This dissertation concerns the development of efficient numerical methods to estimate quantities of the form $w^T f(A)v$ where $v, w$ are given vectors or block vectors, $A$ is a symmetric or nonsymmetric matrix and $f$ a smooth function. The research is motivated by the connection between the Lanczos process and Gauss-type quadrature rules. In addition, this dissertation discusses efficient algorithms for the analysis of large-scale complex networks.

First we consider the estimates of upper and lower bounds for matrix functions of the form $w^T f(A)v$ with $A \in \mathbb{R}^{n \times n}$ a large matrix, and $v, w \in \mathbb{R}^n$. When the matrix $A$ is large, the dominating computational effort for evaluating these estimates is the evaluation of matrix-vector products with $A$ and possibly also with $A^T$. The calculation of anti-Gauss rules requires one more matrix-vector product evaluation with $A$ and maybe also with $A^T$ than the computation of the corresponding Gauss rule. We describe a simplification of anti-Gauss quadrature rules that requires the evaluation of the same number of matrix-vector products as the corresponding Gauss rule. This simplification makes the computational effort for evaluating the simplified anti-Gauss rule negligible when the corresponding Gauss rule already has been computed.

Next we consider methods based on quadrature rules computed via block Lanczos algorithms. We derive generalized anti-Gauss quadrature rules for the symmetric and nonsymmetric block Lanczos algorithms, allowing one to compute approximate bounds for elements...
of $W^T f(A)V$, where $W, V \in \mathbb{R}^{n \times k}$, $1 < k \ll n$, for symmetric or nonsymmetric matrices $A$. These methods are applied to the computation of functions of an adjacency matrix. Moreover, novel quantities are introduced which characterize the nodes of a complex network, and which can be computed cheaply via our block methods.

The remaining numerical methods presented rely on multiple orthogonal polynomials, which generalize standard orthogonal polynomials by requiring orthogonality with respect to several inner products or bilinear forms. Such methods are competitive when approximating $W^T f(A)V$ with a large matrix $A$ and the two block vectors $W$ and $V$ with different numbers of columns. Examples show that it may be possible to determine an approximation of $W^T f(A)V$ of the same quality with fewer matrix-vector product evaluations by computing an approximation by Gauss-type quadrature rules associated with multiple orthogonal polynomials than when using a technique based on standard orthogonal polynomials.
GAUSS-TYPE QUADRATURE RULES, WITH APPLICATIONS
IN LINEAR ALGEBRA

A dissertation submitted to
Kent State University in partial
fulfillment of the requirements for the
degree of Doctor of Philosophy

by
Hessah F. Alqahtani

May, 2018
Dissertation written by
Hessah F. Alqahtani
B.S., King Abdulaziz University, Jeddah, Saudi Arabia 2003
M.S., University of Sciences, Pulau Pinang, Malaysia, 2010
M.A., Kent State University, 2014
Ph.D., Kent State University, 2018

Approved by

Dr. Lothar Reichel, Chair, Doctoral Dissertation Committee
Dr. Jun Li, Members, Doctoral Dissertation Committee
Dr. Jing Li
Dr. Arden Ruttan
Dr. Austin Melton

Accepted by

Dr. Andrew Tonge, Chair, Department of Mathematical Sciences
Dr. James Blank, Dean, College of Arts and Sciences
# TABLE OF CONTENTS

ACKNOWLEDGEMENTS .................................................. vii

1 Introduction ......................................................... 1

2 Background, Notation and Definitions ................................. 4
   2.1 Definition of Orthogonal Polynomials ............................. 4
      2.1.1 Matrix Orthogonal Polynomial ............................... 5
   2.2 Quadrature Rules ................................................. 7
   2.3 The Lanczos Algorithm ........................................... 10

3 Simplified Anti-Gauss Quadrature Rules .............................. 14
   3.1 Simplified Anti-Gauss Rules for Symmetric Matrix Functionals . 19
   3.2 Simplified Anti-Gauss Rules for Nonsymmetric Matrix Functionals . 26
   3.3 Simplified Block Anti-Gauss Rules for Functions of a Symmetric Matrix . 33
   3.4 Simplified Block Anti-Gauss Rules for Functions of a Nonsymmetric Matrix . 38
   3.5 Computed Examples ............................................... 41

4 Generalized Block Anti-Gauss Quadrature Rules ..................... 52
   4.1 Scalar-valued Generalized Anti-Gauss Rules ..................... 54
      4.1.1 Scalar-valued Functionals with a Symmetric Matrix ........ 54
      4.1.2 Real-valued Functionals with a Nonsymmetric Matrix .......... 61
   4.2 Generalized Block Anti-Gauss Rules ............................. 71
      4.2.1 Generalized Block Anti-Gauss Rules for Functions of a Symmetric Matrix . 71
4.2.1.1 Generalized Block Anti-Gauss Rules, \( \ell = 1 \) .................................. 72
4.2.1.2 Generalized Block Anti-Gauss Rules, \( \ell = 2 \) .................................. 73
4.2.1.3 Generalized Block Anti-Gauss Rules, \( \ell = 3 \) .................................. 76
4.2.2 Generalized Block Anti-Gauss Rules for Functions of a Nonsymmetric Matrix .......................................................... 77
4.2.2.1 Generalized Block Anti-Gauss Rules, \( \ell = 1 \) .................................. 77
4.2.2.2 Generalized Block Anti-Gauss Rules, \( \ell = 2 \) .................................. 78
4.2.2.3 Generalized Block Anti-Gauss Rules, \( \ell = 3 \) .................................. 81
4.3 Bracketing with Gauss and Generalized Anti-Gauss Rules .................. 87
4.4 Computed Examples ................................................................. 90

5 Multiple Orthogonal Polynomials Applied to
Matrix Function Evaluation ............................................................... 97
5.1 Multiple Orthogonal Polynomials ................................................. 99
5.2 Computed Examples ................................................................. 104

6 Conclusion .............................................................................. 108

BIBLIOGRAPHY ........................................................................ 110
ACKNOWLEDGEMENTS

Praise be to Allah, who taught the man unless he knows. Thank for Allah who brought us out of ignorance and darkness to the light of science and faith. Praise be to Allah, who made us a guidance guided by not misguided nor misleading and peace and blessings be upon the most honourable prophets and messengers. Praise be to Allah, Glory and Exalted, who made me write these words after I completed my doctoral thesis and waiting for the oral defense. Praise be to Him and thank first and last to make my time precious and full of science and research.

