR$-factorization in $S_u^T S_u - \kappa I$. We can assume that the matrix $S_u$ is unreduced, and therefore this matrix is representable with two generators $u$ and $v$. Moreover, the matrix will be nonsingular. Let $S_u$ be as in (57). Let $\tau_i = \sum_{k=1}^{i} v_k^2,\ i = 1, \ldots, n$. An easy calculation reveals that

$$S = {S_u}^T S_u = \begin{pmatrix} u_1\,u_1\tau_1 & u_2\,u_1\tau_1 & u_3\,u_1\tau_1 & \cdots & u_n\,u_1\tau_1 \\ u_2\,u_1\tau_1 & u_2\,u_2\tau_2 & u_3\,u_2\tau_2 & & u_n\,u_2\tau_2 \\ u_3\,u_1\tau_1 & u_3\,u_2\tau_2 & u_3\,u_3\tau_3 & & \vdots \\ \vdots & & & \ddots & \\ u_n\,u_1\tau_1 & u_n\,u_2\tau_2 & \cdots & & u_n\,v_n\tau_n \end{pmatrix} \tag{58}$$

is a symmetric semiseparable matrix. Let us denote the generators for this symmetric semiseparable matrix $S$ by $\hat{u}$ and $\hat{v}$,

$$\begin{aligned} \hat{u} = u &= (u_1, u_2, u_3, \cdots, u_n) \\ \hat{v} &= (u_1\tau_1, u_2\tau_2, u_3\tau_3, \cdots, u_n\tau_n). \end{aligned}$$

Therefore $S - \kappa I$ is a diagonal plus semiseparable matrix.

THEOREM 80. *Assume $S_u$ is an unreduced upper triangular semiseparable matrix, then the matrix $S = S_u^T S_u$ will be an unreduced semiseparable matrix. Moreover all the elements in the generators $u$ and $v$ are different from zero.*

PROOF. By equation (58) and knowing, that because of the unreducedness of $S_u$ the elements $u_i$ and $v_i$ are all different from zero, we have no zero blocks in the matrix $S$. Assume however that the semiseparable structure below the diagonal expands above the diagonal. This will lead to a contradiction. Assume for example that the element $u_3 u_2 \tau_2$ can be incorporated in the semiseparable structure below the diagonal. This implies that the $2 \times 2$ matrix

$$\begin{pmatrix} u_2\, u_2\tau_2 & u_3\, u_2\tau_2 \\ u_3\, u_2\tau_2 & u_3\, u_3\tau_3 \end{pmatrix}$$

is of rank 1. This means that

$$u_2\, u_2\tau_2 u_3\, u_3\tau_3 - u_3\, u_2\tau_2 u_3\, u_2\tau_2 = 0$$

this can be simplified, leading to:

$$\tau_3 - \tau_2 = 0$$

which means, that $v_3 = 0$, which is a contradiction. □

We will investigate how the application of one step of $QR$ on the matrix $S - \kappa I$ interacts with the upper triangular matrix $S_u$. Once the first column of the orthogonal transformation applied to the right is completely determined, we switch to the implicit approach. We make use of the structure of the $QR$-factorization as presented in Chapter 7 Section 1. We know that the $Q$ factor of the $QR$-factorization of a diagonal plus semiseparable matrix of order $n$, can be given by the product of $2n-2$ Givens rotations. The first $n-1$ Givens rotations, $G_i, i = 1, \ldots, n-1$, applied from bottom to top transform the semiseparable matrix $S$ into an upper triangular matrix and the diagonal matrix $\kappa I$ into a Hessenberg one. Moreover if $\tilde{Q} = G_1 G_2 \cdots G_{n-1}$, then $S_l = S_u Q_1$ is a lower triangular semiseparable matrix. We have now already performed the first sequence of transformations on the matrix $S - \kappa I$ which corresponds to transforming the upper triangular matrix $S_u$ to lower triangular form $S_l$. We now calculate the first Givens transformation $G_n$ which will annihilate the element in position $(2,1)$ of the Hessenberg matrix $H = Q_1^T(S - \kappa I)$. The knowledge of this Givens transformation will determine the first columnn of the transformation performed on the right of the matrix $S_u$.

Taking into account that $S_l$ is a lower triangular matrix, and

$$\begin{aligned} H &= Q_1^T(S - \kappa I) \\ &= Q_1^T S_u^T S_u - \kappa Q_1^T \\ &= S_l^T S_u - \kappa Q_1^T, \end{aligned}$$

when denoting the Givens transformation embedded in $G_{n-1}$ as

$$\begin{pmatrix} c_{n-1} & -s_{n-1} \\ s_{n-1} & c_{n-1} \end{pmatrix},$$

we have that the first two elements of the first column of $H$ are

$$\begin{pmatrix} (S_l)_{11}(S_u)_{11} - \kappa c_{n-1} \\ \kappa s_{n-1} \end{pmatrix}.$$

This means that we can determine the Givens transformation $G_n$ rather easily.

Moreover, taking a closer look at the first column of $Q = Q_1 Q_2$ we have

$$\begin{aligned} Qe_1 &= Q_1 Q_2 e_1 \\ &= G_1 G_2 \cdots G_{n-1} G_n G_{n+1} \cdots G_{2n-2} e_1 \\ &= G_1 G_2 \cdots G_{n-1} G_n e_1 \end{aligned}$$

since $G_i e_1 = e_1$ for $i = n+1, \ldots, 2n-2$. This means that the first column of the $Q$ factor of the $QR$-factorization of $S - \kappa I$ depends only on the product $G_1 G_2 \cdots G_{n-1} G_n$.

Furthermore, let

$$\tilde{S}_l = S_u G_1 G_2 \cdots G_{n-1} G_n = S_l G_n. \tag{59}$$

If we do not perform any transformations to the right of the matrix anymore involving the first column, we will be able to apply the implicit $Q$-theorem. From now on we will start with the implicit approach. The bulge, just created by the Givens transformation $G_n$, will be chased by a sequence of Givens transformations performed on the left and the right of the matrix.

**3.4. Chasing the bulge.** The matrix (59) differs from a lower triangular semiseparable matrix, because there is a bulge in position $(1, 2)$. In order to retrieve the lower triangular semiseparable structure an algorithm is presented in this section. At each step of the algorithm the bulge is chased down one position along the superdiagonal, by applying orthogonal transformations to $\tilde{S}_l$. Only orthogonal transformations with the first column equal to $e_1$ are applied to the right of $\tilde{S}_l$. Before describing the algorithm, we consider the following proposition.

PROPOSITION 81. *Let*

$$C = \begin{pmatrix} u_1 v_1 & \alpha & 0 \\ u_2 v_1 & u_2 v_2 & u_2 v_3 \end{pmatrix} \tag{60}$$

*with $u_1, u_2, v_1, v_2, v_3$ all different from zero. Then there exists a Givens transformation $G$ such that,*

$$\tilde{C} = C \begin{pmatrix} 1 & 0 \\ 0 & G \end{pmatrix} \tag{61}$$

*has a linear dependency between the first two columns of $\tilde{C}$.*

PROOF. The theorem is proven by explicitly constructing the Givens transformation $G$

$$G = \frac{1}{\sqrt{1+t^2}} \begin{pmatrix} t & -1 \\ 1 & t \end{pmatrix}. \tag{62}$$

Taking (60) and (62) into account, (61) can be written in the following way,

$$\tilde{C} = \frac{1}{\sqrt{1+t^2}} \begin{pmatrix} u_1v_1\sqrt{1+t^2} & t\alpha & -\alpha \\ u_2v_1\sqrt{1+t^2} & tu_2v_2 + u_2v_3 & -u_2v_2 + tu_2v_3 \end{pmatrix}.$$

Dependency between the first two columns leads to the following condition on the coefficients of the previous matrix:

$$t\alpha \left(\sqrt{1+t^2}\, u_2v_1\right) = \left(\sqrt{1+t^2}\, u_1v_1\right)(tu_2v_2 + u_2v_3)$$

Simplification leads to:

$$t\alpha u_2 = u_1 \left(tu_2v_2 + u_2v_3\right).$$

Extracting the factor $t$ out of the previous equation proves the existence of the Givens transformation $G$:

$$t = \frac{v_3}{\left(\frac{\alpha}{u_1} - v_2\right)}.$$

The denominator is clearly different from zero. Otherwise the left 2 by 2 block of the matrix in equation (60) would have been already of rank 1 and we would have chosen the Givens transformation equal to the identity matrix.

□

The next theorem yields the algorithm that transforms $\hat{S}_l$ into a lower triangular semiseparable matrix.

THEOREM 82. *Let $\hat{S}_l$ be an unreduced lower triangular semiseparable matrix. For which the strictly upper triangular part is zero except for the entry* $(1,2)$. *Then there exist two orthogonal matrices $\tilde{U}$ and $\tilde{V}$ such that*

$$\tilde{S}_l = \tilde{U}^T \hat{S}_l \tilde{V}$$

*is a lower triangular semiseparable matrix and $\tilde{V}e_1 = e_1$.*

PROOF. The theorem is proven by constructing an algorithm which transforms $\hat{S}_l$ into a lower triangular semiseparable matrix, in which the orthogonal transformations applied to the right have the first column equal to $e_1$. Without loss of generality we assume $\hat{S}_l \in \mathbb{R}^{5\times 5}$. Let

$$\hat{S}_l = \begin{pmatrix} \boxtimes & \otimes & 0 & 0 & 0 \\ \boxtimes & \boxtimes & 0 & 0 & 0 \\ \boxtimes & \boxtimes & \boxtimes & 0 & 0 \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & 0 \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \boxtimes \end{pmatrix},$$

where $\otimes$ denotes the bulge to be chased. Moreover, the entries of the matrix satisfying the semiseparable structure are denoted by $\boxtimes$. At the first step a Givens transformation $\tilde{U}_1^T$ is applied to the left of $\hat{S}_l$ in order to annihilate the bulge,

$$\tilde{U}_1^T \hat{S}_l = \begin{pmatrix} \times & 0 & 0 & 0 & 0 \\ \times & \times & 0 & 0 & 0 \\ \boxtimes & \boxtimes & \boxtimes & 0 & 0 \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & 0 \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \boxtimes \end{pmatrix}. \tag{63}$$

Although $\tilde{U}_1^T \hat{S}_l$ is lower triangular, the semiseparable structure is lost in its first two rows. In order to retrieve it, a Givens rotation $\tilde{V}_1$, constructed according to Theorem 81, and acting to the second and the third column of $\tilde{U}_1^T \hat{S}_l$ is applied to the right, in order to make the first two columns, in the lower triangular part, proportional.

$$\tilde{U}_1^T \hat{S}_l = \left(\begin{array}{ccccc} \boxtimes & 0 & 0 & 0 & 0 \\ \times & \times & 0 & 0 & 0 \\ \boxtimes & \boxtimes & \boxtimes & 0 & 0 \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & 0 \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \boxtimes \end{array}\right) \xrightarrow{\tilde{U}_1^T \hat{S}_l \tilde{V}_1} \left(\begin{array}{ccccc} \boxtimes & 0 & 0 & 0 & 0 \\ \boxtimes & \boxtimes & \otimes & 0 & 0 \\ \boxtimes & \boxtimes & \boxtimes & 0 & 0 \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & 0 \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \boxtimes \end{array}\right).$$

Hence, applying $\tilde{U}_1^T$ and $\tilde{V}_1$ to the right and to the left of $\hat{S}_l$, respectively, the bulge is moved one position down the superdiagonal (see Fig. 8.5), retrieving the semiseparable structure in the lower triangular part. Recursively applying the latter procedure the matrix $\tilde{U}_4^T \cdots \tilde{U}_1^T \hat{S}_l \tilde{V}_1 \cdots \tilde{V}_4$ will be lower triangular semiseparable matrix. Then the theorem holds choosing $\tilde{U}^T = \tilde{U}_4^T \cdots \tilde{U}_1^T$ and $\tilde{V} = \tilde{V}_1 \cdots \tilde{V}_4$.

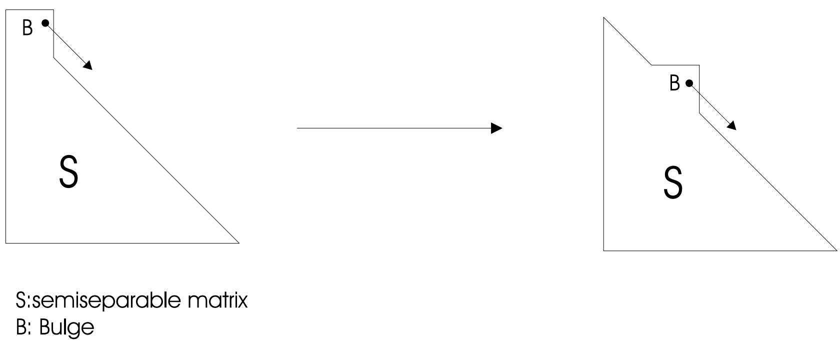


FIGURE 8.5: Graphical representation of the chasing

□

**3.5. One iteration of the new method applied to upper triangular semiseparable matrices.** In this section we describe one iteration of the proposed algorithm for computing the singular values of an upper triangular semiseparable matrix. Moreover we prove the equivalence between the latter iteration and one iteration of the $QR$-method applied to an upper triangular semiseparable matrix. Before we start with the $QR$-iteration the corresponding upper triangular semiseparable matrix needs to be in unreduced form. We have to apply deflation. One iteration of the proposed method consists of the following 4 steps.

Step 1. $n-1$ Givens transformations $G_1, \ldots, G_{n-1}$ are performed to the right of $S_u$ swapping it into a lower triangular semiseparable matrix $S_l$.

Step 2. One more Givens transformation $G_n$ is computed in order to introduce the shift. As seen in Section 3.4, the application of this Givens transformation to the right of the lower triangular semiseparable matrix creates a bulge.

Step 3. The bulge is chased by $n-2$ Givens transformations, $G_{n+1}, \ldots, G_{2n-2}$ on the left of the matrix, and also $n-2$ Givens transformations on the right retrieving the lower triangular semiseparable structure.

Step 4. The latter matrix is swapped back to upper triangular form. This is done by applying $n-1$ more Givens transformations to the left, without destroying the semiseparable structure.

Step 4. Apply deflation to the resulting upper triangular semiseparable matrix.

NOTE. Actually, the Givens rotations of Step 1 and Step 4 are not explicitely applied to the matrix, the semiseparable structure allows us to calculate the resulting semiseparable matrix (the representation of this matrix) in a very cheap way with $O(n)$ flops.

NOTE. If the shift is not considered, only Step 1 and Step 4 are performed at each iteration of the method.

We still have to prove that one iteration of the latter method is equivalent to one iteration of the $QR$-method applied to an upper triangular semiseparable matrix $S_u$, i.e. , if $\tilde{Q}$ is the orthogonal factor of the $QR$-factorization of $S_u^T S_u - \kappa I = S - \kappa I$, and $Q$ is the matrix of the product of the orthogonal matrices applied to the right of $S_u$ during one iteration of the proposed method, then $\tilde{Q}$ and $Q_2$ have the same columns up to the sign, i.e.,

$$\tilde{Q} = Q \operatorname{diag}(\pm 1, \ldots, \pm 1).$$

Assume a $QR$-step is performed on the matrix $S$. This can be written in the following form:

$$\tilde{Q}^T S \tilde{Q} = \tilde{S}. \tag{64}$$

Now we show how one iteration of the new method performed on the matrix $S_u$ can also be rewritten in terms of the matrix $S = {S_u}^T S_u$. This is achieved in the following way. (All the steps mentioned in the beginning of the section are now applied to the matrix $S_u$.) First the $n-1$ Givens transformations are performed on the matrix $S_u$, transforming the matrix into lower triangular semiseparable form. In the following equations, all the transformations performed on the matrix $S_u$ are fitted in the equation $S = {S_u}^T S_u$ to see what happens to the matrix $S$.

$$\begin{aligned} S_l^T S_l &= \left(G_{n-1}^T \cdots G_1^T {S_u}^T\right)\left(S_u G_1 \cdots G_{n-1}\right) \\ &= G_{n-1}^T \cdots G_1^T S G_1 \cdots G_{n-1}. \end{aligned}$$

One more Givens transformation $G_n$ is now applied, introducing a bulge in the matrix $S_l$.

$$G_n^T S_l^T S_l G_n = G_n^T G_{n-1}^T \cdots G_1^T S G_1 \cdots G_{n-1} G_n.$$

Taking Theorem 81 into account, there exist two orthogonal matrices $U$ and $V$, with $V e_1 = e_1$, such that $U^T S_l G_n V$ is again a lower triangular semiseparable matrix.

This leads to the equation:

$$V^T G_n^T S_l^T U U^T S_l G_n V = V^T G_n^T G_{n-1}^T \cdots G_1^T S G_1 \cdots G_{n-1} G_n V.$$

The final step consists of swapping the lower triangular matrix $U^T S_l G_n V^T$ back into upper triangular semiseparable form. This is accomplished by applying $n-1$ more Givens rotations to the left of $U^T S_l G_n V$. Thus, denoted by $Q = G_1 \cdots G_{n-1} G_n V$ and taking (58) into account, we have

$$Q^T S_u^T S_u Q = S^{(2)} \tag{65}$$

with $Q$ orthogonal and $\hat{S}$ a semiseparable matrix.

We observe that $\tilde{Q} e_1 = Q e_1$. This holds because the first $n$ Givens transformations are the same Givens transformations as performed in a $QR$-step on $S$, and the fact that $V e_1 = e_1$.

One can apply the implicit $Q$ theorem now (Theorem 70), which justifies our approach. Hence we developed a $QR$-variant for upper triangular semiseparable matrices. Combined with the results of Part 2, we can now compute the singular values of not only upper triangular semiseparable matrices, but also of arbitrary matrices.

## Conclusions

In this chapter we developed implicit $QR$-steps for symmetric semiseparable and Hessenberg-like matrices. Combined with the tools provided in the previous part and the theoretical results of the previous chapter we have now an algorithm to compute the eigenvalues of matrices via semiseparable matrices. Moreover, we also translated the symmetric eigenvalue solver towards a method for calculating the singular values via intermediate upper triangular semiseparable matrices.

CHAPTER 9

# Implementations of $QR$-related algorithms and numerical experiments

In the previous two chapters different theoretical results were combined, in order to build an implicit $QR$-algorithm for semiseparable matrices. We know how such an algorithm should be designed. The only unsolved problem yet is the implementation. For the implementation we choose the Givens-vector representation as defined in Chapter 2. In the previous chapters, nothing was mentioned about the deflation criterion, i.e., when do we assume that an eigenvalue is accurate enough, or even more, when should we deflate an almost block diagonal matrix into several blocks. This problem, together with a fast computation for the norms of the off-diagonal blocks is solved in this chapter too. The final important problem related to the eigenvalue decomposition is the calculation of the eigenvectors. Two different possible ways will be explained on how to compute them.

In the first section of this chapter an $O(n)$ implementation will be given to solve diagonal plus semiseparable systems of equations via the $QR$-factorization of the diagonal plus semiseparable matrix. As we know from the previous chapter the $QR$-factorization consists of applying twice a sequence of $n-1$ Givens transformations. The algorithm is therefore also divided in two separate parts. In a first part, the first sequence of Givens transformations is applied on three matrices: the lower triangular part of the semiseparable matrix, the strictly upper triangular part of the semiseparable matrix and the diagonal matrix. Recombining these three matrices shows that the resulting Hessenberg matrix has the upper triangular part of rank 2. This upper triangular part will be represented by two Givens-vector representations. The application of the second sequence of Givens transformations will transform the Hessenberg matrix into a strictly upper triangular matrix of rank 2, represented by two Givens-vector representations, and a diagonal. Using this representation of the upper triangular matrix, the system can be solved via backward substitution in order $O(n)$.

In the second section, we will design the implementation of a $QR$-step on a symmetric semiseparable matrix without shift. This corresponds to performing a similarity transformation, with the first sequence of Givens transformations as designed in Section 1. Instead of performing first all the transformations on the left side of the semiseparable matrix, and then performing all the transformations on the right side of the resulting semiseparable matrix, we will perform the transformations simultaneously. This means that at the same time as performing the transpose of a

Givens transformation on the left, we will perform the Givens transformation also on the right. The details of the implementation are given in this section.

Section 3 discusses the reduction of a matrix to the unreduced form. Attention is paid to a criterion stating wether or not the semiseparable structure extends above the diagonal.

Section 4 gives the details on the implementation of an implicit $QR$-step on a symmetric semiseparable matrix, with shift. As the $QR$-step with shift, initially performs a $QR$-step without shift, we assume that the matrix we start with has the structure of the resulting matrix from the previous section. The $QR$-step with shift is initialized by performing the similarity Givens transformation $G_n$, which will divide the matrix in two semiseparable matrices: one semiseparable matrix of dimension 2 by 2 at the upper left position, and one semiseparable matrix of dimension $n-1$ by $n-1$ at the lower right position. Both of the semiseparable matrices will be represented by the Givens-vector representation. The consecutive Givens transformations performed on the matrix, will increase the dimension of the upper left semiseparable matrix by one at each transformation, and they will decrease the dimension of the lower right semiseparable matrix by one at each step. At each step therefore the Givens-vector representation of the upper left and lower right semiseparable matrix have to be updated.

Deflation is the subject of following section. It is investigated when we should divide the symmetric semiseparable matrix in different blocks, such that all the eigenvalues of these blocks approximate very well the eigenvalues of the original matrix. In the tridiagonal approach the original matrix is divided, if the corresponding subdiagonal elements are close enough to zero. This is logical because the subdiagonal elements contain all the information of the corresponding lower left block, which only has the upper right element, which is the subdiagonal element, different from zero. In our semiseparable case however the lower left blocks are full. Calculating the norms of all of the lower right blocks, to see whether they are close enough to zero, is a very slow operation. Fortunately we can compute all the norms of the off-diagonal blocks in $O(n)$. Also two different criterions whether an off-diagonal block is small enough or not, are discussed.

In several cases, not only the eigenvalues are desired, but sometimes also some or all of the eigenvectors are needed. Section 6 considers two possibilities for calculating the eigenvectors of the symmetric semiseparable matrix. The first method consists of performing all the orthogonal transformations, needed to compute all the eigenvalues via the implicit $QR$-algorithm, on the identity matrix. In this way one constructs a matrix $Q$, for which the columns are the eigenvectors of the original semiseparable matrix. The second method is preferable if only few of the eigenvectors are needed. This method calculates via inverse iteration the eigenvectors connected to a particular eigenvalue. To compute the eigenvectors of an arbitrary symmetric matrix $A$, which first needs to be transformed to semiseparable form, one needs to store the orthogonal transformation used to transform the matrix $A$ into semiseparable form.

Section 7 briefly discusses the implementation of the implicit $QR$-algorithm for calculating the singular values and the implicit $QR$-algorithm for Hessenberg-like matrices.

In Section 8 different numerical experiments are performed on the eigenvalue solver based on semiseparable matrices. Arbitrary matrices are transformed to a similar symmetric semiseparable one and afterwards the implicit $QR$-algorithm is applied to the semiseparable matrix in order to calculate its eigenvalues. Four different experiments are reported. A first experiment investigates an almost block matrix and compares the tridiagonal approach with the semiseparable one. In a second experiment a matrix is constructed, for which the eigenvalues form a stair. After the reduction to tridiagonal and the corresponding reduction to the semiseparable form, the diagonal elements of both the tridiagonal and the semiseparable matrix are compared. In a third experiment the eigenvalues of two difficult matrices are calculated via the semiseparable and the tridiagonal approach. In a last experiment the accuracy of the eigenvalues, computed by two different deflation criterions, using the semiseparable approach is compared.

In the last section several numerical experiments are performed using the algorithm to compute the singular values via intermediate upper triangular semiseparable matrices. Two types of experiments are performed. In one type of experiment the accuracy between the bidiagonal and the upper triangular semiseparable approach are compared. In the second experiment the number of $QR$-steps needed to compute all the singular values, via both the algorithms are compared.

## 1. The implementation of the $QR$-factorization of semiseparable plus diagonal matrices

Here we will describe the $O(n)$ implementation to solve a system of equations where the coefficient matrix is a semiseparable plus diagonal matrix. The implementation we present computes first the $QR$-factorization of the semiseparable plus diagonal matrix and will be used to solve the system via backward substitution. First we recall the $QR$-solver and the algorithm as introduced in [**178**]. Suppose a semiseparable matrix $S$ and a diagonal matrix $D$ are given. The first sequence of Givens transformations applied to the matrix $S$ will transform the matrix into an upper triangular matrix $\tilde{R} = Q_1^T S$. Moreover, because the matrix we started from was a semiseparable matrix, the upper triangular factor $\tilde{R}$, will have semiseparability rank 2. Performing now these $n-1$ Givens transformations on the diagonal $D$ this results in a Hessenberg matrix $\tilde{H}$, for which the upper triangular part is of semiseparability rank 1. We have even more structure because the transformations performed on the semiseparable and the diagonal are the same, we can combine the Hessenberg matrix $\tilde{H}$ and $\tilde{R}$ to a matrix whose upper triangular part has semiseparability rank 2.

This can briefly be formulated into a theorem

THEOREM 83 (From [**178**]). *Suppose a semiseparable matrix $S$ and a diagonal $D$ are given, then there exists a sequence of Givens transformations $G_1, \ldots, G_{n-1}$, such that the following equality holds:*

$$G_{n-1}^T \cdots G_1^T \left(S+D\right) = \tilde{R} + \tilde{H} = H,$$

*where $\tilde{R}$ is an upper triangular semiseparable matrix with semiseparability rank 2. $\tilde{H}$ denotes a Hessenberg matrix with the upper triangular part of semiseparability rank 1. The matrix $H$ is a Hessenberg matrix, whose upper triangular part is semiseparable and of semiseparability rank 2.*

As we want an upper triangular matrix and not a Hessenberg one, a final sequence of Givens transformations $Q_2^T$ is needed, to annihilate the subdiagonal elements of the Hessenberg matrix $H$. These Givens transformations are performed from top to bottom. Moreover, because these Givens transformations are performed on the matrix $H$ the upper triangular semiseparable structure will be destroyed a little bit. Instead of an upper triangular part of semiseparability rank 2, we have now a strictly upper triangular part of semiseparability rank 2. This knowledge about the structure makes it possible to implement this algorithm in $O(n)$ operations and to solve systems with semiseparable plus diagonal matrices in a fast way. We will now construct in more detail the implementation, which calculates the $QR$-factorization of a semiseparable matrix, and solves a system of equations with it.

For the implementation the semiseparable plus diagonal matrix is not split up in two parts, as in the theoretical overview, but into three separate parts, namely the lower triangular part of the semiseparable matrix, the strictly upper triangular part of the semiseparable matrix and a diagonal. In fact we have the following decomposition

$$\begin{aligned}
& S+D \\
= \;& L_S + U_S + D \\
= \;& \begin{pmatrix}
c_1 d_1 & 0 & 0 & \dots & 0 & 0 \\
c_2 s_1 d_1 & c_2 d_2 & 0 & & \vdots & \vdots \\
c_3 s_2 s_1 d_1 & c_3 s_2 d_2 & c_3 d_3 & \ddots & & \\
\vdots & & \ddots & \ddots & 0 & 0 \\
c_{n-1} s_{n-2} \cdots s_1 d_1 & \dots & & & c_{n-1} d_{n-1} & 0 \\
s_{n-1} s_{n-2} \cdots s_1 d_1 & \dots & & & s_{n-1} d_{n-1} & d_n
\end{pmatrix} \\
+ \;& \begin{pmatrix}
0 & r_1 e_1 & r_2 t_1 e_1 & \dots & r_{n-2} t_{n-3} \cdots t_1 e_1 & t_{n-2} t_{n-3} \cdots t_1 e_1 \\
0 & 0 & r_2 e_2 & & \vdots & \vdots \\
0 & 0 & 0 & \ddots & & \\
\vdots & & \ddots & \ddots & r_{n-2} e_{n-2} & t_{n-2} e_{n-2} \\
0 & \dots & & & 0 & e_{n-1} \\
0 & \dots & & & 0 & 0
\end{pmatrix} \\
+ \;& \begin{pmatrix}
\hat{d}_1 & & & \\
& \hat{d}_2 & & \\
& & \ddots & \\
& & & \hat{d}_n
\end{pmatrix}.
\end{aligned}$$

In the previous splitting of the semiseparable plus diagonal matrix, we already assumed that our semiseparable parts are represented with the Givens-vector representation. For the lower triangular semiseparable matrix we denote the representation with $G$ and $d$, for the strictly upper triangular part we use $H$ and $e$:

$$\begin{aligned} G &= \begin{pmatrix} c_1 & c_2 & \dots & c_{n-1} \\ s_1 & s_2 & \dots & s_{n-1} \end{pmatrix} \\ d &= (d_1, d_2, \dots, d_n) \\ H &= \begin{pmatrix} r_1 & r_2 & \dots & r_{n-2} \\ t_1 & t_2 & \dots & t_{n-2} \end{pmatrix} \\ e &= (e_1, e_2, \dots, e_{n-1}). \end{aligned}$$

The Givens transformations $G_i$ and $H_i$ are denoted as

$$G_i = \begin{pmatrix} c_i & -s_i \\ s_i & c_i \end{pmatrix} \quad H_i = \begin{pmatrix} r_i & -t_i \\ t_i & r_i \end{pmatrix}.$$

Let us denote with $\hat{d}_i$ the diagonal elements of the matrix $D$. As we already used $G_i$ for the Givens transformations involved in the Givens-vector representation of the matrix we denote with $\tilde{G}_i$ the Givens transformations from the $QR$-factorization. These Givens transformations satisfy $\tilde{G}_1 = G_{n-1}, \dots, \tilde{G}_{n-1} = G_1$. Applying $\tilde{G}_1^T, \dots, \tilde{G}_{n-1}^T$ to the lower triangular semiseparable matrix from bottom to top, will transform this matrix into an upper triangular semiseparable matrix. We take a closer look at the matrix $L_S + D$ after applying the Givens transformation $\tilde{G}_1^T$ to the left. We get the following structure for the lower triangular matrix $L_S$:

$$\begin{pmatrix} c_1 d_1 & 0 & 0 & \dots & & 0 & 0 \\ c_2 s_1 d_1 & c_2 d_2 & 0 & & & \vdots & \vdots \\ c_3 s_2 s_1 d_1 & c_3 s_2 d_2 & c_3 d_3 & \ddots & & & \\ \vdots & & \ddots & \ddots & & 0 & 0 \\ c_{n-2} s_{n-3} \cdots s_1 d_1 & \dots & & & c_{n-2} d_{n-2} & 0 & 0 \\ s_{n-2} s_{n-3} \cdots s_1 d_1 & \dots & & & s_{n-2} d_{n-2} & d_{n-1} & s_{n-1} d_n \\ 0 & \dots & & & & 0 & c_{n-1} d_n \end{pmatrix}.$$

The diagonal matrix $D$ will change in the following way:

$$\begin{pmatrix} \hat{d}_1 & & & & \\ & \hat{d}_2 & & & \\ & & \ddots & & \\ & & & c_{n-1}\hat{d}_{n-1} & s_{n-1}\hat{d}_n \\ & & & -s_{n-1}\hat{d}_{n-1} & c_{n-1}\hat{d}_n \end{pmatrix}.$$

It can be seen now that the upper triangular zero part is slowly filled with elements depending on each other. In fact a semiseparable part is created from bottom to top, for which the Givens transformations in the Givens-vector representation are the Givens transformations $G_{n-1}, \dots, G_1$. One more Givens transformation $\tilde{G}_2^T$ is applied such that we can also see the pattern in the expansion for the elements in the vector of the new representation. After applying the transformation $\tilde{G}_2^T$ on the rows $n-1$ and $n-2$ we get the following matrices:

$$\begin{pmatrix} c_1 d_1 & 0 & 0 & \dots & & 0 & 0 \\ c_2 s_1 d_1 & c_2 d_2 & 0 & & & \vdots & \vdots \\ c_3 s_2 s_1 d_1 & c_3 s_2 d_2 & c_3 d_3 & \ddots & & & \\ \vdots & & \ddots & \ddots & & 0 & 0 \\ s_{n-3} s_{n-4} \cdots s_1 d_1 & \dots & & & d_{n-2} & s_{n-2} d_{n-1} & s_{n-2} s_{n-1} d_n \\ 0 & \dots & & & 0 & c_{n-2} d_{n-1} & c_{n-2} s_{n-1} d_n \\ 0 & \dots & & & & 0 & c_{n-1} d_n \end{pmatrix}$$

and

$$\begin{pmatrix} \hat{d}_1 & & & & & \\ & \hat{d}_2 & & & & \\ & & \ddots & & & \\ & & & c_{n-2} \hat{d}_{n-2} & s_{n-2} \left( c_{n-1} \hat{d}_{n-1} \right) & s_{n-2} s_{n-1} \hat{d}_n \\ & & & -s_{n-2} \hat{d}_{n-2} & c_{n-2} \left( c_{n-1} \hat{d}_{n-1} \right) & c_{n-2} s_{n-1} \hat{d}_n \\ & & & & -s_{n-1} \hat{d}_{n-1} & c_{n-1} \hat{d}_n \end{pmatrix}.$$

After all the Givens transformations $G_i$ are applied on the matrix $L_S + D$, the subdiagonal elements from top to bottom are the following ones:

$$\left( -s_1 \hat{d}_1, \dots, -s_{n-2} \hat{d}_{n-2}, -s_{n-1} \hat{d}_{n-1} \right).$$

The new vector in the Givens-vector representation from bottom to top has the following form:

$$\tilde{d} = \left( d_n + \hat{d}_n, d_{n-1} + c_{n-1} \hat{d}_{n-1}, d_{n-2} + c_{n-2} \hat{d}_{n-2}, \dots, d_2 + c_2 \hat{d}_2, d_1 + c_1 \hat{d}_1 \right).$$

The procedure described above is an $O(n)$ procedure, when implementing everything with the Givens-vector representation. However, we did not yet took into consideration the strictly upper triangular part $U_S$. We know that this matrix has the representation from left to right. This is an advantage in the next procedure, because the Givens transformations are performed on the left, they will not change the dependencies between the columns. The rough idea will be, to apply a Givens transformation, which will clearly disturb the semiseparable structure, because of fill-in in the diagonal. However, because still a following sequence of Givens transformations is needed , we can take out a term in the corresponding diagonal element of $\tilde{d}$ and add it to our transformed matrix $U_S$, such that the dependency between the columns is maintained. In this way, we will create an upper triangular semiseparable matrix for which the dependencies between the columns did not change, only the new diagonal elements need to be calculated, this is illustrated in the following equations.