After thanking Allah Almighty and that we never thank Him enough, I would like to thank the blessing of my life my dear parents for their patience and for letting me find a way forward. Thank you, my dear parents, for sowing the ambition in my heart until what I am now. Thank you, my dear parents, for your support and sincere prayers, which makes me very happy. Thank you, for your patience and waiting for us in days and years until we return home.

Thank you, my brothers and sisters, for your support, encouragement and motivation, special thanks for your trust and belief.

Thank you, dear Nora, Fahad, Haya, Muflih, Sara and Mohammad for carrying the torch of success and your absolute confidence, constant support, and for every moment you stand by my side in the difficult journey. May Allah bless all of you with the best reward ever. Thank you all individually.

I could not get the success easily without the support of my beloved husband and children. A ton of thanks to my husband and children for their sacrifices and patience to achieve my goal. They supported and helped me unconditionally in my success. They share the most accurate details who rejoice to my joy and sadness for my sorrow. May Allah bless
them with the best reward ever.

"Allah and his angels and the people of the heavens and the earth until the ant in its hole and even the whale to convey their prayers upon the teachers of goodness to all people". 

I would like to owe my thanks and appreciation to my Professor Lothar Reichel for his patience, advice and guidance for the last five years. I am really proud to have a professor like him who is full of knowledge and experience which makes someone learn a lot from him in both scientific and personal sides.

I am really happy to have the highest degree in my field which gives me joy and satisfaction that I am able to do it successfully. May Allah make my work and knowledge in the benefit of my nation and people.

I would also like to thank King Abdul Aziz University at Rabigh for their support and giving me the opportunity to complete the doctoral degree and thanks to the Saudi Arabian Cultural Mission (SACM) in United States for their efforts and cooperation during studying in PhD level and their support for my family. O my Nation, I am proud of you. May Allah keep you safe and secure. I would like to thank the department of Mathematics, representing the administration and the advisor of grad students at Kent State University for the cooperation and assistance provided by them. 

Last but not least, I dedicate my doctoral thesis to my dear mother and I want to say to her: the love of science is planted in your heart and inherited from you. Finally, Praise be to Allah, the Lord of the universe.

December 1st, 2017

12 Rabi I 1439 H
الأهداف

الحمد لله الذي علم الإنسان مالم يعلم، الحمد لله الذي اخرجنا من جهل الظلام إلى نور العلم
والإيمن، الحمد لله الذي جعل لنا هدأة مهتدين لا ضالين ولا مضلين، والصلاة والسلام على أئرس
الأنيبين والرسل.

أن الحمد والفضل والثناء على سببه نعمة يعمتني أكتب هذه الكلمات بعد أن اتصل علي نعمة
إمام يثبت الدخان، وانتظار ساعة المناقشة. الحمد لله والشكر أولا وأخيرا على أن أجل وقتى عامرا
بالعلم والبحث. بعد شكر الله سبحانه وتعالى فإن كنا لن بنعث به مثله الحمد ولا مقام الشكر.
ألا أتقوم بالشكر لوالدي العزيز overflowing على صبرهما ودعاهما لي بأن أخذ السير وأحقق الهدف، شكرًا
لك ولدي العزيز على بذل الطموح في قلبي حتى أبلو ما أنا في هذا من علم ونور. شكرًا لك ولدتي
العذراء على دعمك ودعائك المخلص الذي يحل صدري ويريح على قلبي. شكرًا لك على صبرك
وانتظارك لنا بالأيام والآوار حتى نعود لأرض الوطن.
شكرًا لك إخوتي جمعًا على دعمك وتشجيعك المحفز الدافع. شكرًا على قلبي، أشكرك عزيزي
نوره على عمل مشعل العلم وعلى تفتيك المطلق. شكرًا عزيزي هيا على تعزيزك الدائم، شكرًا فهد
وسارة ومفتي وملك عزيزي محمد. شكرًا لك جمعًا ووداء فردا.
لا ينال العلم براعة الجسد بل ينال العلم بذيل الألفتحيات، استغرقت رحلة الدكتوراه الكثير من
النجاح من يشاركوني أوق الفاصل وهو من يفرحون لفرحني ويجزون لحزني، عى الله أن
يجزيهم خير الجزاء وأوفروا.
رداً على الله وملاكاته وأهل السماوات والأرض حتى النملة في جها وعذب النحو ليصلون على معلمي
الناس الخير. وما علمي ذلك إلا خير أدين بالشكر والفضل إلى معلمي الفاحش /لوثر على جهدي وصوله
واضحه إرشادة في طيلة الخمس سنوات المنصرمة. وإلى أثر أن من الله علي يدكتور مثله تملك من
الخبرة والعلم ما يجعل الإنسان يتعلم منه الكثير على المستوى الدراسي والشخصي.
أريد من معاذننا ذا أن يبلغنا
ما ليس يبلغه من نفسه الزمن
ما أصدق قوله في همة نفسي فإنني أريد أن تبلغ نفسيعالم يستطيع الدهر نفسه. فاني وأن ذلك لأشكر نفسي على ثقتها وهمتها وطموحها لبلوغ مراتب المجيد ودرجات العلي لعل الله أن يجعل عملي وعلمي من صالح العمل وينفع به الأمه.

كما أناقذ بالشكر لجامعة الملك عبد العزيز فرع رابع على ثقتها ومنحي الفرصة لإكمال مرحلة الدكتوراه، والشكر موصول للملحقية الثقافية السعودية في الولايات المتحدة الأمريكية على جهودهم وخدماتهم المقدمة لي ولعائلتي طيلة مرحلة الدراسة. وما ذالك إذاً بك يا وطني عسى الله أن يديم عليك أمنك وأمانك. كما اشكر قسم الرياضيات في جامعة كنت ممتهناً بالأدارة ومشارف الدراسات العليا على الخدمات والدعم المقدم من جنبهم.

وفي نهاية المطاف فإنني أهدي رسالتي التي بين يديكم لوالديتي العزيزة ولكي أخبرها بأن حب العلم مغروس في قلبي وورثته منك عسى الله أن يكتب لنا الأجر جميعاً. واخير دعوانا أن الحمد لله رب العالمين.