Applying the Givens transformation $\tilde{G}_1^T = G_{n-1}^T$ on the left of the matrix $U_S$ we get the following structure:

$$\left(\begin{array}{cccccc} 0 & r_1e_1 & r_2t_1e_1 & \dots & r_{n-2}t_{n-3}\cdots t_1e_1 & t_{n-2}t_{n-3}\cdots t_1e_1 \\ 0 & 0 & r_2e_2 & & \vdots & \vdots \\ 0 & 0 & 0 & \ddots & & \\ \vdots & & \ddots & \ddots & r_{n-2}e_{n-2} & t_{n-2}e_{n-2} \\ 0 & \dots & & & 0 & c_{n-1}e_{n-1} \\ 0 & \dots & & & 0 & -s_{n-1}e_{n-1} \end{array}\right)$$

To create an upper triangular semiseparable matrix, an element $x_{n-1}$ has to be added in position $(n-1, n-1)$ such that the last two columns will satisfy again the semiseparable structure. This means in fact that an element $x_{n-1}$ has to be added such that the following block will be of rank 1:

$$\left(\begin{array}{cc} r_{n-2}e_{n-2} & t_{n-2}e_{n-2} \\ x_{n-1} & c_{n-1}e_{n-1} \end{array}\right).$$

When doing so, the corresponding element of $\tilde{d}$, namely $\tilde{d}_{n-1}$ has to be diminished with this term $x_{n-1}$. (Note that this can be done without any problem, because the Givens transformations applied to the left are the same for $L_S$ and $U_S$.)

We can now continue and perform the next Givens transformation $\tilde{G}_2$, on the matrix in which we added the element $x_{n-1}$ (Only the last four rows and columns are shown):

$$\left(\begin{array}{cccc} 0 & r_{n-3}e_{n-3} & r_{n-2}t_{n-3}e_{n-3} & t_{n-2}t_{n-3}e_{n-3} \\ 0 & 0 & c_{n-2}r_{n-2}e_{n-2} + s_{n-2}x_{n-1} & c_{n-2}t_{n-2}e_{n-2} - s_{n-2}c_{n-1}e_{n-1} \\ 0 & 0 & c_{n-2}x_{n-1} - s_{n-2}r_{n-2}e_{n-2} & c_{n-2}c_{n-1}e_{n-1} - s_{n-2}t_{n-2}e_{n-2} \\ 0 & 0 & 0 & -s_{n-1}e_{n-1} \end{array}\right).$$

We can continue this process and calculate $x_{n-2}$. Using this procedure, one can design an $O(n)$ calculation transforming $U_S$ into an upper triangular semiseparable part having hte following representation from left to right. For the Givens transformations we have:

$$\tilde{H} = \left(\begin{array}{ccccc} 0 & r_1 & r_2 & \dots & r_{n-2} \\ 1 & t_1 & t_2 & \dots & t_{n-2} \end{array}\right)$$

and the new diagonal elements, denoted with $\tilde{e}$, are:

$$\left(\frac{c_1r_1e_1 + s_1x_2}{r_1}, \frac{-s_1r_1e_1 + c_1x_2}{r_1}, \dots, \frac{-s_{n-2}r_{n-2}e_{n-2} + c_{n-2}x_{n-1}}{r_{n-2}}, -s_{n-1}e_{n-1}\right).$$

One should not forget to substract from the elements of the vector $\tilde{d}$ the corresponding elements $x_i$.

NOTE. The implementation as designed here, presumes that we are in the ideal case, i.e., there are no divisions by zero, all the elements are different from zero etc. The practical implementation, is changed a little bit, such that also the not ideal situations are dealt with in a correct way.

The final step consists of transforming the Hessenberg matrix into an upper triangular matrix. We will not describe this step in detail like in the previous ones, because it is a straightforward generalization of the Givens transformations applied to $U_S$. First we need to swap the representation of $G$, and $\tilde{d}$ towards a representation from left to right. In this way, both the representations of the upper triangular part of the Hessenberg matrix are from left to right. The application of the second sequence of Givens transformations to annihilate the subdiagonal elements of the Hessenberg matrix, will therefore not change the dependencies between the columns, but only the diagonal will be disturbed. Therefore we exclude the diagonal, and calculate the new Givens-vector representation of the strictly upper triangular part of the matrix $R$. In this way we get a diagonal and two Givens-vector representations which will represent the strictly upper triangular part.

Solving a linear system of equations with the $QR$-algorithm for semiseparable matrices is an easy application of the above decomposition. First we calculate the $QR$-factorization of $S+D = QR$. Suppose we want to solve the system $(S+D)x = b$. This is equivalent to solving $Rx = Q^T b$, and then solving the final part via backward substitution. This final backward substitution can also be implemented in $O(n)$. To do so the same techniques as described in Section 8.2 have to be applied.

This algorithm can be used for example to calculate eigenvectors corresponding to particular eigenvalues, via inverse iteration.

## 2. Effective $O(n)$ implementation of the implicit $QR$-algorithm without shift for symmetric semiseparable matrices

The algorithmic description presented here will describe one step of the $QR$-algorithm, applied in an implicit way to semiseparable matrices. Later on in this chapter, we will discuss the calculation of the eigenvectors and the cutting criterion. Moreover, one should be aware, that the combination of this algorithm together with the reduction to a symmetric semiseparable matrix, gives a tool to compute the eigendecomposition of arbitrary symmetric matrices. How to combine these algorithms is obvious and will not be explained. The implementation can be downloaded from `http://www.cs.kuleuven.ac.be/~marc/`. Not all the details of the implementation are given, but the mathematical ideas, behind the design are included. They should make it possible for the reader to implement the algorithm.

First we have to perform a sequence of Givens transformations from bottom to top on the rows, and at the same time from right to left on the columns. This corresponds to performing a $QR$-step without shift on the original symmetric semiseparable matrix.

Suppose our semiseparable matrix $S$ is built up with the Givens transformations $G$ and the vector elements $d$. We will now perform the first $n-1$ Givens transformations on both sides of the matrix, and we will retrieve the representation of the resulting semiseparable matrix. The matrix $S$ has the following structure:

$$\left( \begin{array}{ccc} S^{(n-2)} & c_{n-1}R_{n-1}^T & s_{n-1}R_{n-1}^T \\ c_{n-1}R_{n-1} & c_{n-1}d_{n-1} & s_{n-1}d_{n-1} \\ s_{n-1}R_{n-1} & s_{n-1}d_{n-1} & d_n \end{array} \right) \tag{66}$$

where $S^{(n-2)}$ represents a semiseparable matrix of order $n-2$ and the Givens transformations are denoted in the usual manner. We denote the Givens transformations used in the $QR$-factorization with $\tilde{G}_i$. We remark that the Givens transformations needed for the first step of the $QR$-algorithm are exactly the same Givens transformations from the representation, more precisely we have the following equivalences $\tilde{G}_1 = G_{n-1}, \ldots, \tilde{G}_{n-1} = G_1$. This is an advantage, because these Givens transformations do not need to be calculated anymore.

Applying the first transformation $\tilde{G}_1^T = G_{n-1}^T$ on the left of matrix (66) gives us the following equations

$$\hat{d}_1 \quad = \quad -s_{n-1}^2 d_{n-1} + c_{n-1} d_n$$

and the matrix looks like:

$$\left( \begin{array}{ccc} S^{(n-2)} & c_{n-1} R_{n-1}^T & s_{n-1} R_{n-1}^T \\ R_{n-1} & d_{n-1} & s_{n-1}(c_{n-1} d_{n-1} + d_n) \\ 0 & 0 & \hat{d}_1 \end{array} \right).$$

Applying the transformation on the right gives the following equations:

$$\left( \begin{array}{ccc} S^{(n-2)} & R_{n-1}^T & 0 \\ R_{n-1} & \tilde{d}_{n-1} & s_{n-1}\hat{d}_1 \\ 0 & s_{n-1}\hat{d}_1 & c_{n-1}\hat{d}_1 \end{array} \right),$$

with $\tilde{d}_{n-1} = \left(1 + s_{n-1}^2\right) c_{n-1} d_{n-1} + s_{n-1}^2 d_n$. When denoting the new representation from right to left with $\hat{G}$ and $\hat{d}$, we get:

$$\hat{G}_1 = \left( \begin{array}{cc} c_{n-1} & -s_{n-1} \\ s_{n-1} & c_{n-1} \end{array} \right)$$

and $\hat{d}_1$. This procedure can be continued to find all the Givens transformations $\hat{G}_i$ and the vector $\hat{d}$. Note once more that this representation is constructed from right to left and only the new diagonal elements need to be calculated.

## 3. The reduction to unreduced form

With the mathematical details given in the previous section, the reader should be able to derive the numerical implementation of the algorithm to transform the matrix into unreduced form. Therefore, we do not include further details of this implementation.

A remaining question that we want to address in detail is the following: when do we assume an element above the diagonal to be includable in the lower semiseparable structure? We will design a proper numerical criterion. Suppose we have a 2 by $n$ matrix. We will assume that this matrix is of rank 1 if the following relation holds, between the smallest $\sigma_2$ and the largest singular value $\sigma_1$:

$$\sigma_2 \leq \epsilon_M \sigma_1 \tag{67}$$

where $\epsilon_M$ denotes the machine precision.

Suppose we are performing the reduction as described in Section 4 of Chapter 7 and we perform the following Givens transformation $G_{k+1}^T$ on row $k$ and $k+1$ to annihilate row $k$ up to the diagonal: (Only the first $k+1$ elements are shown.)

$$A = G_{k+1}^T \begin{pmatrix} c_k R_k & c_k d_k & e \\ s_k R_k & s_k d_k & d_{k+1} \end{pmatrix} = \begin{pmatrix} R_k & d_k & b \\ 0 & 0 & \delta \end{pmatrix}.$$

We will assume these two rows to be dependent if the criterion (67) is satisfied Calculating the singular values of this last $2 \times (k+1)$ submatrix $A$, via calculating the eigenvalues of $AA^T$ gives: (with $a^2 = d_k^2 + \|R_k\|_2^2$.)

$$\begin{aligned} \lambda_1 &= \frac{a^2+b^2}{2} + \frac{\delta^2}{2} + \frac{1}{2}\sqrt{a^4 + 2a^2b^2 - 2a^2\delta^2 + b^4 + 2b^2\delta^2 + \delta^4} \\ \lambda_2 &= \frac{a^2+b^2}{2} + \frac{\delta^2}{2} - \frac{1}{2}\sqrt{a^4 + 2a^2b^2 - 2a^2\delta^2 + b^4 + 2b^2\delta^2 + \delta^4}. \end{aligned}$$

(The efficient calculation of the norm $\|R_k\|_2^2$ will be addressed in Section 5.) We assume $\delta$ to be small (i.e. $\delta^2 \ll a^2+b^2$), then $\lambda_1$ and $\lambda_2$ can be approximated as:

$$\begin{aligned} \lambda_1 &\approx a^2+b^2 \\ \lambda_2 &\approx \frac{\delta^2 a^2}{a^2+b^2}. \end{aligned}$$

In this way we get

$$\begin{aligned} \sigma_1 &\approx \sqrt{a^2+b^2} \\ \sigma_2 &\approx \frac{|\delta||a|}{\sqrt{a^2+b^2}}. \end{aligned}$$

Therefore we assume that this matrix is of rank 1 if

$$|\delta||a| \leq \epsilon_M \left(a^2+b^2\right).$$

## 4. Effective $O(n)$ implementation of the implicit $QR$-algorithm with shift for symmetric semiseparable matrices

As the implicit $QR$-algorithm with shift also starts with performing a step of $QR$ without shift, we assume that we start with a matrix as given at the end of Section 2. The following part in the algorithm performs in fact the next $n-1$ Givens transformations on the matrix. It starts with one special Givens transformation $G_n$. Because the following sequence of Givens transformations will divide the matrix into two semiseparable parts, we have to store twice a symmetric semiseparable matrix. The decreasing lower right semiseparable matrix will be stored in the Givens-vector representation $G$,$d$. While the growing upper left part will also be stored in the Givens-vector representation $\hat{G}, \hat{d}$.

Suppose we first perform the special Givens transformation $G_n$ (see Section 1.3 Chapter 8 for more details concerning the transformation $G_n$) on the matrix $S_1$, which looks like (this is different from the matrix in the previous section, because the representation is from right to left now):

$$\begin{pmatrix} d_n & s_{n-1}d_{n-1} & s_{n-1}R_{n-1} \\ s_{n-1}d_{n-1} & c_{n-1}d_{n-1} & c_{n-1}R_{n-1} \\ s_{n-1}R_{n-1}^T & c_{n-1}R_{n-1}^T & S_1^{(n-2)} \end{pmatrix}. \tag{68}$$

Applying the first special Givens transformation $G_n^T$ on the left of the matrix (68), we get:

$$\begin{pmatrix} c_n d_n + s_n s_{n-1} d_{n-1} & (c_n s_{n-1} + s_n c_{n-1})\, d_{n-1} & (c_n s_{n-1} + s_n c_{n-1})\, R_{n-1} \\ -s_n d_n + c_n s_{n-1} d_{n-1} & (-s_n s_{n-1} + c_n c_{n-1})\, d_{n-1} & (-s_n s_{n-1} + c_n c_{n-1})\, R_{n-1} \\ s_{n-1} R_{n-1}^T & c_{n-1} R_{n-1}^T & S_1^{(n-2)} \end{pmatrix}.$$

Applying the Givens transformation $G_n$ on the right gives us

$$\begin{pmatrix} \hat{d}_1 & \alpha_1 & f_1 R_{n-1} \\ \alpha_1 & \tilde{d}_{n-1} & f_2 R_{n-1} \\ f_1 R_{n-1}^T & f_2 R_{n-1}^T & S_1^{(n-2)} \end{pmatrix},$$

with

$$\begin{aligned} \hat{d}_1 &= c_n^2 d_n + s_n^2 c_{n-1} d_{n-1} + 2 c_n s_n c_{n-1} d_{n-1}, \\ f_1 &= (c_n s_{n-1} + s_n c_{n-1}), \\ f_2 &= (-s_n s_{n-1} + c_n c_{n-1}), \\ \alpha_1 &= -c_n s_n d_n + \left( \left( c_n^2 - s_n^2 \right) s_{n-1} - s_n c_n c_{n-1} \right) d_{n-1}, \\ \tilde{d}_{n-1} &= s_n^2 d_n + \left( -2 c_n s_n s_{n-1} + c_n^2 c_{n-1} \right) d_{n-1}. \end{aligned}$$

The lower right reduced semiseparable matrix can be constructed by the old representation and the knowledge of $\tilde{d}_{n-1}$ and the factor $f_2$.

The upper left three by three block can now be used to construct the next Givens transformation according to Theorem 71. One can clearly see that this procedure, can be repeated to find the new diagonal element $\hat{d}_2$ and the first subdiagonal element $\hat{d}_1^{(s)}$, and so on.

Also the new representation is built up at the same time, by storing extra information concerning the values of $\alpha_1$ and $\tilde{d}_{n-1}$.

NOTE. The implementation as presented here presumes once more the ideal situation. This implementation, is suitable for semiseparable matrices coming from the similarity reduction of a symmetric matrix to semiseparable form, but for the general class of semiseparable matrices it is not suitable. In this implementation we assumed that only the Givens transformations from Proposition 71 is used. In practice the implementation changes a little bit to make it suitable for general semiseparable matrices. For a full understanding the software can be downloaded and investigated.

## 5. Deflation after a step of the $QR$-algorithm

An important, yet uncovered topic, is the deflation or cutting criterion. When should we divide the semiseparable matrix into smaller blocks, without losing too much information. For semiseparable matrices two things have to be taken into consideration.

The first point of difference with the tridiagonal approach, is the fact that an off-diagonal element in the tridiagonal matrix, has all the information corresponding to the non-diagonal block in which the element appears. This is straightforward, because all the other elements are zero. This is however not the case for semiseparable matrices; in fact they are dense matrices. This means that we should derive a way to calculate the norms of the off-diagonal blocks in a fast way. Moreover comparing the norms of all the off-diagonal blocks towards the cutting criterion should in total cost not more than $O(n)$ operations. Otherwise this would be the slowest step in the algorithm, which is unacceptable.

The second issue is whether the norm of the block is small enough to divide the problem into two subproblems or not. This is a difficult problem and in fact we will test two different cutting criteria, and see what the difference in accuracy is. The two cutting criteria which will be compared in the numerical experiments section are the aggressive and the normal cutting criterion [**91**]. The aggressive criterion allows deflation when the norm of the block is relatively smaller than the square root of the machine precision. The normal criterion allows deflation when the norm is relatively smaller then the machine precision. Denoting the machine precision with $\epsilon_M$, we consider the following two deflation criteria: The aggressive:

$$\|S(i+1:n,1:i)\|_F \leq \sqrt{|d_i d_{i+1}|}\sqrt{\epsilon_M}$$

or the normal deflation criterion:

$$\|S(i+1:n,1:i)\|_F \leq \sqrt{|d_i d_{i+1}|}\epsilon_M,$$

where the $d_i$ denote the diagonal elements of the matrix. When the deflation criterion is satisfied, deflation is allowed and the matrix $S$ is divided into two matrices $S(1:i,1:i)$ and $S(i+1:n,i+1:n)$, thereby neglecting the block $S(i+1:n,1:i)$.

In the remaining part of this section we will derive an order $n$ algorithm to compute the norms of the off-diagonal blocks and to use them in the current cutting criterion. The semiseparable structure should be exploited when calculating these norms. An easy calculation shows, that for a semiseparable matrix $S$ with the Givens-vector representation the following equations are satisfied:

$$\begin{aligned}
\|S(2:n,1:1)\|_F &= \sqrt{(s_1 d_1)^2} \\
\|S(3:n,1:2)\|_F &= \sqrt{(s_2 s_1 d_1)^2 + (s_2 d_2)^2} \\
&= |s_2|\sqrt{(s_1 d_1)^2 + d_2^2} \\
&= |s_2|\sqrt{\|S(2:n,1:1)\|_F^2 + d_2^2}
\end{aligned}$$

This process can be continued and in general we get:

$$\|S(i+1:n,1:i)\|_F = |s_i|\sqrt{\|S(i:n,1:i-1)\|_F^2 + d_i^2}.$$

This formula allows us to derive an $O(n)$ algorithm to compute and use the norms of these blocks in the actual cutting criterion.

## 6. Computing the eigenvectors

For computing the eigenvectors of the corresponding semiseparable matrix, we distinguish two cases. If one desires all the eigenvectors or if one only needs a few.

**6.1. Computing all the eigenvectors.** If one desires all the eigenvectors of the semiseparable matrix $S$, one can store all the orthogonal transformations performed in the implicit $QR$-algorithm. In this way we construct the matrix $Q$, such that:

$$Q^T S Q = \Lambda,$$

where $\Lambda$ is a diagonal matrix containing all the eigenvalues of the matrix $S$. This can be rewritten as

$$SQ = Q\Lambda.$$

Hence, that the columns of the orthogonal matrix $Q$ are the eigenvectors of the matrix $S$.

**6.2. Selected eigenvectors.** Suppose not all the eigenvectors, but just a few of them are desired. Performing or storing all the transformations as described above is too expensive. We can compute these eigenvectors via inverse iteration.

Suppose we have already a good approximation of the eigenvalues. With inverse iteration applied to the semiseparable matrix and these approximations, the eigenvectors can be computed efficiently.

In fact, if one wants to calculate the eigenvector corresponding to the eigenvalue $\lambda_i$, the following system of equations needs to be solved several times:

$$(S - \tilde{\lambda} I)x = b,$$

with $\tilde{\lambda}$ close to the eigenvalue $\lambda_i$. The problem above corresponds to solving a system with a semiseparable plus diagonal matrix. This can be achieved in order $O(n)$ with the algorithm presented in Part I, Chapter 1, Section 7.

We can solve the system mentioned above now in a stable and accurate way, and therefore we are able to calculate the eigenvectors of the semiseparable matrix via inverse iteration. In the numerical tests the following algorithm from [**162**] is used.

While (true)
    Solve the system $(S - \kappa I)y = x$ ;
    $\hat{x} = y/\|y\|_2$;
    $w = x/\|y\|_2$;
    $\rho = \hat{x}^T w$; (Rayleigh quotient)
    $\mu = \kappa + \rho$;
    $\kappa = \mu$;
    $r = w - \rho\hat{x}$;
    $x = \hat{x}$;
    if$(\|r\|_2/\|A\|_F \leq \epsilon)$ leave while;
end

Starting with a random vector $x$ and the shift $\kappa$ equal to an approximation of an eigenvalue coming from the implicit $QR$-algorithm, the eigenvectors can be calculated in a fast and accurate way.

**6.3. The eigenvectors of an arbitrary symmetric matrix $A$.** Suppose our semiseparable matrix $S$ is the result of the orthogonal similarity transformations as described in Part 2, applied to a matrix $A$ and one is interested in the eigenvectors of the original matrix. We compute via one of the techniques described above the eigenvectors of $S$. These eigenvectors need to be transformed to the original eigenvectors.

The most natural way to achieve this goal, is to store the orthogonal transformations performed while reducing the matrix $A$ into the semiseparable matrix $S$. This corresponds to keeping either a sequence of $n-1$ Householder transformations plus $n(n-1)/2$ Givens transformations, or keeping (when using Givens transformations instead of Householder transformations) $n(n-1)$ Givens transformations.

## 7. The implementations of the implicit $QR$-algorithms for the Hessenberg-like matrices and the upper triangular semiseparable matrices

The implementations of the $QR$-algorithms for the Hessenberg-like and the upper triangular semiseparable matrices can be downloaded from `http://www.cs.kuleuven.ac.be/~marc`. The algorithms are implemented in a similar way as the $QR$-algorithm for the symmetric case and we will not go into the details of this implementation.

## 8. Tests on the symmetric eigenvalue solver

In this section several numerical tests are performed to compare the traditional algorithm for finding all the eigenvalues with the new semiseparable approach. The algorithm is based on the $QR$-step as described in Section 2 and Section 4, and implemented in a recursive way: if division in blocks is possible (i.e., that the deflation criterion is satisfied) because of the convergence behavior, then these blocks are dealt with separately.

Before starting the numerical tests some remarks have to be made: first of all the complexity of the reduction of a symmetric matrix into a similar semiseparable one costs $9n^2 + O(n)$ flops more than the reduction of a matrix to tridiagonal form. An implicit $QR$-step applied to a symmetric tridiagonal matrix costs $31n$ flops while it costs $\approx 10n$ flops more for a symmetric semiseparable matrix. However, this increased complexity is compensated when comparing the number of iteration steps the traditional algorithm needs with the number of steps the semiseparable algorithm needs. Figures about these results can be found in the following tests.

**8.1. The block experiment.** This experiment is taken from [**140**, p. 153]. Suppose we have a symmetric matrix $A^{(0)}$ of dimension $n$ and we construct the following matrices $T(m,\delta)$ using $A^{(0)}$, where $m$ denotes the number of blocks, and $\delta$ are the small subdiagonal elements, between the blocks. For example:

$$T(3,\delta) = \left(\begin{array}{c|c|c} A^{(0)} & \blacksquare & \\ \hline \blacksquare & A^{(0)} & \blacksquare \\ \hline & \blacksquare & A^{(0)} \end{array}\right) \qquad \blacksquare = \delta.$$

We get the following results: For $n = 10$ we compare for the semiseparable and the tridiagonal approach the maximum number of iterations for any eigenvalue to converge. This is done for a varying size of $\delta$, $A^{(0)}$ has eigenvalues $1:10$. Both of the algorithms use the same normal deflation criterion.

The difficulty in this experiment are the small off-diagonal elements. The reduction to the tridiagonal form will leave these elements quite small, in this way our resulting tridiagonal matrix is almost a block diagonal. The algorithm we perform now, will not deflate the blocks. In this way we can see the difficulties the implicit $QR$-algorithm encounters, trying to solve this problem. The semiseparable approach will perform better, because the reduction to semiseparable form, will already rearrange these small elements, because of the $QL$ steps performed while reducing these matrices. In this way the overall number of steps before convergence can occur is less than the corresponding number of steps of the tridiagonal approach. For $m = 10$:

| $\delta$ | $10^{-13}$ | $10^{-12}$ | $10^{-11}$ | $10^{-10}$ | $10^{-9}$ | $10^{-8}$ | $10^{-7}$ |
|---|---|---|---|---|---|---|---|
| semiseparable $QR$ | 3 | 3 | 2 | 2 | 2 | 3 | 3 |
| tridiagonal $QR$ | 4 | 4 | 4 | 4 | 4 | 4 | 3 |

For $m = 25$:

| $\delta$ | $10^{-15}$ | $10^{-14}$ | $10^{-13}$ | $10^{-12}$ | $10^{-11}$ |
|---|---|---|---|---|---|
| semiseparable $QR$ | 4 | 4 | 4 | 3 | 3 |
| tridiagonal $QR$ | 4 | 5 | 5 | 4 | 4 |
| $\delta$ | $10^{-10}$ | $10^{-9}$ | $10^{-8}$ | $10^{-7}$ | |
| semiseparable $QR$ | 3 | 2 | 2 | 2 | |
| tridiagonal $QR$ | 3 | 4 | 2 | 3 | |

For $m = 40$

| $\delta$ | $10^{-19}$ | $10^{-18}$ | $10^{-17}$ | $10^{-16}$ | $10^{-15}$ | $10^{-14}$ | $10^{-13}$ |
|---|---|---|---|---|---|---|---|
| semiseparable $QR$ | 4 | 5 | 4 | 4 | 4 | 4 | 4 |
| tridiagonal $QR$ | 4 | 5 | 4 | 5 | 5 | 4 | 4 |
| $\delta$ | $10^{-12}$ | $10^{-11}$ | $10^{-10}$ | $10^{-9}$ | $10^{-8}$ | $10^{-7}$ | |
| semiseparable $QR$ | 3 | 1 | 1 | 1 | 3 | 3 | |
| tridiagonal $QR$ | 4 | 5 | 4 | 4 | 4 | 5 | |

It can be seen clearly that the tridiagonal approach has more difficulties in finding particular eigenvalues. Figure 9.1 gives a comparison in the complete number of $QR$-steps for the last experiment ($m = 40$).

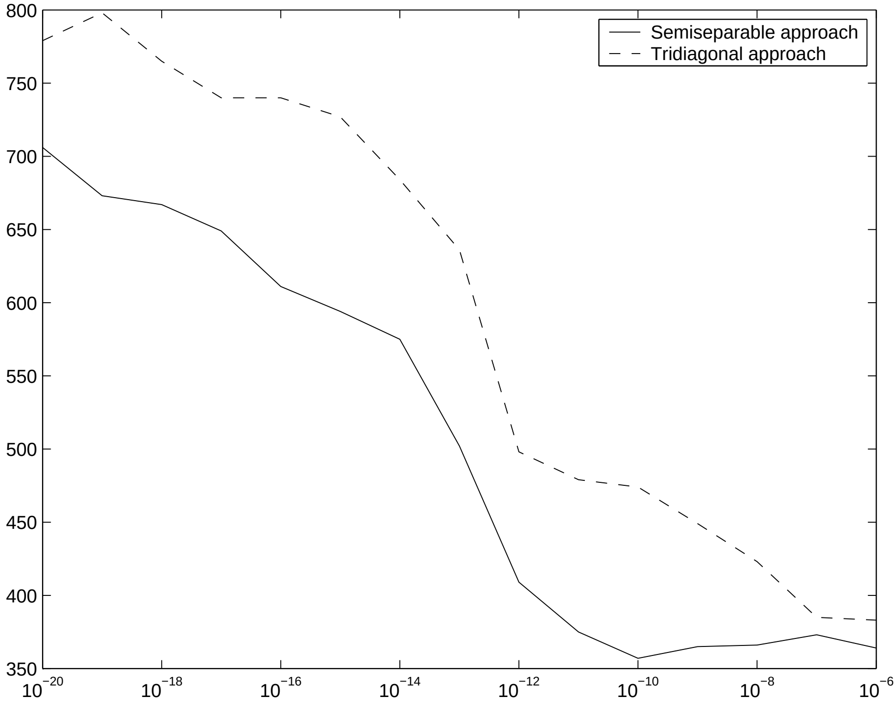


FIGURE 9.1: Total number of steps compared to several values of $\delta$.

The figure shows that the semiseparable approach needs less iterations than the traditional approach. The matrices involved are of size 400. It can be seen that for $\delta$ in the neighborhood of $10^{-10}$ the number of $QR$-steps with the semiseparable approach are even less than 400. This can also be seen in the table for $m = 40$.

**8.2. Stewart's devil's stairs.** In the following example (from [**163**]), we do not apply the $QR$-algorithm, but we only take a look at the diagonal elements of the semiseparable matrix and the tridiagonal one, after the reduction step. We compare these diagonal elements with the real eigenvalues, which are Stewart's devil's stairs. In Figure 9.2 one can see 10 stairs, with gaps of order 50 between the stairs. All the blocks are of size 10.

It can be seen that the devil's stairs are approximated much better by the diagonal elements of the semiseparable than by the diagonal elements of the tridiagonal matrix.

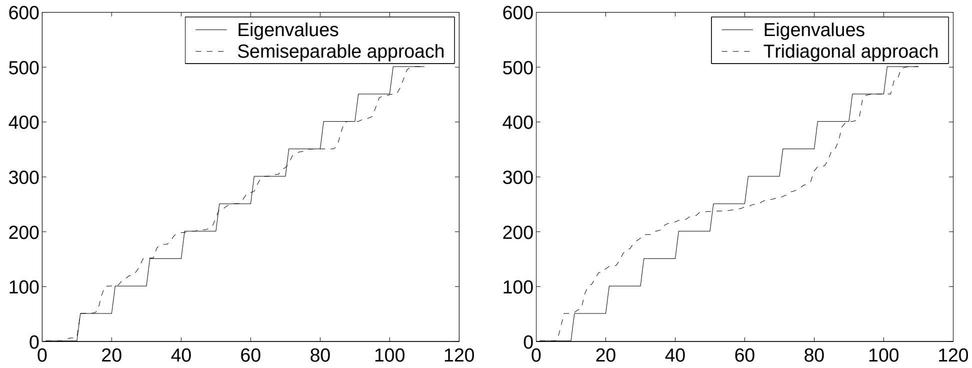


FIGURE 9.2: Stewart's Devil's stairs

**8.3. Problem matrices.** The following difficult matrices can be found in [**47**]. We take some of the eigenvalue problems which the traditional $QR$-algorithm cannot solve in finite precision. We consider the following matrix: $A = DPD$, where $D = \mathrm{diag}(10^{20}, 10^{10}, 1)$,

$$A = \begin{pmatrix} 10^{40} & 10^{29} & 10^{19} \\ 10^{29} & 10^{20} & 10^{9} \\ 10^{19} & 10^{9} & 1 \end{pmatrix} \text{ and } P = \begin{pmatrix} 1 & 0.1 & 0.1 \\ 0.1 & 1 & 0.1 \\ 0.1 & 0.1 & 1 \end{pmatrix}.$$

The eigenvalues of the matrix $A$ are the following:

$$\Lambda = [1.00000 \cdot 10^{40}, 9.90000 \cdot 10^{19}, 9.81818 \cdot 10^{-1}].$$

The eigenvalues computed by the routine $eig(\cdot)$ in MATLAB gives the following results:

$$[-3.85544 \cdot 10^{23}, 9.90002 \cdot 10^{-1}, 1.00000 \cdot 10^{40}].$$

One can see that the eigenvalue solver of MATLAB, could only compute one eigenvalue correctly. The eigenvalues computed by the semiseparable procedure are

$$[1.00000 \cdot 10^{40}, 9.81818 \cdot 10^{-1}, 9.90000 \cdot 10^{19}].$$

All these eigenvalues have at least six correct digits.

The next matrix we investigate is constructed in a similar way as the previous one, suppose we have $D = \mathrm{diag}(10^{20}, 10^{10}, 1)$, $A = DPD$, $\mu = 10^{-6}$,

$$A = \begin{pmatrix} 10^{40} & 9.99 \cdot 10^{29} & 9.99 \cdot 10^{19} \\ 9.99 \cdot 10^{29} & 10^{20} & 9.99 \cdot 10^{9} \\ 9.99 \cdot 10^{19} & 9.99 \cdot 10^{9} & 1 \end{pmatrix} \text{ and } P = \begin{pmatrix} 1 & 1-\mu & 1-\mu \\ 1-\mu & 1 & 1-\mu \\ 1-\mu & 1-\mu & 1 \end{pmatrix}.$$

The eigenvalues of this matrix are $\Lambda = [10^{40}, 2 \cdot 10^{14}, 1.5 \cdot 10^{-6}]$. The eigenvalue solver of MATLAB cannot solve this problem. The implicit $QR$-algorithm for semiseparable matrices gives the following eigenvalues.

$$[1.000000000000000 \cdot 10^{40}, 1.999999000102929 \cdot 10^{14}, 1.499999749837038 \cdot 10^{-06}]$$

these results are correct up to six digits.

These results show immediately, that the algorithm performs in several cases much better than the traditional $QR$ with tridiagonal matrices. In some cases the $QR$-steps do not need to be performed anymore, because the reduction already reveals all the information.