حرر في ١ ديسمبر ٢٠١٧
١٢ ربيع الأول ١٤٣٩
CHAPTER 1

Introduction

The aim of this dissertation is to describe and explain the beautiful mathematical relationship between matrices, orthogonal polynomials, quadrature rules and the Lanczos algorithm. Primary concerns are to obtain efficient numerical methods to estimate or in some cases to bound quantities like $w^T f(A)v$, where $v, w$ are given vectors or block vectors, $A$ is a symmetric or nonsymmetric matrix, and $f$ a smooth function. These numerical methods are used as tools to solve and analyze important problems arising in science and industry, for example the solution of large-scale linear discrete ill-posed problems and the analysis of large-scale complex networks.

In the early 1970s, Gene Golub and collaborators exploited the connection between matrix functions of the form $w^T f(A)v$, Stieltjes integrals, Gauss-type quadrature rules, and the Lanczos process to derive powerful inexpensive algorithms for the computation of upper and lower bounds for $w^T f(A)v$. This dissertation further elaborates on the application of Gauss-type quadrature rules and anti-Gauss quadrature rules to matrix computations. In addition to methods based on the classical Lanczos algorithm and the block Lanczos algorithm, we present methods based on the use of multiple orthogonal polynomials associated with different measures.

In chapter 2 we provide the necessary mathematical background, some notation and definitions to give the reader a good overview of the main connections between orthogonal polynomials, Gauss quadrature rules, as well as the Lanczos algorithm.

In chapter 3 we consider the computation of estimates of upper and lower bounds for matrix functions of the form $w^T f(A)v$ with $A \in \mathbb{R}^{n\times n}$ a large matrix, $f$ a function, and
$v, w \in \mathbb{R}^n$. This kind of matrix functions arises in many applications such as network analysis and the solution of ill-posed problems. When $A$ is symmetric, $u = v$, and derivatives of $f$ do not change sign in the convex hull of the spectrum of $A$, a technique described by Golub and Meurant [45] allows the computation of fairly inexpensive upper and lower bounds. This technique is based on approximating $v^T f(A) v$ by a pair of Gauss and Gauss-Radau quadrature rules. However, this approach is not guaranteed to provide upper and lower bounds when derivatives of the integrand $f$ change sign, when the matrix $A$ is nonsymmetric, or when the vectors $v$ and $w$ are replaced by “block vectors” with several columns.

In the latter situations estimates of upper and lower bounds can be computed quite inexpensively by evaluating pairs of Gauss and anti-Gauss quadrature rules. When the matrix $A$ is large, the dominating computational effort for evaluating these estimates is the evaluation of matrix-vector products with $A$ and possibly also with $A^T$. The calculation of anti-Gauss rules requires one more matrix-vector product evaluation with $A$ and maybe also with $A^T$ than the computation of the corresponding Gauss rule. The third chapter describes a simplification of anti-Gauss quadrature rules that requires the evaluation of the same number of matrix-vector products as the corresponding Gauss rule. This simplification makes the computational effort for evaluating the simplified anti-Gauss rule negligible when the corresponding Gauss rule already has been computed.

Unfortunately, it is difficult to ascertain whether the values determined by Gauss and anti-Gauss rules bracket the value of the given real-valued matrix functional. Therefore, generalizations of anti-Gauss rules have recently been described, such that pairs of Gauss and generalized anti-Gauss rules may determine upper and lower bounds for real-valued matrix functionals, even when pairs of Gauss and (standard) anti-Gauss rules do not. The available generalization requires the matrix that defines the functional to be real and symmetric. Chapter 4 extends generalized anti-Gauss rules in several ways: The real-valued matrix functional may be defined by a nonsymmetric matrix. Moreover, extensions that
can be applied to matrix-valued functions are presented. Estimates of element-wise upper and lower bounds are then determined. Finally, modifications that yield simpler formulas are described.

In chapter 5 we consider matrix functions of the form $w^T f(A)v$, where the two block vectors $w$ and $v$ have different numbers of columns. This leads to the use of multiple orthogonal polynomials. In this chapter the recurrence formulas used have more than three terms. We develop a new technique that gives a good approximation for $w^T f(A)v$. Based on this approach we make a connection between the use of multiple orthogonal polynomial and Gauss quadrature rules. We calculate the elements of a banded Hessenberg matrix by using the discretized Stieltjes procedure. We describe how this approach is related to multiple orthogonal polynomials on the real line, defined by orthogonality conditions with respect to $r > 1$ different measures.

In each chapter we include a section with computed examples to show the application of the developed numerical methods. We discuss applications to functions of a symmetric or nonsymmetric adjacency matrix, and numerical examples demonstrate the validity of the approximate bounds in this setting. Moreover, inspired by the work of Estrada and collaborators [28, 30, 31, 32, 33], we review quantities describing properties of nodes in directed or undirected networks.
CHAPTER 2

Background, Notation and Definitions

2.1 Definition of Orthogonal Polynomials

A sequence of orthogonal polynomials is a family of polynomials such that any two
polynomials of different degrees are orthogonal to each other with respect to some inner
product. We will define orthogonal polynomials in either a finite or infinite interval \([a, b]\) of
the real line. Our interest is in the problem of approximating Stieltjes integrals of the form

\[
I_f = \int_a^b f(x) dw(x),
\]

where \(w(x) \geq 0\) is a nondecreasing function defined on \([a, b]\). Here, \(-\infty \leq a < b \leq \infty\).

**Definition 1.** Let \(P\) be the linear space of polynomials of suitably low degree. We define
an inner product (relative to the measure \(dw\)) of two polynomials \(p\) and \(q \in P\)

\[
\langle p, q \rangle := \int_a^b p(x)q(x) dw(x)
\]

The norm of \(p\) is defined as

\[
\|p\|_w := \left( \int_a^b p(x)^2 dw(x) \right)^{\frac{1}{2}}.
\]

We also consider discrete inner products such as

\[
\langle p, q \rangle := \sum_{j=1}^m p(x_j)q(x_j) w_j^2,
\]

The values \(x_j\) are referred to as points or nodes and the values \(w_j\) as weights.
Definition 2. The polynomials $p$ and $q$ are said to be orthogonal (with respect to inner products (2.2) or (2.4) if $\langle p, q \rangle = 0$. A set of polynomials are orthonormal if they mutually orthogonal and if $\langle p, p \rangle = 1$ for each polynomial $p$ in the set. The polynomials in a set are said to be monic orthogonal polynomials if they are orthogonal, monic and their norms are strictly positive.