**8.4. Cutting off the last eigenvalue.** In Figure 9.3, we compared the accuracy of the eigenvalues, depending on the deflation criterion. A sequence of matrices of varying size was generated, with equally spaced eigenvalues in the interval $[0, 1]$. The eigenvalues for these matrices were calculated by using the aggressive and the normal deflation criterion. For both these tests the absolute error of the residuals was computed and plotted in the next figure. One can clearly see, that applying the aggressive deflation criterion is almost as good as the normal deflation criterion.

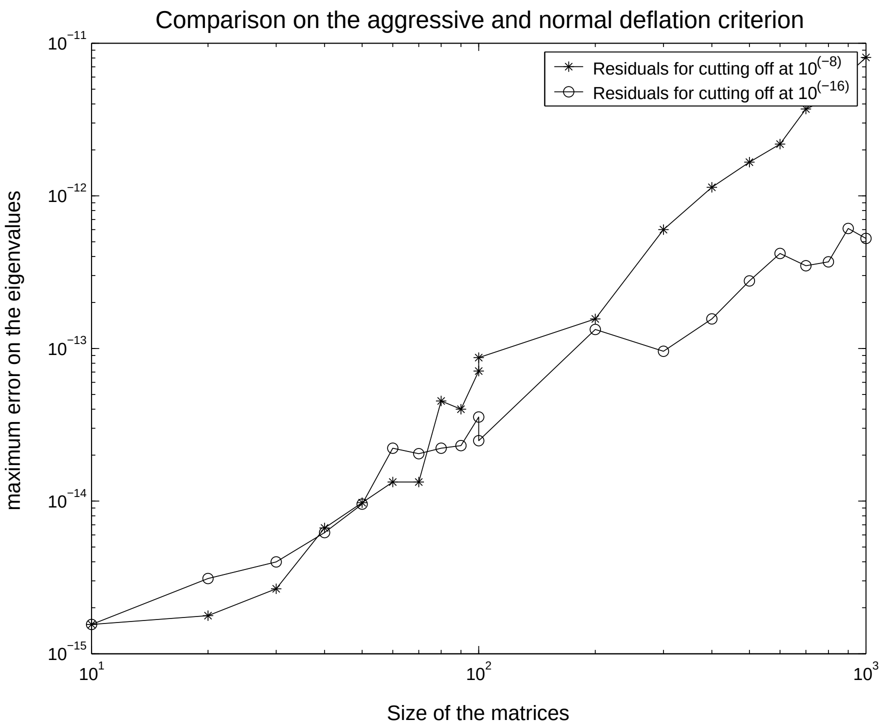


FIGURE 9.3: Comparing different deflation criteria

## 9. Numerical examples connected to the singular value $QR$-algorithm

In this section we perform tests on some arbitrary matrices to compute their singular values. First the matrices are reduced to upper triangular semiseparable form, and then the implicit $QR$-algorithm to compute the singular values is applied on these matrices.

Numerical tests are performed comparing the behavior of the traditional approach via bidiagonal and the proposed semiseparable approach for computing the singular values of several matrices. Special attention is paid to the accuracy of both algorithms for different sets of the singular values and to the number of $QR$-steps needed to compute all the singular values of the matrices. The first figure shows a comparison of the number of implicit $QR$-iterations performed (the line connects the average number of iterations for each experiment). It shows that the semiseparable approach needs less steps in order to find the singular values but do not forget that a semiseparable implicit $QR$-step costs a little more than a corresponding step on the bidiagonal.

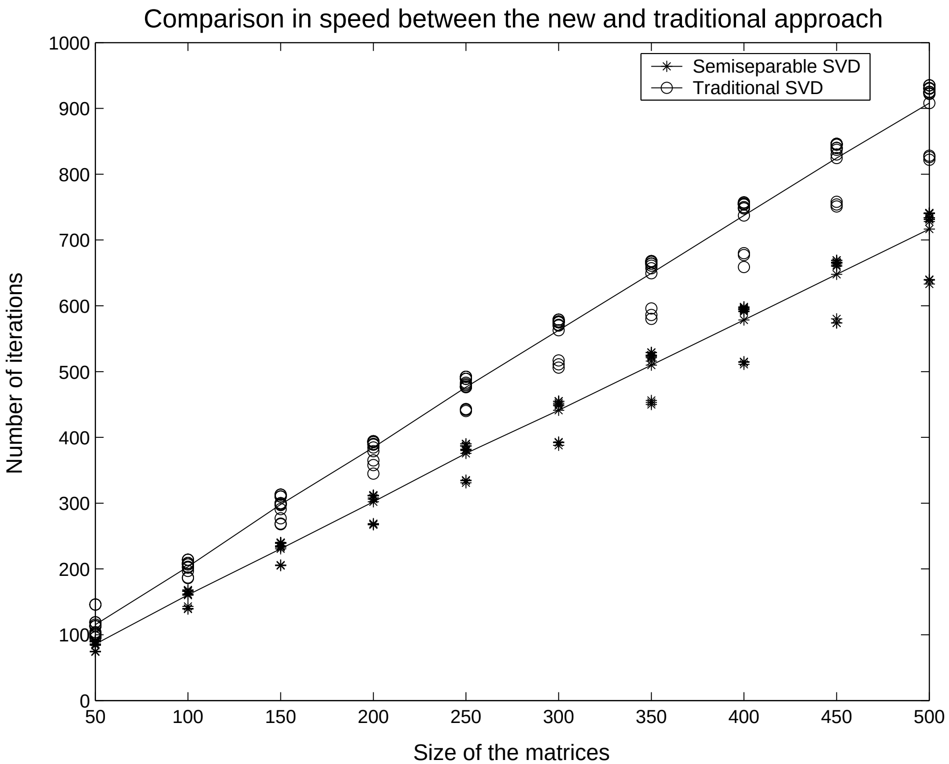


FIGURE 9.4: Number of implicit $QR$-steps

Figure 9.5 and Figure 9.6 show comparisons in accuracy of the two approaches (A line is drawn connecting the average error for different experiments. The error is the maximum relative error between the computed and the exact singular values). In Figure 9.5 the singular values were chosen equally spaced, in Figure 9.6 the singular values ranged from 1 to $n$. Both figures show that the two approaches are equally accurate.

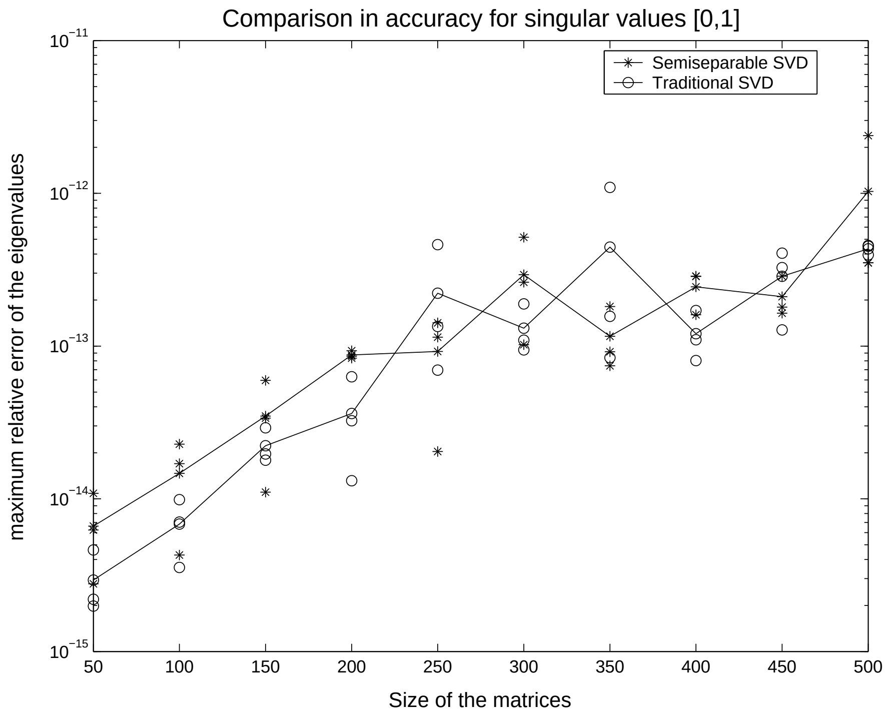


FIGURE 9.5: Comparison in accuracy

## Conclusions

In this chapter we provided the mathematical details behind the implementations of the different implicit $QR$-algorithms, the reduction to unreduced form and the $QR$-factorization of semiseparable plus diagonal matrices. Moreover we also provided a cutting criterion for the division in blocks, and a mathematical investigation deciding wether or not a block is of rank 1. Briefly we spoke about the computation of all or of a few eigenvectors, about the implementation for the Hessenberg-like and about the computation of the singular values. Finally some numerical tests were performed using the implicit $QR$-algorithm for semiseparable matrices. We tested the algorithm on an "almost" block matrix and compared the results with the algorithm based on tridiagonal matrices. We also tested the algorithm on a matrix with the eigenvalues in stair-form and we showed that the diagonal elements approximated already quite well the eigenvalues after the reduction to semiseparable form. We tested again some problem matrices from Chapter 6, Part 2 which the eigenvalue solver based on tridiagonal matrices cannot solve. We received accurate results. Finally some experiments were performed comparing the number of steps and the accuracy of the tridiagonal and the semiseparable approach.

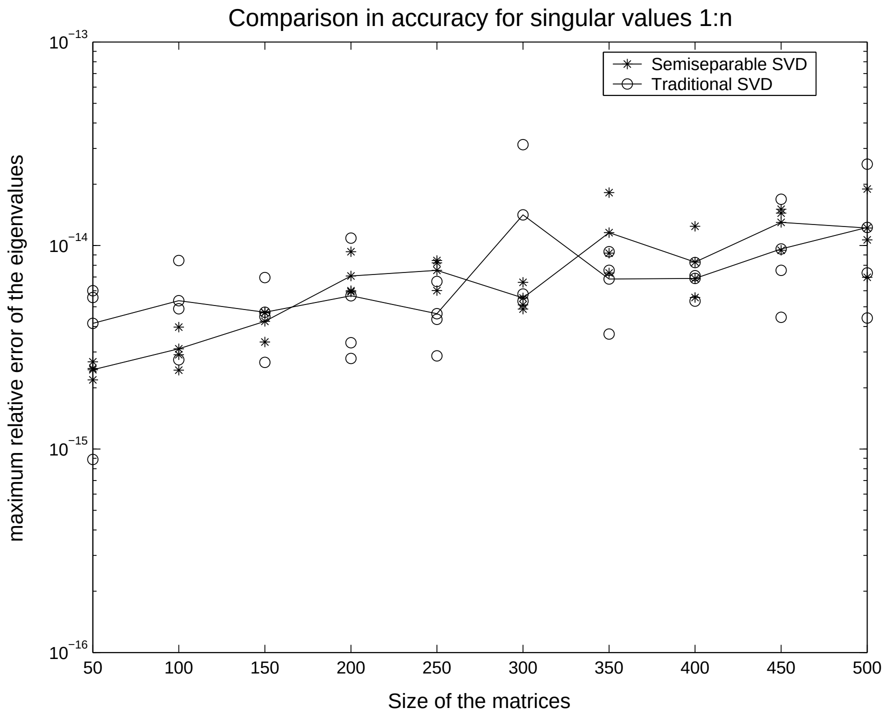


Figure 9.6: Comparison in accuracy

CHAPTER 10

# Software package for semiseparable matrices

This chapter gives an overview of the routines that are included in the software package for semiseparable matrices. The software is written for the programming environment MATLAB. The package can be downloaded from `http://www.cs.kuleuven.ac.be/~marc/software/`.

The tools contained in the software package are not always restricted to the implementations, as described in the thesis. Several of the implementations were adapted, such that they are applicable to a larger class of matrices, e.g. to unsymmetric matrices, lower triangular semiseparable matrices etc.

The tools described here are designed for, or working with: symmetric or unsymmetric **S**emi**S**eparable matrices (**SS**), **U**pper triangular **S**emiseparable matrices (**US**), **L**ower triangular **S**emiseparable matrices (**LS**) and **H**essenberg-**L**ike matrices (**HL**).

The given descriptions are very short. More help can be found when considering these routines in MATLAB and calling the help information of the function, e.g.

```
> help CSS;
```

will give much more detailed information about the routine **CSS**.

## 1. General tools

### 1.1. Retrieving the Givens-vector representation.

- **BRSS:** ***B**uild the **R**epresentation of a **S**emi**S**eparable matrix*
  The function will retrieve the Givens-vector representation from a semiseparable input matrix. The input matrix can either be symmetric or unsymmetric. For an unsymmetric the lower triangular part and the strictly upper triangular part will be represented by two sequences of Givens transformations and two vectors.
- **BRHL:** ***B**uild the **R**epresentation of a **H**essenberg-**L**ike matrix.*
  The function will retrieve the Givens-vector representation of a Hessenberg-like matrix. The output consists of a Givens-vector representation, representing the lower triangular part, and the strictly upper triangular part of the Hessenberg-like matrix.
- **BRUS & BRLS:** Similar as above for **US** and **LS** matrices.

### 1.2. Constructing the matrices from the representation.

- **BSS:** ***B**uild a full **S**emi**S**eparable matrix.*
  Given the Givens-vector representation of either a symmetric or an unsymmetric matrix, the function will build a full semiseparable matrix from it.

- **BUS & BLS & BHL:** The same as above for **US**, **LS** and **HL** matrices.

### 1.3. Various tools.

- **SR:** ***S****wap the* ***R****epresentation.*
  A semiseparable matrix can be represented with the Givens-vector representation from two directions. This function changes this direction.
- **MULSS:** ***MUL****tiply a vector with a* ***S****emi**S**eparable matrix.*
  Performs an order $O(n)$ matrix vector multiplication. The matrix can either be symmetric or unsymmetric, in the unsymmetric case two sequences of Givens-vector representations have to be given as input.
- **MULUS & MULLS & MULHL:** The same as above for **US**, **LS** and **HL** matrices.
- **DETSS:** *Calculate the* ***DET****erminant of a* ***S****emi**S**eparable matrix.*
  Given a symmetric or unsymmetric semiseparable matrix, represented with the Givens-vector representation, the function will calculate the determinant in $O(n)$.
- **DETUS & DETLS & DETHL:** The same as above for **US**, **LS** and **HL** matrices.

## 2. Reduction Algorithms

- **CSS:** **C***onstruction of a* **S***ymmetric* **S***emiseparable matrix.*
  The function reduces via orthogonal similarity transformations the symmetric input matrix into semiseparable form. The presented output is in the Givens-vector representation form.
- **CUS & CLS & CHL:** Similar as above for **US**, **LS** and **HL** matrices.

## 3. $QR$-tools

### 3.1. $QR$-solver.

- **SOLVSSD:** ***SOL****ve a system with a* ***S****emi**S**eparable plus* ***D****iagonal matrix.*
  This routine solves a system of equations for which the coefficient matrix is of diagonal plus semiseparable form in $O(n)$ operations. The semiseparable part is represented using the Givens-vector representation.
- **SOLVHLD:*****SOL****ve a system with a* ***H****essenberg-**L**ike plus* ***D****iagonal matrix.*

### 3.2. Reduction to unreduced form.

- **REDSS** ***RED****uction of a* ***S****emi**S**eparable matrix to unreduced form.*
  This function transforms via orthogonal similarity transformations the semiseparable matrix to unreduced form.
- **REDUS & REDHL:** The same as above for **US** and **HL** matrices.

### 3.3. An implicit $QR$-step without shift.

- **QRSS:** *A* ***QR****-step on a* ***S****emi**S**eparable matrix.*
  This routine performs a step of $QR$ without shift on a semiseparable matrix represented with the Givens-vector representation.
- **QRUS & QRHL:** The same as above for **US** and **HL** matrices.

### 3.4. An implicit $QR$-step with shift.

- **QRSSS:** *A **QR**-step with **S**hift on a **S**emi**S**eparable matrix.*
  This routine performs a step of $QR$ with shift on a semiseparable matrix represented with the Givens-vector representation.
- **QRSUS & QRSHL:** The same as above for **US** and **HL** matrices.

### 3.5. The eigen(singular)value decomposition.

- **EIGSS:** *The **EIG**envalue decomposition of a **S**emi**S**eparable matrix.*
  This routine calculates the eigenvalue decomposition of a symmetric semiseparable matrix in the Givens-vector representation.
- **EIGHL:** The same as above for **HL** matrices.
- **SVDUS:** *The **SVD** of **U**pper triangular **S**emiseparable matrices.*
  This routine calculates the singular value decomposition of an upper triangular semiseparable matrix represented with the Givens-vector representation.

## 4. Eigenvalue and singular value tools

- **SSEIG:** ***S**emi**S**eparable **EIG**envalue decomposition.*
  This routine, calculates the eigenvalue decomposition of a symmetric matrix by first transforming the original matrix to a similar semiseparable one.
- **HLEIG:** ***H**essenberg-**L**ike **EIG**envalue decomposition.*
  This routine, calculates the eigenvalue decomposition of a matrix by first transforming the original matrix to a similar Hessenberg-like one.
- **USSVD:** ***U**pper triangular **S**emiseparable **SVD**.*
  This routine calculates the singular value decomposition of matrices, by first transforming them to an upper triangular semiseparable matrix.