For more details about orthogonal polynomials; see [46] chapter 2. For our purposes in this dissertation, the most important property of orthonormal polynomials is that they satisfy a three term recurrence relation of the form

\begin{equation}
\lambda q_{j-1}(\lambda) = \beta_{j-1}q_{j-2}(\lambda) + \alpha_jq_{j-1}(\lambda) + \beta_jq_j(\lambda), \quad j = 1, 2, \ldots,
\end{equation}

with $q_{-1}(\lambda) := 0$ and $q_0(\lambda) := (1, 1)^{-1/2}$.

2.1.1 Matrix Orthogonal Polynomial

In the previous section, the coefficients of the polynomials were scalars. Since we include in this dissertation the block case for the function matrix evaluation, we also consider polynomial with matrix coefficients. There are several ways to define orthogonal polynomials whose coefficients are square matrices. The coefficients to be considered in this subsection are $2 \times 2$ matrices, but these results can easily be generalized to square matrices of any order. This will be done in later chapters.

Definition 3. For $\lambda$ real, a matrix polynomial $p_i(\lambda)$ which is, in our case, a $2 \times 2$ matrix is defined by

\begin{equation}
p_i(\lambda) = \sum_{j=0}^{i} \lambda^j C_{j}^{(i)},
\end{equation}

where the $C_{j}^{(i)}$ are given $2 \times 2$ real matrices. If the leading coefficient is the identity matrix, the matrix polynomial is said to be monic.
The measure \( dw(x) \) is now a matrix of order 2 that we suppose to be symmetric and positive semidefinite. Moreover, we assume that if \( \lambda_1 < \lambda_2 \), then \( dw(\lambda_2) - dw(\lambda_1) \) is positive semidefinite. The integral \( \int_a^b f(\lambda)dw(\lambda) \) is a \( 2 \times 2 \) symmetric matrix.

The inner product of two matrix polynomials \( p \) and \( q \) is defined as

\[
\langle p, q \rangle := \int_a^b p(\lambda)dw(\lambda)q(\lambda)^T.
\]

Note that this defines a matrix and we have to be careful about the order for the matrix multiplications under the integral sign. The matrix polynomials in a sequence \( p_k, k = 0, 1, ... \) are said to be orthonormal if

\[
\langle p_i, p_j \rangle = \delta_{i,j}I_2,
\]

where \( \delta_{i,j} \) the Kronecker delta function and \( I_2 \) the identity matrix of order 2.

**Theorem 4.** ([45]) Sequences of matrix orthogonal polynomials satisfy a block three-term recurrence,

\[
\lambda p_{j-1}(\lambda) = p_j(\lambda)\Gamma_j + p_{j-1}(\lambda)\Omega_j + p_{j-2}(\lambda)\Gamma_j^{T}T_{j-1}, \quad j = 1, 2, ..., \\
p_0(\lambda) := I_2, \quad p_{-1}(\lambda) := O_2,
\]

where the recursion coefficients \( \Gamma_j \) and \( \Omega_j \) are \( 2 \times 2 \) matrices with real entries. Moreover, \( \Omega_j \) is symmetric and \( \Gamma_j \) can be chosen to be upper triangular. The matrix \( O_2 \in \mathbb{R}^{2 \times 2} \) denotes the zero matrix.

The proof is essentially the same as the scalar case; see [46]. Introduce the matrix

\[
P_m(\lambda) := [p_0(\lambda), \ldots, p_{m-1}(\lambda)] \in \mathbb{R}^{2 \times 2m}.
\]

Then the block three-term recurrence (2.9) for the polynomials \( p_0, \ldots, p_m \) can be expressed in matrix form as

\[
\lambda P_m(\lambda) = P_m(\lambda)T_m + p_m(\lambda)\Gamma_mE_m^{T},
\]
where

\begin{equation}
T_m := \\
\begin{bmatrix}
\Omega_1 & \Gamma_T^1 & 0 \\
\Gamma_1 & \Omega_2 & \Gamma_T^2 \\
& \ddots & \ddots \\
& & \Gamma_{m-2} & \Omega_{m-1} & \Gamma_{T_{m-1}} \\
0 & \Gamma_{m-1} & \Omega_m
\end{bmatrix} \in \mathbb{R}^{2m \times 2m}
\end{equation}

and \( E_i := [e_{2i-1}, e_{2i}] \) denotes a “block axis vector” of appropriate size with \( 2 \times 2 \) blocks. Thus, the \( i \)th block of \( E_i \) is \( I_2 \) and all other blocks vanish. We note that if \( \theta_r \) is an eigenvalue of \( T_m \) and if we choose \( w = w_r \) to be a block vector of two column whose components are the first two components of the corresponding eigenvector, then \( P(\theta_r)w \) is this eigenvector (because of the relations that are satisfied) and \( \Gamma_m \) is nonsingular with \( p^T_{m}(\theta_r)w = 0 \).

### 2.2 Quadrature Rules

Consider scalar-valued quadrature rules of the form:

\begin{equation}
Qf = \sum_{i=1}^{\ell} f(x_i)w_i,
\end{equation}

where ideally \( Qf \approx I f \) for many functions of interest. As above, the \( x_i \) are the nodes of the quadrature rule \( Q \), and the numbers \( w_i \) are the weights of \( Q \). Quadrature rules have an important property, known as the degree of exactness. We say that a quadrature rule has degree of exactness \( d \) if \( d \) is the largest positive integer such that \( Qp = Ip \) for all polynomials \( p \) of degree \( d \).

It is well known that there exists a unique optimal choice of the nodes and weights such that the degree of exactness of (2.12) is \( 2\ell - 1 \). This choice is optimal in the sense that it maximizes the degree of exactness.

We denote this quadrature rule in this dissertation by \( G_{\ell} \) and it is given by:

\begin{equation}
G_{\ell}f := \sum_{i=1}^{\ell} f(\theta_i^{(\ell)})\eta_i^{(\ell)},
\end{equation}

7
for appropriate choices of $\theta_i^{(\ell)}$ and $\eta_i^{(\ell)}$. This rule is known as the $\ell$-point Gauss quadrature rule with respect to $w$, and the corresponding parameters $\theta_i^{(\ell)}$ and $\eta_i^{(\ell)}$ are called the Gauss nodes and Gauss weights, respectively. At first it may seem amazing that we can achieve this degree of exactness with only $\ell$ nodes. However, consider that when presented with the problem of finding a quadrature rule with $\ell$ nodes and degree of exactness $2\ell - 1$, the unknowns are the nodes and weights, totaling $2\ell$ in number, and the constraints also number $2\ell$, namely the exactness for polynomials of degree $0 \leq i \leq 2\ell - 1$. Thus, it is not really surprising that there exists a unique solution; on the other hand, the problem of computing the optimal weights and nodes is nonlinear, so the equal number of unknowns and constraints does not guarantee a unique solution.

The following well-known result describes the important connection between orthogonal polynomials and Gauss quadrature. See, for example, [66] for a complete proof.

**Theorem 5.** Let $\ell \in \mathbb{N}$. The nodes of the $\ell$-point Gauss quadrature rule with respect to (2.1) coincide with the zeros of the orthonormal polynomial $q_\ell \in \mathbb{P}^\ell$ with respect to $w$.

We now provide a characterization of the Gauss nodes and weights, which is central to some of the numerical methods discussed in this dissertation. See [48, 70].

**Theorem 6.** Let $T_\ell = Y_\ell \Theta_\ell Y_\ell^T$ be a spectral decomposition, where $Y_\ell^T Y_\ell = I_\ell$ and $\Theta_\ell = \text{diag}[\theta_1^{(\ell)}, \theta_2^{(\ell)}, \ldots, \theta_\ell^{(\ell)}]$ with $\theta_1^{(\ell)} < \theta_2^{(\ell)} < \ldots < \theta_\ell^{(\ell)}$. Then,

1. The eigenvalues $\theta_j^{(\ell)}$ of $T_\ell$ are the zeros of $q_\ell$, hence the nodes of the $\ell$-point Gauss quadrature rule with respect to $w$.

2. The corresponding Gauss weights are given by

$$\eta_j^{(\ell)} := \langle 1, 1 \rangle - (e_1^T Y_\ell e_j)^2, \quad 1 \leq j \leq \ell.$$
Proof. Setting $\lambda = \theta_j^{(\ell)}$ in (2.5) yields
\[
\theta_j^{(\ell)} \left[ q_0(\theta_j^{(\ell)}), \ldots, q_{\ell-1}(\theta_j^{(\ell)}) \right] = \left[ q_0(\theta_j^{(\ell)}), \ldots, q_{\ell-1}(\theta_j^{(\ell)}) \right] T_{\ell+1}\ell
\]
\[
= \left[ q_0(\theta_j^{(\ell)}), \ldots, q_{\ell-1}(\theta_j^{(\ell)}) \right] T_\ell.
\]
Thus, writing $Y_\ell = [y_1, \ldots, y_\ell]$, unit eigenvectors are given by
\[
y_j = c_j[q_0(\theta_j^{(\ell)}), q_1(\theta_j^{(\ell)}), \ldots, q_{\ell-1}(\theta_j^{(\ell)})]^T
\]
for appropriately chosen constants $c_j$. By the first column of the equation $Y_\ell Y_\ell^T = I_\ell$, for $0 \leq i \leq \ell - 1$ we have
\[
\sum_{j=1}^{\ell} c_j^2 q_0(\theta_j^{(\ell)}) q_i(\theta_j^{(\ell)}) = \delta_{0i} = \int_a^b q_0(t) q_i(t) dw(t),
\]
and letting $\eta_j^{(\ell)} := c_j^2 = <1, 1> (e_1^T Y_\ell e_j)^2$,
\[
\sum_{j=1}^{\ell} \eta_j^{(\ell)} q_i(\theta_j^{(\ell)}) = \int_a^b q_i(t) dw(t).
\]
That is,
\[
\begin{bmatrix}
q_0(\theta_1^{(\ell)}) & q_0(\theta_2^{(\ell)}) & \cdots & q_0(\theta_\ell^{(\ell)}) \\
q_1(\theta_1^{(\ell)}) & q_1(\theta_2^{(\ell)}) & \cdots & q_1(\theta_\ell^{(\ell)}) \\
\vdots & \vdots & \ddots & \vdots \\
q_{\ell-1}(\theta_1^{(\ell)}) & q_{\ell-1}(\theta_2^{(\ell)}) & \cdots & q_{\ell-1}(\theta_\ell^{(\ell)})
\end{bmatrix}
\begin{bmatrix}
\eta_1^{(\ell)} \\
\eta_2^{(\ell)} \\
\vdots \\
\eta_\ell^{(\ell)}
\end{bmatrix}
= \begin{bmatrix}
\int_a^b q_0(t) dw(t) \\
\vdots \\
\int_a^b q_{\ell-1}(t) dw(t)
\end{bmatrix}.
\]
By the invertibility of the matrix with $(i, j)$ entry $q_{i-1}(\theta_j^{(\ell)})$ (see, for example, [66]), the parameters $\eta_j^{(\ell)}$ are the Gauss weights.  

In some applications, one is interested in finding the quadrature rule with maximum degree of exactness subject to the condition of one or several prescribed nodes, but with the rest of the nodes and all of the weights free to be chosen.
In 1996, Laurie introduced the \((\ell + 1)\)-point anti-Gauss quadrature rule \(\tilde{G}_{\ell+1}\), which is chosen such that the quadrature error is exactly opposite that of the \(\ell\)-point Gauss quadrature rule for polynomials of degree at most \(2\ell + 1\), denoted \(P_{2\ell+1}\), that is, such that
\[
(I - \tilde{G}_{\ell+1}) p = -(I - G_\ell) p, \quad p \in P_{2\ell+1}.
\]
Therefore, for \(f \in P_{2\ell+1}\), the quadrature rules \(G_\ell\) and \(\tilde{G}_{\ell+1}\) provide bounds for the exact value of the integral (2.1). Of course, it is rarely the case that the function of interest, \(f\), is a polynomial. However, if \(f\) can be approximated well on the interval \([a, b]\) by a polynomial of degree at most \(2\ell + 1\), the two quadrature rules provide approximate bounds for (2.1), which in many cases constitute true bounds. This technique of computing approximate bounds was applied to matrix functions in [23]. In chapters three and four, block anti-Gauss quadrature rules are introduced and utilized in the context of network analysis.