# Conclusions and future research

In this thesis, we presented results for the computation of eigenvalues and singular values via semiseparable matrices. We will briefly highlight the main results, indicate which of these results were newly discovered by us, and point out which topics can be considered for future research.

In the first chapter of the manuscript, we thoroughly investigated the nullity theorem, and provided several examples connected to semiseparable and related matrices. Moreover, based on the article of Barrett and Feinsilver [**9**], we provided an alternative proof of the nullity theorem. Based on this investigation we chose as definition for semiseparable matrices, the rank definition, because it is more general than the generator definition. The extensions of the nullity theorem are also our work and offers new research possibilities. The presented theorems are quite strong with respect to decompositions of structured rank matrices, but they are restricted to the class of invertible matrices. It is worth analyzing the behavior of the nullity theorem for singular matrices w.r.t. the pseudo inverse. This can give rise to very general theorems, predicting the structure of factorizations for structured rank matrices. Furthermore we compared in this chapter the generator representable semiseparable matrices with the class of semiseparable matrices. We discovered common misunderstandings, and numerical instabilities when using this class of matrices. Furthermore, we proved strong theorems, with respect to the embedding of generator representable matrices in the class of semiseparable matrices, and we proved that the closure of the class of generator representable matrices for pointwise convergence gave us the class of semiseparable matrices.

Because of our alternative definition, we defined a representation of semiseparable matrices in a solid way in the second chapter. We investigated two types of common representations, the generator representation and the representation with the diagonal and the subdiagonal. As the generator and diagonal subdiagonal representations do not cover the complete class of semiseparable matrices, we developed a new representation, based on a sequence of Givens transformations and a vector. Briefly we mentioned an alternative representation by Dario Bini and Luca Gemignani [**19**], which however uses more parameters than our representation. As the Givens-vector representation is new, we had to develop different algorithms for it. We proposed an algorithm to retrieve the representation, an algorithm to compute the determinant of a matrix represented with the Givens-vector representation and an algorithm to multiply a semiseparable matrix given by this Givens-vector representation and a vector.

In the third chapter an overview of some applications and algorithms already known for semiseparable matrices was presented. Also an overview of the existing literature on the topic of the thesis was made.

The results that we developed in Part II and Part III are new, as far as we know. We will briefly highlight some of the most interesting results which also open new research possibilities.

In the fourth chapter a reduction to reduce matrices using an orthogonal similarity transformation to semiseparable or Hessenberg-like form was presented as well as an orthogonal reduction to upper triangular semiseparable form. These reductions are interesting and worth further investigation. A possible extension, which is a direct consequence of the research, is the reduction to semiseparable plus diagonal form. The reduction to semiseparable plus diagonal form is interesting because it can speed up convergence, as the convergence behavior of this reduction could be interpreted as a step of $QR$ with shift (see subspace iteration convergence behavior).

The fifth chapter analyzed the convergence behavior of the different reduction algorithms to semiseparable form. A first type of convergence behavior is the Lanczos convergence behavior. Because the Lanczos convergence behavior first reveals the extreme eigenvalues, in case of positive eigenvalues, also the smallest one is approximated sometimes rather well. We are currently investigating an updating technique (based on [**20**]) which will give us the eigenvalues of the already reduced part in semiseparable form. Hence also the smallest singular value can be calculated in a reasonable time.

The second convergence behavior is a type of nested subspace iteration performed on the already reduced part of the matrix. Deeper investigations in this type of convergence behavior revealed that the iteration corresponds to a step of $QR$ without shift. An important topic raises here: is it possible to include a shift in this subspace iteration step; this could increase convergence speed, and reduce the number of iterations of the forthcoming $QR$-steps, needed to compute the whole spectrum. The inclusion of this shift is related to the reduction of a matrix to semiseparable plus diagonal form.

In Chapter 6 different numerical experiments were performed to compare the traditional reduction to tridiagonal form and the reduction to semiseparable form. A specific type of experiment compared the relative error of both approaches for some difficult matrices, more precisely some graded matrices. It appears that the semiseparable approach gives more accurate results. It is interesting to investigate more in detail the convergence behavior and the accuracy of this reduction, as the computation of accurate eigenvalues is an important topic (see, e.g., [**46, 47, 50, 140, 162, 189**]).

The last part is dedicated to the development of implicit $QR$-algorithms for semiseparable matrices. We developed the notion of unreducedness for Hessenberg-like matrices and we proved how different matrices can be transformed into this form. Very important in this part is the formulation of an implicit $Q$-theorem in its very general form, based on the unreducedness and the unreduced number of the corresponding Hessenberg-like matrices. We proved that the structure of semiseparable/Hessenberg-like, and semiseparable/Hessenberg-like plus diagonal matrices is maintained under one step of the $QR$-algorithm. Currently we are investigating in more detail which general structures (w.r.t. rank) are preserved under one step of the $QR$-algorithm. More precisely in the thesis it was shown, that for singular matrices, sometimes the $QR$-factorization has to satisfy certain properties needed to maintain the structure. Also this can be further investigated for more general structures.

In Chapter 8 different implicit $QR$-algorithms were derived. We did not consider the Francis $QR$-step for the Hessenberg-like case nor multi-shifts (see [**51, 106**]). This means that we can speed up convergence for the symmetric case, and more general that complex conjugate eigenvalues in the Hessenberg-like case can be approximated by the eigenvalues of a $2 \times 2$ real submatrix.

Also the convergence behavior of the semiseparable matrices inside the $QR$-algorithm needs more detailed convergence analysis. Based on the convergence rates for tridiagonal matrices and the papers [**43, 44, 136**] describing the decay rates of the elements of the inverse of a tridiagonal matrix, it could be possible to give explicit formulas for the accuracy of the eigenvalues in terms of the size of the elements of the semiseparable matrix. Also formulas could be deduced predicting the decay rates of the elements in the semiseparable $QR$-algorithm.

The last chapter of Part III showed us some numerical examples, under which Stewart's Devil's stairs. Also in the sixth chapter an example was included showing the approximation of the diagonal elements of the semiseparable matrix, with respect to the distribution of the eigenvalues. It appears that the reduction of a matrix to semiseparable form also possesses some kind of rank-revealing properties. This should also be looked at in more detail.

The final chapter contained some information about the software package corresponding to this thesis, implemented in MATLAB. One of our next tasks is the implementation of the software in a higher level language such as C, C++, Java or FORTRAN.

In the remainder of this concluding chapter, we present some new directions for research, which follow from the thesis. These new directions cover investigations of important classes of matrices arising in different applications.

Currently a lot of research is done on the class of $\mathcal{H}$-matrices and recursively semiseparable matrices. This class of matrices can be used for solving different types of problems, e.g., state-space models (see [**48**] and the references therein), integral equations, elliptic pde's,... [**21, 93, 103**], Lyapunov and Riccati equations [**94**].

When solving eigenvalue problems via the $QR$-algorithm for recursively semiseparable matrices and $\mathcal{H}$-matrices, it is not clear if the currently available representations lead to numerically stable implementations. We are convinced that the Givens-vector representation can be generalized for these classes of matrices.

A second important research topic, is the $QR$-algorithm itself. No current algorithm exists for performing $QR$ steps on recursively semiseparable matrices taking into account the structure of the matrix, although $QR$-decompositions are available [**36**]. Important questions related to this subject are the following ones: are the factors $Q$ and $R$ structured in any way; even more, are they also recursively semiseparable matrices. Provided with the techniques of the phd-thesis, namely generalizations of the nullity theorem [**9, 73, 74, 164**], it seems possible to predict specific structures of the $Q$ and $R$ factors of the decomposition of the recursively semiseparable or $\mathcal{H}$ matrix. Moreover, using directly the nullity theorem, strong theorems concerning the structure of these matrices could be derived in correspondence with papers [**9, 133, 150, 151, 154, 164, 191**]. (More references connected to the inverses of band and block band matrices can be found in Chapter 3.)

Finally, based on the theorems and tools provided in this thesis, it seems that a $QR$-algorithm for a new class of structured rank matrices can be developed. The new class of structured rank matrices can be defined in the following way: when neglecting some kind of block band in the matrix, the remaining, smaller upper triangular and lower triangular parts are of low rank. When decreasing the size of the block band, the lower and upper triangular parts increase in size, thereby also increasing the rank of the blocks. This new type of matrices, can be seen as the extension of semiseparable plus diagonal matrices [**35, 38, 130, 178**], where the matrix in question belongs to different classes of semiseparable plus block diagonal matrices, thereby varying the semiseparability rank and the block diagonal matrix. The difference with this class and the class of $\mathcal{H}$-matrices is the fact that for the latter class, all the different blocks come from different low rank matrices while for this type of matrix the lower triangular part corresponding to a certain block diagonal matrix comes from 1 semiseparable matrix of semiseparability rank $k$. Different types of applications, involving $\mathcal{H}$-matrices or recursively semiseparable matrices, will probably fit in this context, thereby reducing the number of parameters and the complexity with respect to the approach described above.

# Bibliography


1. E. L. Allgower, *Criteria for positive definiteness of some band matrices*, Numerische Mathematik **16** (1970), 157–162. {**69**}
2. ———, *Exact inverse of certain band matrices*, Numerische Mathematik **21** (1973), 279–284. {**67, 69**}
3. E. Asplund, *Inverses of matrices $a_{ij}$ which satisfy $a_{ij} = 0$ for $j > i + p$*, Mathematica Scandinavica **7** (1959), 57–60. {**11, 66**}
4. S. O. Asplund, *Finite boundary value problems solved by Green's matrix*, Mathematica Scandinavica **7** (1959), 49–56. {**66**}
5. R. Baltzer, *Theorie und anwendung der determinanten*, fifth ed., S. Hirzel, 1881. {**52**}
6. J. Baranger and M. Duc-Jacquet, *Matrices tridiagonales symmetriques et matrices factorisables*, Revue Française d'Informatique et de Recherche Operationelle **5** (1971), no. R-3, 61–66. {**68**}
7. W. W. Barrett, *A theorem on inverse of tridiagonal matrices*, Linear Algebra and Its Applications **27** (1979), 211–217. {**9, 11, 70**}
8. W. W. Barrett and P. J. Feinsilver, *Gaussian families and a theorem on patterned matrices*, Journal of Applied Probability **15** (1978), 514–522. {**58, 69**}
9. ———, *Inverses of banded matrices*, Linear Algebra and Its Applications **41** (1981), 111–130. {**5, 9, 11, 70, 195, 198**}
10. L. Berg, *Auflösung von gleichungssystemen mit einer bandmatrix*, Zeitschrift für Angewandte Mathematik und Mechanik **57** (1977), 373–380. {**69**}
11. W. J. Berger and E. Saibel, *On the inversion of continuant matrices*, Journal of the Franklin Institute **6** (1953), 249–253. {**66**}
12. R. Bevilacqua, *Structural and computational properties of band matrices*, appeared in R. Bevilacqua et al., Complexity of structured computational problems, Applied mathematics monographs, CNR, giardini editori e stanpatori, Pisa (1991), 131–188. {**66**}
13. R. Bevilacqua and M. Capovani, *Proprietà delle matrici tridiagonali ad elementi ed a blocchi*, Monografie dell'I.E.I., 1972. {**66**}
14. ———, *Proprietà delle matrici a banda ad elementi ed a blocchi*, Bolletino Unione Matematica Italiana **5** (1976), no. 13-B, 844–861. {**69**}
15. R. Bevilacqua, B. Codenotti, and F. Romani, *Parallel solution of block tridiagonal linear systems*, Linear Algebra and Its Applications **104** (1988), 39–57. {**71**}
16. R. Bevilacqua and G. M. Del Corso, *Existence, uniqueness and algorithms for matrix unitary reduction to semiseparable form*, Tech. Report TR-03-12, University of Pisa, July 2003. {**74, 139, 144**}
17. R. Bevilacqua, G. Lotti, and F. Romani, *Storage compression of inverses of band matrices*, Computers & Mathematics with Applications **20** (1990), no. 8, 1–11. {**72**}
18. R. Bevilacqua, F. Romani, and G. Lotti, *Parallel inversion of band matrices*, Computers and Artificial Intelligence **9** (1990), no. 5, 493–501. {**72**}
19. D. Bini, L. Gemignani, and V. Pan, *QR-like algorithms for generalized semiseparable matrices*, Tech. Report 1470, Department of Mathematics, University of Pisa, 2004. {**27, 37, 39, 74, 130, 148, 195**}
20. C. F. Borges and W. B. Gragg, *A parallel divide and conquer algorithm for the generalized real symmetric definite tridiagonal eigenproblem*, Numerical Linear Algebra (Kent, OH, 1992) (1993), 11–29, de Gruyter, Berlin. {**196**}

21. S. Börm, L. Grasedyck, and W. Hackbusch, *Introduction to hierarchical matrices with applications*, Tech. Report 18, Max-Planck-Institut, Leipzig, 2002. {**74, 197**}
22. M. Braun, S. A. Sofianos, D. G. Papageorgiou, and I. E. Lagaris, *An efficient Chebyshev Lanczos method for obtaining eigensolutions of the Schrödinger equation on a grid*, Journal of Computational Physics **126** (1996), no. 2, 315–327. {**101**}
23. R. A. Brualdi and J. J. Q. Massey, *More on structure-ranks of matrices*, Linear Algebra and Its Applications **183** (1993), 193–199. {**72**}
24. B. Bukhberger and G. A. Emel'yanenko, *Methods of inverting tridiagonal matrices*, Computational Mathematics and Mathematical Physics (translated from Zhurnal Vychislitel'noĭ Matematiki i Matematicheskoĭ Fiziki) **13** (1973), 10–20. {**68**}
25. A. Bultheel, A. Cuyt, W. Van Assche, M. Van Barel, and B. Verdonk, *Generalizations of orthogonal polynomials*, TW report 375, Dept. Computer Science, 3001 Heverlee (Leuven), December 2003. {**64**}
26. A. Bultheel, M. Van Barel, and P. Van Gucht, *Orthogonal basis functions in discrete least-squares rational approximation*, Journal of Computational and Applied Mathematics **164-165** (2004), 175–194. {**64**}
27. R. L. Burden and J. D. Faires, *Numerical analysis*, fourth ed., PWS-Kent publishing company, 1988. {**63**}
28. W. S. Burnside and A. W. Panton, *An introduction to determinants*, ch. from The theory of equations, Hodges, Figgis, & Co. and Longmans, Green & Co., 1899. {**52**}
29. W. L. Cao and W. J. Stewart, *A note on inverses of Hessenberg-like matrices*, Linear Algebra and Its Applications **76** (1986), 233–240. {**71**}
30. M. Capovani, *Sulla determinazione della inversa delle matrici tridiagonali e tridiagonali a blocchi*, Calcolo **7** (1970), 295–303. {**68**}
31. ———, *Su alcune proprietà delle matrici tridiagonali e pentadiagonali*, Calcolo **8** (1971), 149–159. {**68**}
32. K. M. Chandan, *The inverse of a certain matrix with an application*, Annals of Mathematical Statistics **38** (1967), 1289–1292. {**58, 67**}
33. S. Chandrasekaran, P. Dewilde, M. Gu, T. Pals, X. Sun, A.-J. van der Veen, and D. White, *Fast stable solvers for sequentially semi–separable linear systems of equations and least squares problems*, http://cobalt.et.tudelft.nl/∼allejan/publications.html, preprint, 2001. {**64, 74**}
34. S. Chandrasekaran, P. Dewilde, M. Gu, T. Pals, and A.-J. van der Veen, *Fast stable solver for sequentially semi-separable linear systems of equations*, Lecture Notes in Computer Science **2552** (2002), 545–554. {**38, 64, 74**}
35. S. Chandrasekaran and M. Gu, *Fast and stable eigendecomposition of symmetric banded plus semi-separable matrices*, Linear Algebra and Its Applications **313** (2000), 107–114. {**19, 73, 84, 198**}
36. ———, *A fast and stable solver for recursively semi-separable systems of linear equations*, Contemporary Mathematics **281** (2001), 39–53. {**64, 74, 197**}
37. ———, *Fast and stable algorithms for banded plus semiseparable systems of linear equations*, SIAM Journal on Matrix Analysis and its Applications **25** (2003), no. 2, 373–384. {**73**}
38. ———, *A divide and conquer algorithm for the eigendecomposition of symmetric block-diagonal plus semi-separable matrices*, Numerische Mathematik **96** (2004), no. 4, 723–731. {**73, 84, 198**}
39. D. Colton and R. Kress, *Inverse acoustic and electromagnetic scattering theory*, second ed., Springer-Verlag, 1998. {**75**}
40. C. Corduneanu, *Integral equations and applications*, Press Syndicate of the University of Cambridge, 1991. {**63**}
41. C. E. Cullis, *Matrices and determinoids*, Readership lectures, vol. 1, Cambridge, University Press, 1913. {**52**}
42. J. K. Cullum and R. A. Willoughby, *Lanczos algorithms for large symmetric eigenvalue computations*, Birkhäuser, Boston, 1985. {**92**}
43. S. Demko, *Inverses of band matrices and local convergence of spline projections*, SIAM Journal on Numerical Analysis **14** (1977), no. 4, 616–619. {**69, 197**}