If \(w\) has exactly \(\ell \geq 1\) points of increase on \((a, b)\) (so that \(w\) is a step function), then \(\mathcal{I}(p^2) > 0\) for all \(p \in P_{\ell-1}\), and (2.2) is an inner product when restricted to polynomials of degree at most \(\ell - 1\). In this case, the Gram-Schmidt process can be used to generate orthonormal polynomials \(q_i\) of degree up to \(\ell - 1\); the polynomial of degree \(\ell\) generated by the recurrence necessarily vanishes on the (finite) support of \(w\) (and thus cannot be normalized). Accordingly, the nodes of the \(\ell\)-point Gauss quadrature rule coincide with the support of \(w\), the weights coincide with the jumps of \(w\), and the rule is exact for all functions. Integration with respect to measures with finitely many points of increase plays an important role in the approximation of matrix functions.

2.3 The Lanczos Algorithm

In this section we introduce the Lanczos algorithm for symmetric matrices as well as its block version, and also the nonsymmetric Lanczos algorithm. These algorithms are the vehicle to compute quadrature formulas and estimate bilinear forms. The symmetric Lanczos algorithm can be viewed as a simplification of Arnoldi’s method for the particular
case when the matrix is symmetric. When $A$ is symmetric then the Hessenberg matrix of Arnoldi’s becomes symmetric tridiagonal. This leads to a three-term recurrence in the Arnoldi’s process.

Let $A$ be a real symmetric matrix of order $n$. We introduce the Lanczos algorithm as a means of computing an orthonormal basis for the Krylov subspace. Let $v$ be a given vector and

$$K_m := [v, Av, \ldots, A^{m-1}v].$$

be the Krylov matrix of dimension of $n \times m$.

The subspace that is spanned by the columns of the matrix $K_m$ is called a Krylov subspace and denoted by $\mathcal{K}_m(A,v)$. There is a strong relationship between the Lanczos algorithm and orthogonal polynomials. To begin, recall that, if the grade of $v$ is not less than $m$, then the subspace $\mathcal{K}_m(A,v)$ is of dimension $m$ and consists of all vectors of the form $q(A)v$, where $q$ is polynomial with degree not exceeding $m - 1$. In this case there is an isomorphism between $\mathcal{K}_m(A,v)$ and $\mathbb{P}^{m-1}$, the space of polynomials of degree not exceeding $m - 1$, which is defined by

$$q \in \mathbb{P}^{m-1} \longrightarrow x = q(A)v \in \mathcal{K}_m(A,v).$$

Moreover, we may equip the subspace $\mathbb{P}^{m-1}$ with the inner product

(2.14) $\langle p, q \rangle = (p(A)v, q(A)v)$.

This is indeed an inner product for $p, q \in \mathbb{P}^{m-1}$ when $m$ does not exceed $\mu$, the grade of $v$. The symmetric Lanczos algorithm defined in Chapter 3 determines a sequence of orthonormal vectors $v_1, v_2, \ldots$, which can be expressed as

$$v_i = q_{i-1}(A)v$$

for certain polynomials $q_{i-1}$ of degree $i - 1$. The orthogonality of the $v_i$’s translates into the orthogonality of these polynomials with respect to the inner product (2.14).
It is known that real orthonormal polynomials satisfy a three-term recurrence. Moreover, the Lanczos procedure is nothing but the Stieltjes algorithm for computing a sequence of orthonormal polynomials with respect to the inner product (2.14). It is known that the characteristic polynomial of the tridiagonal matrix

\[
 T_m := \begin{bmatrix}
 \alpha_1 & \beta_1 & & 0 \\
 \beta_1 & \alpha_2 & \beta_2 & \\
 & \beta_2 & \alpha_3 & \\
 & & \ddots & \ddots & \beta_{m-2} \\
 & & & \ddots & \ddots & \beta_m \\
 0 & & & & \beta_{m-2} & \alpha_{m-1} & \beta_{m-1} \\
 & & & & \beta_{m-1} & \alpha_m 
\end{bmatrix} \in \mathbb{R}^{m \times m}
\]

(2.15)

produced by the Lanczos algorithm minimizes the norm \( \| \cdot \| \) over the monic polynomials. Lanczos algorithm is described by Algorithm 1 in chapter 3.

When the matrix \( A \) is not symmetric, we cannot generally construct a vector \( v_{i+1} \) orthogonal to all the previous basis vectors by using only two vectors \( v_i \) and \( v_{i-1} \) as is done in the symmetric Lanczos algorithm. In other words, we lose the nice property of having a short recurrence. To work around this problem, an algorithm for nonsymmetric matrices was introduced by C. Lanczos in 1950. Its goal is to construct two biorthogonal sequences of vectors for a nonsymmetric matrix \( A \). Fortunately, this can be done using short recurrences. The drawback is that the algorithm may break down. The standard development of a Lanczos algorithm for nonsymmetric matrices depends upon using the matrix \( A \) and its transpose \( A^T \). The reader can find the full nonsymmetric Lanczos algorithm in chapter 3.

We also consider the symmetric block Lanczos algorithm that was proposed by Golub and Underwood; see [49]. We restrict ourselves to the case of \( 2 \times 2 \) blocks. Let \( X_0 \) be an \( n \times 2 \) given matrix, such that \( X_0^T X_0 = I_2 \) where \( I_2 \) is the \( 2 \times 2 \) identity matrix. Let \( X_{-1} = 0 \)
be an $n \times 2$ matrix, and let $A \in \mathbb{R}^{n \times n}$ be symmetric. Then, for $k = 1, 2, \ldots$

$$
\Omega_k = X_{k-1}^TAX_{k-1},
$$

$$
(2.16) \quad R_k = AX_{k-1} - X_{k-1}\Omega_k - X_{k-2}\Gamma_{k-1}^T,
$$

$$
X_k\Gamma_k = R_k.
$$

The last step of the algorithm is QR factorization of $R_k$ such that $X_k$ is $n \times 2$ with $X_k^TX_k = I_2$. The matrix $\Gamma_k$ is $2 \times 2$ upper triangular and $\Omega_k$ is a $2 \times 2$ symmetric coefficient matrix.

The matrix $R_k$ may be rank deficient and in this case $\Gamma_k$ is singular. How to address this problem is described in [49]. One of the columns of $X_k$ can be chosen arbitrarily. To complete the algorithm, we choose this column to be orthogonal to the other column of $X_k$ and the previous block vectors $X_j$. We can, for instance, choose (randomly) another vector and orthogonalize it against the previous ones. The block tridiagonal matrix that is produced by the algorithm has the same structure as (2.11).

The block Lanczos algorithm generates a sequence of matrices $X_j$ such that

$$
X_j^TX_i = \delta_{ij}I_2.
$$

We can relate the iterates $X_i$ to a matrix polynomial $p_k$. In this chapter we presented the symmetric block Lanczos algorithm for block size $k = 2$; see chapter 3 for the block Lanczos algorithms for block size $k \geq 2$ and $A$ symmetric or nonsymmetric.
CHAPTER 3

Simplified Anti-Gauss Quadrature Rules

We consider matrix functions of the form

\[(3.1) \quad F(A) := w^T f(A)v,\]

where \(A \in \mathbb{R}^{n \times n}\) is a large matrix, \(v, w \in \mathbb{R}^n\) are vectors, \(f\) is a function such that \(f(A)\) is defined, and the superscript \(T\) denotes transposition. Also expressions of the form

\[(3.2) \quad F(A) := W^T f(A)V,\]

where \(V, W \in \mathbb{R}^{n \times k}\) are skinny “block vectors” with \(2 \leq k \ll n\) columns will be discussed.

The need to evaluate matrix functions of the forms (3.1) and (3.2) arises in a variety of applications; see [6, 8, 9, 15, 16, 19, 20, 35, 39, 46, 54, 60] for many illustrations. Functions \(f\) of interest include

\[f_1(t) := (t + \mu)^{-1}, \quad f_2(t) := \exp(\mu t), \quad f_3(t) := \cosh(\sqrt{t}),\]

where \(\mu\) is a positive parameter. For instance, expressions of the form \(f_1(A^T A)\) arise in Tikhonov regularization of linear discrete ill-posed problems; see, e.g., [19, 20]. Then \(\mu \geq 0\) is a regularization parameter and, typically, computations have to be carried out for several values of this parameter. Applications with the function \(f_2\) include the solution of systems of constant-coefficient ordinary differential equations. Here the parameter \(\mu\) represents time [42] and the functions (3.1) and (3.2) furnish linear combinations of several solution components. Further, expressions \(\mu f_1(\mu^2 A)\) and \(f_2(A)\) arise in network analysis with \(A\) an adjacency matrix for the graph that represents the network, and so do the functions \(f_3(A^T A)\) and \(f_3(AA^T)\); see [4, 10, 31] for illustrations.
The computational effort required to evaluate (3.1) or (3.2) for one or several values of \( \mu \) by first calculating \( f(A) \) and then computing (3.1) or (3.2) may be prohibitive when the matrix \( A \) is large; see Higham [52] for descriptions and analyses of many algorithms for the evaluation of matrix functions. Moreover, the matrix \( f(A) \) generally is dense also when \( A \) is sparse, in which case the storage requirement for \( f(A) \) can be problematic. It is therefore important to be able to approximate expressions of the forms (3.1) and (3.2) without explicitly computing \( f(A) \) when \( A \) is a large matrix.

Golub and Meurant [45, 46] describe a technique for determining upper and lower bounds for expressions of the form (3.1) when the matrix \( A \) is symmetric, upper or lower bounds for the spectrum of \( A \) are known, and the derivatives of \( f \) do not change sign on the convex hull of the spectrum. This technique exploits the connection between matrix functions (3.1) with \( w = v \), Stieltjes integrals, Gauss-type quadrature rules, and the symmetric Lanczos process. We will outline this technique. Introduce the spectral factorization

\[
A = SAS^{-1}, \quad \Lambda = \text{diag}[\lambda_1, \lambda_2, \ldots, \lambda_n],
\]

with the eigenvalues ordered according to \( \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n \) and \( S \in \mathbb{R}^{n \times n} \) an orthogonal matrix. Thus, \( S^{-1} = S^T \). Then

\[
f(A) = Sf(\Lambda)S^{-1}.
\]

The rare situation when the matrix \( A \) does not have a spectral factorization is discussed by Pozza et al. [64]. Let \( v \in \mathbb{R}^n \) be a unit vector and let \( w = v \). Define the vector \([\nu_1, \nu_2, \ldots, \nu_n] := v^T S \). Then the function (3.1) can be written as

\[
F(A) = v^T S f(\Lambda) S^{-1} v = \sum_{j=1}^{n} f(\lambda_j) \nu_j^2.
\]

The right-hand side can be expressed as a Stieltjes integral

\[
\mathcal{I}f := \int_{-\infty}^{\infty} f(t) d\nu(t),
\]
where the distribution function \( \nu(t) \) is a nondecreasing step function defined on \( \mathbb{R} \) with a jump of height \( \nu_j^2 \) at \( t = \lambda_j \) for \( 1 \leq j \leq n \). It follows from \( \|v\| = 1 \) that the measure \( d\nu(t) \) is of total mass one. Here and throughout this dissertation \( \| \cdot \| \) denotes the Euclidean vector norm or the spectral matrix norm.

The \( m \)-point Gauss quadrature rule associated with the measure \( d\nu(t) \) is of the form

\[
(3.6) \quad G_m f := \sum_{j=1}^{m} f(\theta_j) \gamma_j^2,
\]

and is characterized by the property that

\[
(3.7) \quad I f = G_m f \quad \forall f \in \mathbb{P}^{2m-1},
\]

where \( \mathbb{P}^{2m-1} \) denotes the set of polynomials of degree at most \( 2m - 1 \). The nodes \( \theta_j \) of the quadrature rule are distinct and known to be the zeros of an \( m \)th degree orthogonal polynomial with respect to the inner product

\[
(3.8) \quad (f, g) := I(fg).
\]

When the integrand \( f \) is \( 2m \) times continuously differentiable in the interval \([\lambda_1, \lambda_n]\), the error in the quadrature rule can be expressed as

\[
(3.9) \quad E_m f := (I - G_m)f = \frac{f^{(2m)}(\theta_G)}{(2m)!} \cdot \int_{-\infty}^{\infty} \prod_{j=1}^{m}(t - \theta_j)^2 d\nu(t)
\]

for some \( \theta_G \in [\lambda_1, \lambda_n] \), where \( f^{(2m)}(t) \) denotes the \( 2m \)th derivative; see, e.g., [44, 46] for proofs. It follows that when \( f^{(2m)}(t) \) is of known constant sign in the interval \([\lambda_1, \lambda_n]\), the sign of the error \( E_m f \) also is known. For instance, when \( f^{(2m)}(t) > 0 \) for \( \lambda_1 \leq t \leq \lambda_n \), we have \( E_m f > 0 \) and, therefore, \( G_m f < If \). If in addition \( f^{(2m+1)}(t) < 0 \) for \( \lambda_1 \leq t \leq \lambda_n \), then an upper bound for \( If \) can be determined analogously by the application of an \((m+1)\)-point Gauss-Radau quadrature rule with a fixed node at \( t \leq \lambda_1 \); see [45, 46] for details.