44. S. Demko, W. F. Moss, and P. W. Smith, *Decay rates for inverses of band matrices*, Mathematics of Computation **43** (1984), 491–499. {**70, 197**}
45. J. W. Demmel, *Applied numerical linear algebra*, SIAM, 1997. {**2, 92, 93**}
46. J. W. Demmel and W. Kahan, *Accurate singular values of bidiagonal matrices*, SIAM Journal on Scientific Computing **11** (1990), no. 5, 873–912. {**196**}
47. J. W. Demmel and K. Veselić, *Jacobi's method is more accurate than QR*, SIAM Journal on Matrix Analysis and its Applications **13** (1992), no. 4, 1204–1245. {**117, 185, 196**}
48. P. Dewilde and A.-J. van der Veen, *Time-varying systems and computations*, Kluwer academic publishers, Boston, June 1998. {**38, 64, 74, 75, 197**}
49. ———, *Inner-outer factorization and the inversion of locally finite systems of equations*, Linear Algebra and Its Applications **313** (2000), 53–100. {**38, 74**}
50. I. S. Dhillon, *A new $O(n^2)$ algorithm for the symmetric tridiagonal eigenvalue/eigenvector problem*, Ph.D. thesis, University of California, 1989. {**196**}
51. A. A. Dubrulle and G. H. Golub, *A multishift QR iteration without computation of the shifts*, Numerical Algorithms **7** (1994), no. 2-4, 173–181. {**197**}
52. Y. Eidelman, *Fast recursive algorithm for a class of structured matrices*, Applied Mathematical Letters **13** (2000), 57–62. {**74**}
53. Y. Eidelman and I. Gohberg, *Inversion formulas and linear complexity algorithm for diagonal plus semiseparable matrices*, Computers & Mathematics with Applications **33** (1996), no. 4, 69–79. {**72**}
54. ———, *Fast inversion algorithms for diagonal plus semiseparable matrices*, Integral Equations and Operator Theory **27** (1997), no. 2, 165–183. {**72**}
55. ———, *A look ahead block schur algorithm for diagonal plus semiseparable matrices*, Computers & Mathematics with Applications **35** (1997), no. 10, 25–34. {**72**}
56. ———, *On a new class of structured matrices*, Integral Equations and Operator Theory **34** (1999), 293–324. {**38, 64, 74**}
57. ———, *Algorithms for inversion of diagonal plus semiseparable operator matrices*, Integral Equations and Operator Theory **44** (2002), no. 2, 172–211. {**74**}
58. ———, *A modification of the Dewilde van der Veen method for inversion of finite structured matrices*, Linear Algebra and Its Applications **343-344** (2002), 419–450. {**38, 74**}
59. ———, *Fast inversion algorithms for a class of structured operator matrices*, Linear Algebra and Its Applications **371** (2003), 153–190. {**74**}
60. V. Eijkhout and B. Polman, *Decay rates of inverses of banded $M$-matrices that are near to Toeplitz matrices*, Linear Algebra and Its Applications **109** (1988), 247–277. {**71**}
61. L. Elsner, *Some observations on inverses of band matrices and low rank perturbations of triangular matrices*, Acta Technica Academiae Scientiarum Hungaricae **108** (1997-1999), no. 1-2, 41–48. {**11, 72**}
62. L. Elsner and C. Giersch, *Metabolic control analysis: Separable matrices and interdependence of control coefficients*, Journal of theoretical Biology **193** (1998), 649–661. {**75**}
63. D. K. Faddeev, *Properties of the inverse of a Hessenberg matrix*, Numerical methods and questions in the organizations of calculations 5, Zapiski Nauchnykh Seminarov Leningradskogo Otdeleniya Matematicheskogo Instituta imeni V. A. Steklova **111** (1981), 177–179, (in Russian). {**66**}
64. S. M. Fallat, M. Fiedler, and T. L. Markham, *Generalized oscillatory matrices*, Linear Algebra and Its Applications **359** (2003), 79–90. {**14, 49**}
65. D. Fasino, *Rational Krylov matrices and QR-steps on hermitian diagonal-plus-semiseparable matrices*, submitted for publication, 2004. {**74**}
66. D. Fasino and L. Gemignani, *Structural and computational properties of possibly singular semiseparable matrices*, Linear Algebra and its Applications **340** (2001), 183–198. {**19, 73**}
67. ———, *A Lanczos type algorithm for the QR-factorization of regular Cauchy matrices*, Numerical Linear Algebra with Applications **9** (2002), 305–319. {**64**}
68. ———, *Direct and inverse eigenvalue problems, for diagonal-plus-semiseparable matrices*, Numerical Algorithms **34** (2003), 313–324. {**64, 73**}

69. D. Fasino, N. Mastronardi, and M. Van Barel, *Fast and stable algorithms for reducing diagonal plus semi separable matrices to tridiagonal and bidiagonal form*, Contemporary Mathematics **323** (2003), 105–118. {**74, 84**}
70. P. Favati, G. Lotti, F. Romani, and P. Rózsa, *Generalized band matrices and their inverses*, Calcolo **28** (1991), 45–92. {**72**}
71. M. Fiedler, *Structure ranks of matrices*, Linear Algebra and Its Applications **179** (1993), 119–127. {**5, 11–13, 72**}
72. ———, *Basic matrices*, Linear Algebra and Its Applications **373** (2003), 143–151. {**5, 11, 14, 27, 32, 34, 35, 74**}
73. M. Fiedler and T. L. Markham, *Completing a matrix when certain entries of its inverse are specified*, Linear Algebra and Its Applications **74** (1986), 225–237. {**5, 8–11, 71, 198**}
74. ———, *Rank-preserving diagonal completions of a matrix*, Linear Algebra and Its Applications **85** (1987), 49–56. {**11, 71, 198**}
75. ———, *Generalized totally nonnegative matrices*, Linear Algebra and Its Applications **345** (2002), 9–28. {**14, 49**}
76. M. Fiedler and V. Pták, *On matrices with non-positive off-diagonal elements and positive principal minors*, Czechoslovak Mathematical Journal **12** (1962), no. 87, 382–400. {**49**}
77. M. Fiedler and Z. Vavřín, *Generalized Hessenberg matrices*, Linear Algebra and Its Applications **380** (2004), 95–105. {**5, 11, 14, 27, 32, 34, 35, 74**}
78. R. A. Frazer, W. J. Duncan, and A. R. Collar, *Elementary matrices*, Cambridge, university press, 1965. {**58**}
79. F. R. Gantmacher, *On non symmetric Kellogg kernels*, Doklady Acadademii Nauk (USSR) **1** (1936), no. 10, 3–5. {**50**}
80. F. R. Gantmacher and M. G. Krein, *On a special class of determinants related to Kellog integral kernels*, Matematicheskiĭ Sbornik **40** (1933), 501–508. {**49**}
81. ———, *Sur les matrices oscillatoires*, Comptes Rendus Mathématique Académie des Sciences Paris **201** (1935), 577–579. {**49**}
82. ———, *Sur les matrices oscillatoires et complètement non négatives*, Compositio Mathematica **4** (1937), 445–476. {**49, 66**}
83. ———, *Oscillation matrices and kernels and small vibrations of mechanical systems*, AMS Chelsea publishing, 2002. {**xvii, 10, 48, 50–52, 56, 57, 66, 68**}
84. W. Gautschi, *Numerical analysis, an introduction*, Birkhauser, 1997. {**63**}
85. L. Gemignani and D. Fasino, *Fast and stable solution of banded-plus-semiseparable linear systems*, Calcolo **39** (2002), no. 4, 201–217. {**73**}
86. W. L. Gifford, *A short course in the theory of determinants*, Macmillan and Co., 1893. {**52**}
87. ———, *Determinants*, fourth ed., Mathematical monographs, vol. 3, John Wiley and Sons, 1906. {**52**}
88. I. Gohberg and M. A. Kaashoek, *Time varying linear systems with boundary conditions and integral operators, I. the transfer operator and its properties*, Integral Equations and Operator Theory **7** (1984), 325–391. {**75**}
89. I. Gohberg, T. Kailath, and I. Koltracht, *Linear complexity algorithms for semiseparable matrices*, Integral Equations and Operator Theory **8** (1985), no. 6, 780–804. {**71**}
90. G. H. Golub and W. Kahan, *Calculating the singular values and pseudo-inverse of a matrix.*, SIAM Journal on Numerical Analysis **2** (1965), 205–224. {**85**}
91. G. H. Golub and C. F. Van Loan, *Matrix computations*, third ed., The Johns Hopkins university Press, 1996. {**2, 36, 89, 92, 98, 130, 135, 137, 149, 156, 159, 160, 180**}
92. R. A. Gonzales, J. Eisert, I. Koltracht, M. Neumann, and G. Rawitscher, *Integral equation method for the continuous spectrum radial Schrödinger equation*, Journal of Computational Physics **134** (1997), 134–149. {**74**}
93. L. Grasedyck and W. Hackbusch, *Construction and arithmetics of $\mathcal{H}$-matrices*, Computing **70** (2003), no. 4, 295–334. {**197**}
94. L. Grasedyck, W. Hackbusch, and B. N. Khoromskij, *Solution of large scale algebraic matrix Riccati equations by use of hierarchical matrices*, Computing **70** (2003), no. 2, 121–165. {**197**}
95. F. A. Graybill, *Matrices with applications in statistics*, Wadsworth international group, Belmont, California, 1983. {**58, 61, 62, 75**}

96. F. A. Graybill, C. D. Meyer, and R. J. Painter, *Note on the computation of the generalized inverse of a matrix*, SIAM Review **8** (1966), no. 4, 522–524. {**58**}
97. B. G. Greenberg and A. E. Sarhan, *Matrix inversion, its interest and application in analysis of data*, Journal of the American Statistical Association **54** (1959), 755–766. {**58, 66, 67, 69**}
98. L. Greengard and V. Rokhlin, *On the numerical solution of two-point boundary value problems*, Communications on Pure and Applied Mathematics **44** (1991), 419–452. {**74**}
99. T. N. E. Greville, *On a problem concerning band matrices with Toeplitz inverses*, Proceedings of the eigth Manitoba Converence on Numerical Mathematics and Computing (1978), 275–283. {**69**}
100. G. J. Groenewald, M. A. Petersen, and A. C. M. Ran, *LU- versus UL-factorization of integral operators with semiseparable kernel*, http://www.cs.vu.nl/~ran/publications.html, accepted for publication in Integral Equations and Operator Theory, 2003. {**63, 74**}
101. ———, *Characterization of integral operators with semiseparable kernel and symmetries*, http://www.cs.vu.nl/~ran/publications.html, 2004. {**74**}
102. W. H. Gustafson, *A note on matrix inversion*, Linear Algebra and Its Applications **57** (1984), 71–73. {**8, 71**}
103. W. Hackbusch and B. N. Khoromskij, *A sparse $\mathcal{H}$-matrix arithmetic: general complexity estimates*, Journal of Computational and Applied Mathematics **125** (2000), no. 1-2, 479–501. {**74, 197**}
104. ———, *A sparse $\mathcal{H}$-matrix arithmetic, part II: Application to multi-dimensional problems*, Computing **64** (2000), no. 1, 21–47. {**74**}
105. S. B. Haley, *Solution of band matrix equations by projection-recurrence*, Linear Algebra and Its Applications **32** (1980), 33–48. {**70**}
106. N. J. Higham, *SIAG/LA prize winners speed up the QR-algorithm*, SIAM News **36** (2003), no. 9, 3–3. {**197**}
107. W. D. Hoskins and G. E. McMaster, *On the inverses of a class of Toeplitz matrices of band width five*, Linear and Multilinear Algebra **4** (1976/77), no. 2, 103–106. {**69**}
108. ———, *Properties of the inverses of a set of band matrices*, Linear and Multilinear Algebra **5** (1977/78), no. 3, 183–196. {**69**}
109. ———, *On the infinity norm of the inverse of a class of Toeplitz matrices of band width five*, Linear and Multilinear Algebra **6** (1978/79), no. 2, 153–156. {**69**}
110. W. D. Hoskins and P. J. Ponzo, *Some properties of a class of band matrices*, Mathematics of Computation **26** (1972), 393–400. {**68**}
111. W. D. Hoskins and M. C. Thurgur, *Determinants and norms for the inverses of a set of band matrices*, Utilitas Mathematica **3** (1973), 33–47. {**68**}
112. K. Hüper, J. Götze, and S. Paul, *Subspace separation by discretizations of double bracket flows*, Algorithms, Architectures and Applications (M. Moonen and B. De Moor, eds.), Elsevier Science B.V., 1995, pp. 251–258. {**101**}
113. Y. Ikebe, *On inverses of Hessenberg matrices*, Linear Algebra and Its Applications **24** (1979), 93–97. {**70, 71**}
114. I. T. Joliffe, *Principal component analysis*, Springer verlag, 1986. {**101**}
115. S.-Y. Kang, I. Koltracht, and G. Rawitscher, *High accuracy method for integral equations with discontinuous kernels*, http://arxiv.org/abs/math.NA/9909006, Submitted for publication. {**63, 74**}
116. S. Karlin, *Total positivity*, vol. 1, Stanford University Press, Stanford, California, 1968. {**66**}
117. A. Kavčić and M. F. Moura, *Matrices with banded inverses: inversion algorithms and factorization of gauss-markov processes*, IEEE Transactions on Information Theory **46** (2000), no. 4, 1495–1509. {**74**}
118. O. D. Kellogg, *The oscillation of functions of an orthogonal set*, American Journal of Mathematics **38** (1916), 1–5. {**50**}
119. ———, *Orthogonal functions sets arising from integral equations*, American Journal of Mathematics **40** (1918), 145–154. {**50**}
120. D. Kershaw, *Inequalities on the elements of the inverse of a certain tridiagonal matrix*, Mathematics of computation **24** (1970), 155–158. {**67**}

121. I. Koltracht, *Linear complexity algorithm for semiseparable matrices*, Integral Equations and Operator Theory **29** (1997), no. 3, 313–319. {**72**}
122. Kounadis, *Inversion de matrices tridiagonales symétriques*, RAIRO Recherche Opérationelle **7** (1973), 91–100. {**68**}
123. E. G. Kounias, *An inversion technique for certain patterned matrices*, Journal of Mathematical Analysis and its Applications **21** (1968), 695–698. {**58, 67, 68**}
124. M. G. Krein, *On the nodes of harmonic oscillations of mechanical systems of a special form*, Matematicheskiĭ Sbornik **41** (1934), 339–348, (Russian). {**49**}
125. A. B. J. Kuijlaars, *Which eigenvalues are found by the Lanczos method*, SIAM Journal on Matrix Analysis and its Applications **22** (2000), no. 1, 306–321. {**115**}
126. J-Y. Lee and L. Greengard, *A fast adaptive numerical method for stiff two-point boundary value problems*, SIAM Journal on Scientific Computing **18** (1997), no. 2, 403–429. {**74**}
127. J. W. Lewis, *Inversion of tridiagonal matrices*, Numerische Mathematik **38** (1982), 333–345. {**39, 66, 70**}
128. N. Mastronardi, S. Chandrasekaran, and S. Van Huffel, *Fast and stable two-way algorithm for diagonal plus semi-separable systems of linear equations*, Numerical Linear Algebra with Applications **8** (2001), no. 1, 7–12. {**73**}
129. ______, *Fast and stable algorithms for reducing diagonal plus semiseparable matrices to tridiagonal and bidiagonal form*, BIT **41** (2003), no. 1, 149–157. {**19, 73, 74**}
130. N. Mastronardi, E. Van Camp, and M. Van Barel, *Divide and conquer type algorithms for computing the eigendecomposition of diagonal plus semiseparable matrices.*, Tech. Report 7 (5/2003), Istituto per le Applicazioni del Calcolo “M. Picone”, Consiglio Nazionale delle Ricerche, Rome, Italy, 2003. {**74, 84, 198**}
131. R. Mattheij and M. Smooke, *Estimates for the inverse of tridiagonal matrices arising in boundary-value problems*, Linear Algebra and Its Applications **73** (1986), 33–57. {**71**}
132. J. J. McDonald, R. Nabben, M. Neumann, H. Schneider, and M. J. Tsatsomeros, *Inverse tridiagonal $Z$-matrices*, Linear and Multilinear Algebra **45** (1998), no. 1, 75–97. {**72**}
133. G. Meurant, *A review of the inverse of symmetric tridiagonal and block tridiagonal matrices*, SIAM Journal on Matrix Analysis and its Applications **13** (1992), 707–728. {**72, 198**}
134. A. P. Mullhaupt and K. S. Riedel, *Banded matrix representation of triangular input normal pairs*, IEEE Transactions on Automatic Control **46** (2001), no. 12, 2018–2022. {**75**}
135. ______, *Low grade matrices and matrix fraction representations*, Numerical Linear Algebra with Applications **342** (2002), 187–201. {**73, 75**}
136. R. Nabben, *Decay rates of the inverse of nonsymmetric tridiagonal and band matrices*, SIAM Journal on Matrix Analysis and its Applications **20** (1999), no. 3, 820–837. {**73, 197**}
137. ______, *Two sided bounds on the inverse of diagonally dominant tridiagonal matrices*, Linear Algebra and Its Applications **287** (1999), 289–305. {**72**}
138. E. Neuman, *The inversion of certain band matrices*, Roczniki Polskiego Towarzystwa Matematycznego **3** (1977), no. 9, 15–24. {**69**}
139. T. Oohashi, *Some representation for inverses of band matrices*, TRU Mathematics **14** (1978), no. 2, 39–47. {**69**}
140. B. N. Parlett, *The symmetric eigenvalue problem*, Classics in Applied Mathematics, vol. 20, SIAM, Philadelphia, 1998. {**2, 92, 182, 196**}
141. E. Pascal, *Die determinanten*, Teubner, B.G., 1900. {**52**}
142. R. M. Pringle and A. A. Rayner, *Generalized inverse matrices with applications to statistics*, Griffins's statistical monographs and courses, vol. 28, Griffin, 1971. {**58**}
143. L. Rehnqvist, *Inversion of certain symmetric band matrices*, BIT **12** (1972), 90–98. {**68**}
144. M. Reiss, *Beiträge zur theorie der determinanten*, B.G. Teubner, Leipzig, 1867. {**52**}
145. S. A. H. Rizvi, *Inverses of quasi-tridiagonal matrices*, Linear Algebra and Its Applications **56** (1984), 177–184. {**71**}
146. C. A. Rohde, *Generalized inverses of partitioned matrices*, Journal of the Society for Industrial & Applied Mathematics **13** (1965), no. 4, 1033–1035. {**58**}
147. F. Romani, *On the additive structure of the inverses of banded matrices*, Linear Algebra and Its Applications **80** (1986), 131–140. {**71**}

148. S. N. Roy, B. G. Greenberg, and A. E. Sarhan, *Evaluation of determinants, characteristic equations and their roots for a class of patterned matrices*, Journal of the Royal Statistical Society. Series B. Methodological **22** (1960), 348–359. {**58, 67**}
149. S. N. Roy and A. E. Sarhan, *On inverting a class of patterned matrices*, Biometrika **43** (1956), 227–231. {**66, 67**}
150. P. Rózsa, *Band matrices and semi-separable matrices*, Colloquia Mathematica Societatis Janos Bolyai **50** (1986), 229–237. {**71, 198**}
151. ______, *On the inverse of band matrices*, Integral Equations and Operator Theory **10** (1987), 82–95. {**71, 198**}
152. P. Rózsa, R. Bevilacqua, P. Favati, and F. Romani, *On the inverse of block tridiagonal matrices with applications to the inverses of band matrices and block band matrices*, Operator Theory: Advances and Applications **40** (1989), 447–469. {**71**}
153. P. Rózsa, R. Bevilacqua, F. Romani, and P. Favati, *On band matrices and their inverses*, Linear Algebra and Its Applications **150** (1991), 287–295. {**72**}
154. P. Rózsa, F. Romani, and R. Bevilacqua, *On generalized band matrices and their inverses*, Proceedings of the Lanczos international centary conference (Philadelphia PA) (J. D. Brown, ed.), SIAM Press, 1994, pp. 109–121. {**72, 198**}
155. Y. Saad, *Numerical methods for large eigenvalue problems*, Manchester University Press, Manchester, UK, 1992. {**92**}
156. ______, *Iterative methods for sparse linear systems*, SIAM, January 2000, Second edition with corrections. {**61**}
157. A. E. Sarhan, E. Roberts, and B. G. Greenberg, *Modified square root method of matrix inversion*, Tech. Report 3, U.S. Army office of ordnance research project, February 195. {**58**}
158. S. Schechter, *Quasi tridiagonal matrices and type insensitive difference equations*, Quarterly of Applied Mathematics **18** (1960-1961), 285–295. {**67**}
159. R. F. Scott, *Theory of determinants and their applications*, second ed., Cambridge University Press, 1904. {**52**}
160. V. N. Singh, *The inverse of a certain block matrix*, Bulletin of the Australian Mathematical Society **20** (1979), 161–163. {**70**}
161. H. P. Jr. Starr, *On the numerical solution of one-dimensional integral and differential equations*, Ph.D. thesis, Yale University, 1992, Research Report YALEU/DCS/RR-888. {**63, 74, 75**}
162. G. W. Stewart, *Matrix algorithms, vol II eigensystems*, SIAM, 1999. {**2, 181, 196**}
163. ______, *The QLP approximation to the singular value decomposition*, SIAM Journal on Scientific and Statistical Computation **20** (1999), no. 4, 1336–1348. {**184**}
164. G. Strang and T. Nguyen, *The interplay of ranks of submatrices*, Submitted to SIAM Review, http://www-math.mit.edu/ gs/papers/papers.html, February 2004. {**10, 11, 74, 198**}
165. D. Szynal and J. Szynal, *À propos de l'inversion des matrices généralisées de Jacobi*, Aplikace Matematiky **17** (1972), 28–32. {**66, 68**}
166. T. Ting, *A method of solving a system of linear equations whose coefficients form a tridiagonal matrix*, Quarterly of Applied Mathematics **22** (1964), no. 2, 105–106. {**67**}
167. T. Torii, *Inversion of tridiagonal matrices and the stability of tridiagonal systems of linear equations*, Information processing in Japan (Joho Shori) **6** (1966), 41–46, (In Japanese). {**67**}
168. ______, *Inversion of tridiagonal matrices and the stability of tridiagonal systems of linear equations*, Technology Reports of the Osaka University **16** (1966), 403–414. {**67, 69**}
169. L. N. Trefethen and D. Bau, *Numerical linear algebra*, SIAM, 1997. {**2, 92**}
170. W. F. Trench, *Inversion of Toeplitz band matrices*, Mathematics of Computation **28** (1974), no. 128, 1089–1095. {**68**}
171. E. E. Tyrtyshnikov, *Mosaic ranks for weakly semiseparable matrices*, Large-Scale Scientific Computations of Engineering and Environmental Problems II (M. Griebel, S. Margenov, and P. Yalamov, eds.), Notes on numerical fluid mechanics, vol. 73, Vieweg, 2000, pp. 36–41. {**66, 73**}
172. V. R. R. Uppuluri and J. A. Carpenter, *the inverse of a matrix occurring in first-order moving-average models*, Sankhyā The Indian Journal of Statistics Series A **31** (1969), 79–82. {**67**}

173. ———, *An inversion method for band matrices*, Journal of Mathematical Analysis and its Applications **31** (1970), 554–558. {**67**}
174. F. Valvi, *Explicit presentation of the inverses of some types of matrices*, Journal of the Institute of Mathematics and its Applications **19** (1977), no. 1, 107–117. {**69**}
175. M. Van Barel, D. Fasino, L. Gemignani, and N. Mastronardi, *Orthogonal rational functions and diagonal plus semiseparable matrices*, Advanced signal processing algorithms, architectures, and implementations XII (F. T. Luk, ed.), Proceedings of SPIE, vol. 4791, 2002, pp. 167–170. {**64, 65, 73, 75**}
176. ———, *Orthogonal rational functions and structured matrices*, Report TW 350, Deptartment of Computer Science, Katholieke Universiteit Leuven, Celestijnenlaan 200A, B-300 Leuven (Heverlee), November 2002. {**64**}
177. M. Van Barel, E. Van Camp, and N. Mastronardi, *Orthogonal similarity transformation into semiseparable matrices of semiseparability rank k*, Report TW 380, Department of Computer Science, Katholieke Universiteit Leuven, Celestijnenlaan 200A, B-3001 Leuven (Heverlee), January 2004. {**74**}
178. E. Van Camp, N. Mastronardi, and M. Van Barel, *Two fast algorithms for solving diagonal-plus-semiseparable linear systems*, Journal of Computational and Applied Mathematics **164-165** (2004), 731–747. {**18, 19, 73, 125–127, 171, 198**}
179. R. Vandebril, M. Van Barel, and N. Mastronardi, *A QR-method for computing the singular values via semiseparable matrices*, Report TW 366, Department of Computer Science, Katholieke Universiteit Leuven, Celestijnenlaan 200A, B-3001 Leuven (Heverlee), August 2003. {**106**}
180. N. Veraverbeke, *Kanstheorie en statistiek*, Course Notes (in dutch), Limburgs Universitair Centrum, 1997. {**58**}
181. M. E. Wall, P. A. Dyck, and T. S. Brettin, *SVDMAN - singular value decomposition analysis of microarray data.*, Bioinformatics **17** (2001), no. 6, 566–568. {**101**}
182. M. E. Wall, A. Rechtsteiner, and L. M. Rocha, *Singular value decomposition and principal component analysis*, A practical approach to microarray data analysis (2003), 1–20, D.P. Berrar, W. Dubitzky, M. Granzow, eds, kluwer. {**101**}
183. D. S. Watkins, *Understanding the QR algorithm*, SIAM Review **24** (1982), no. 4, 427–440. {**98**}
184. ———, *Some perspectives on the eigenvalue problem*, SIAM Review **35** (1993), no. 3, 430–471. {**98**}
185. ———, *QR like algorithms an overview of convergence theory and practice*, Lectures in Applied Mathematics **32** (1996), 879–893. {**98**}
186. D. S. Watkins and L. Elsner, *Chasing algorithms for the eigenvalue problem*, SIAM Journal on Matrix Analysis and its Applications **12** (1991), no. 2, 374–384. {**151**}
187. ———, *Convergence of algorithms of decomposition type for the eigenvalue problem*, Linear Algebra and Its Applications **143** (1991), 19–47. {**98, 100–102**}
188. ———, *Theory of decomposition and bulge-chasing algorithms for the generalized eigenvalue problem*, SIAM Journal on Matrix Analysis and its Applications **15** (1994), no. 3, 943–967. {**151**}
189. J. H. Wilkinson, *The algebraic eigenvalue problem*, Oxford University Press, 1965. {**2, 196**}
190. ———, *Global convergence of tridiagonal QR-algorithm with origin shifts*, Linear Algebra and Its Applications **1** (1968), 409–420. {**149**}
191. T. Yamamoto and Y. Ikebe, *Inversion of band matrices*, Linear Algebra and Its Applications **24** (1979), 105–111. {**70, 71, 198**}
192. M. K. S. Yeung and W. L. Ruzzo, *Principal component analysis for clustering gene expression data*, Bioinformatics **17** (2001), no. 9, 763–774. {**101**}
193. M. K. S. Yeung, J. Tegner, and J. J. Collins (eds.), *Reverse engineering gene networks using singular value decomposition and robust regression*, Proceedings National Academy of Sciences, vol. 99, 2002. {**101**}
194. U. Yoshimasa, *Characterization of 2-type diagonal matrices with an application to order statistics*, Journal of the Hokkaido College of Art and Literature **6** (1955), 66–75. {**66, 67, 70**}

195. T. Zhang, G. H. Golub, and K. H. Law, *Subspace iterative methods for eigenvalue problems*, Linear Algebra and Its Applications **294** (1999), no. 1-3, 239–258. {**98**}

# Author/Editor Index

# Index

# Curriculum Vitae

**Education**:

- Ph.D. Student, Doctoral program in engineering (Applied Mathematics), K.U.Leuven, started in October 2000,
  *Advisor*: Prof. Dr. ir. Marc Van Barel
- "Licentiaat Wiskunde, zuivere wiskunde" (Master in pure Mathematics), K.U.Leuven, June 2000.
  Thesis: "$L^p$ ruimten geassocieerd aan von Neumann-algebras", ($L^p$ spaces associated to von Neumann-algebras) at the Department of Mathematics,
  *Advisor*: Prof. Dr. Johan Quaegebeur.
- "Kandidaat Wiskunde", L.U.C. Diepenbeek, July 1988.

**Research Publications**:

In international journals:

- *A fast method for solving the two-dimensional Helmholtz equation with Robbins boundary conditions* , Hendrickx J., Vandebril R. and Van Barel M., Contemporary Mathematics 323:187-204,2003.

Papers under revision:

- *The Lanczos-Ritz values appearing in an orthogonal similarity reduction of a matrix into semiseparable form*, Van Barel M., Vandebril R. and Mastronardi N., pag. 24, under revision, (internal report: TW360), 2003.
- *An implicit QR-algorithm for semiseparable matrices to compute the eigendecomposition of symmetric matrices*, Vandebril R., Van Barel M. and Mastronardi M., pag. 29, under revision, (internal report: TW367), 2003.
- *A note on the representation and definition of semiseparable matrices*, Vandebril R., Van Barel M. and Mastronardi N., pag. 25, under revision, (internal report: TW368), 2003. *A QR-method for computing the singular values via semiseparable matrices*, Vandebril R., Van Barel M. and Mastronardi N., pag. 28, submitted for publication, (internal report: TW366), 2003.

In preparation (some results can already be found in the thesis):

- *A generalization of the nullity theorem towards decomposition theorems for structured rank matrices and singular matrices*, Vandebril R., Van Barel M. and Delvaux S., in preparation.
- *About the invariance of the semiseparable structure under the QR-algorithm*, Delvaux S., Van Barel M. and Vandebril R., in preparation.

- *Computation of the smallest singular value in an orthogonal similarity reduction of a matrix into semiseparable form*, Vandebril R., Van Barel M., Mastronardi N., Schuermans M. and Van Huffel S., in preparation.
- *Rank revealing factorizations using semiseparable matrices*, Mastronardi N., Van Barel M. and Vandebril R., in preparation.
- *Sufficient and necessary conditions for similarity reductions to obtain the Lanczos-convergence behavior*, Van Barel M., Vandebril R. and Mastronardi N., in preparation.

Internal reports, not published:

- *A new algorithm for solving multivariate polynomial problems by means of interpolation*, Vandebril R., Van Barel M., Ruatta O. and Mourrain, B., internal report: TW343, Dept. Computer Science, K.U.Leuven,2002.
- *A fast solver for a bivariate polynomial vector homogeneous interpolation problem*, Vandebril R. and Van Barel M., internal report: TW342, Dept. Computer Science, K.U.Leuven, 2002.
- *An orthogonal similarity reduction of a matrix into semiseparable form*, Vandebril R., Van Barel M. and Mastronardi, N., internal report: TW355, Dept. Computer Science, K.U.Leuven, 2003.

**Talks** at international meetings and conferences.
(The talks were given by the person whose name is underlined.)

- *A new fast method for solving the Helmholtz equation*, Vandebril R., Van Barel M. and Hendrickx J., SIAM Annual Meeting, July 2001, San Diego, California, United States.
- *A fast solver for a bivariate homogeneous interpolation problem*, Vandebril R. and Van Barel M., ICCAM conference, July 2002, Leuven, Belgium.
- *A new approach with semiseparable matrices to find the eigenvalues of symmetric matrices via the QR-algorithm*, Vandebril R., Van Barel M. and Mastronardi N., GAMM meeting, September 2002, Bielefeld, Germany.
- *On computing the eigendecomposition of dense matrices*, Mastronardi N., Vandebril R., Van Barel M., University of Linkoping, February 2003, Linkoping, Sweden.
- *On computing the eigendecomposition, rank revealing and the singular value decomposition of dense matrices*, Mastronardi N., Vandebril R., Van Barel M. and Elden, L., Workshop: Due giornate di Algebra Lineare Numerica, March 2003, Pisa, Italy.
- *Solving symmetric eigenvalue problems using semiseparable matrices instead of trididagonal ones*, Vandebril R., Van Barel M. and Mastronardi N., Cadzand Workshop, June 2003, Cadzand, Netherlands.
- *Semiseparable matrices and eigenproblems* Van Barel M., Vandebril R., Van Camp E. and Mastronardi N., Russian Academy of Sciences: Institute of Numerical Mathematics, Steklov Institute of Mathematics, Workshop on nonlinear approximations in numerical analysis, June 2003, Moscow, Russia.

- *A note on the representation of semiseparable matrices*, Vandebril R., Van Barel M. and Mastronardi N., Workshop on numerical linear algebra, September 2003, Porto Giardino, Monopoli, Italy.
- *A similarity reduction of a symmetric matrix to a semiseparable one and the QR-algorithm to compute the eigenvalues of this semiseparable matrix* Van Barel M., Vandebril R. and Mastronardi N., Numerical Linear Algebra and Its Applications, XXI International School and Workshop, September 2003, Porto Giardino, Monopoli, Bari, Italy.

**Posters**:

- *A new algorithm for computing the SVD via semiseparable matrices*, Vandebril R., Van Barel M. and Mastronardi N., IAP study day, November 2002, Louvain La Neuve, Belgium.
- *Semiseparable matrices and the SVD*, Vandebril R., Van Barel M. and Mastronardi N., Ph.D.-days in Leuven, December 2002, Leuven, Belgium.

**Research stays:**

- A visit to the research institute INRIA, 2004 Route des lucioles B.P. 93, Sophia Antipolis, France. Cooperation with Galaad, the research group of Bernard Mourrain. From October 15, 2001 until November 14, 2001.

**E-mail**: raf.vandebril@cs.kuleuven.ac.be

**Homepage**: http://www.cs.kuleuven.ac.be/~raf/