If \( f^{(2m)}(t) \) or \( f^{(2m+1)}(t) \) changes sign in \([\lambda_1, \lambda_n]\), then the values of pairs of Gauss and Gauss-Radau rules are not guaranteed to bracket \( If \). This also is the case when we consider
expressions of the form (3.2) with $k \geq 2$. Of course, pairs of Gauss and Gauss-Radau rules can be applied to estimate the quadrature error also in these situations, but there is no theoretical foundation that sheds light on how accurate the error estimates obtained in this manner may be. Moreover, the allocation of the fixed node of a Gauss-Radau rule requires an upper or lower bound for the eigenvalues of $A$. The methods proposed in the present chapter do not require knowledge about the spectrum of $A$.

A seemingly natural approach to estimate the error in the Gauss rule (3.6) is to evaluate the rule $G_{m+1}f$ and use the difference $G_{m+1}f - G_m f$ as an estimate for the error in $G_m f$. However, computed examples reported in [25] show that this difference may be a poor approximation of the integration error (3.9). Moreover, the evaluation of $G_{m+1}f$ requires the computation of one more step of the Lanczos process than the calculation of $G_m f$ and the methods of this chapter.

We note that when the technique described by Golub and Meurant [45, 46] can be applied to determine upper and lower bounds for the expression $v^T f(A) v$, it also can be used to compute upper and lower bounds for (3.1) with $w \neq v$ by using the decomposition

$$w^T f(A) v = \frac{1}{4} ((v + w)^T f(A)(v + w) - (v - w)^T f(A)(v - w))$$

and bounding each term in the right-hand side separately.

Laurie [55] introduced the following quadrature rules, which he referred to as anti-Gauss rules. The $(m + 1)$-point anti-Gauss rule associated with the measure $d\nu(t)$,

$$\tilde{G}_{m+1} f = \sum_{j=1}^{m+1} f(\tilde{\theta}_j) \tilde{\gamma}_j^2,$$

is characterized by the property that

$$(I - \tilde{G}_{m+1}) f = -\mathcal{E}_m f \quad \forall f \in \mathbb{P}^{2m+1},$$

where $\mathcal{E}_m$ denotes the quadrature error (3.9). Pairs of Gauss and anti-Gauss rules, $G_m f$ and $\tilde{G}_{m+1} f$, can be applied to determine inexpensively computable estimates of upper and
lower bounds for the function (3.1) in situations when there is no theoretical justification for using pairs of Gauss and Gauss-Radau rules to estimate the error in the former, for example when

(i) the matrix $A \in \mathbb{R}^{n \times n}$ is symmetric and derivatives of the integrand $f$ change sign in the convex hull of the spectrum of $A$,

(ii) the matrix $A \in \mathbb{R}^{n \times n}$ is symmetric and a required upper or lower bound for the largest or smallest eigenvalues of $A$ is not available, and

(iii) the matrix $A \in \mathbb{R}^{n \times n}$ is nonsymmetric.

Moreover, there is no theoretical foundation for using pairs of block Gauss and block Gauss-Radau rules for estimating the error in block Gauss rules when the latter are used to approximate expressions of the form (3.2) with block vectors $V$ and $W$ that have $k \geq 2$ columns.

Assume for the moment that the matrix $A \in \mathbb{R}^{n \times n}$ is symmetric and very large. The $m$-node Gauss rule $G_m$ associated with the expression (3.1) is computed by carrying out $m$ steps of the symmetric Lanczos process; see [45, 46] or below. The evaluation of matrix-vector products with $A$ is the dominating computational work for calculating $G_m f$ and may be expensive.

The evaluation of the $(m+1)$-node anti-Gauss quadrature rule, $\tilde{G}_{m+1} f$, which is used in conjunction with the $m$-node Gauss rule $G_m f$ to determine estimates of upper and lower bounds for (3.1), requires the computation of one more step of the Lanczos process than the evaluation of $G_m f$ and, therefore, demands the evaluation of one more matrix-vector product with the matrix $A$. To reduce the computational burden, it would be desirable to be able to pair $G_m f$ with a quadrature rule that uses the same Lanczos decomposition. It is the purpose of this chapter to describe such a quadrature rule. We will refer to this new
type of rules as simplified anti-Gauss rules.

This chapter is organized as follows. Section 3.1 reviews the computation of Gauss and anti-Gauss rules when applied to matrix functions of the form (3.1) with a symmetric matrix $A$ and $v = w$, and introduces simplified anti-Gauss rules for determining estimates of upper and lower bounds. The computation of Gauss and simplified anti-Gauss rules for expressions (3.1) with a nonsymmetric matrix $A \in \mathbb{R}^{n \times n}$ is discussed in Section 3.2, and matrix functions of the form (3.2) are considered in Section 3.3 for $A$ symmetric and $V = W \in \mathbb{R}^{n \times k}$, $2 \leq k \ll n$, and in Section 3.4 for $A$ nonsymmetric and $V \neq W$, $V, W \in \mathbb{R}^{n \times k}$. Numerical examples are presented in Section 3.5.

3.1 Simplified Anti-Gauss Rules for Symmetric Matrix Functionals

We first review some properties of Gauss quadrature rules determined by the measure $d\nu$ in (3.5). Further details can be found in [46]. Assume that $A \in \mathbb{R}^{n \times n}$ is symmetric and let $v = w$ in (3.1) be of unit norm. Gauss quadrature rules with respect to the measure $d\nu$ can be conveniently determined by applying the symmetric Lanczos algorithm to the matrix $A$ with initial vector $u_1 := v$; see, e.g., [45, 46] for a detailed discussion of the Lanczos process for symmetric matrices. Application of $m$ steps of the Lanczos process gives the Lanczos decomposition

$$A U_m = U_m T_m + \beta_m u_{m+1} e_m^T,$$

where $U_m$
where \( U_m = [u_1, u_2, \ldots, u_m] \in \mathbb{R}^{n \times m} \) and \( u_{m+1} \in \mathbb{R}^n \) satisfy \( U_m^T U_m = I_m, \| u_{m+1} \| = 1, \)
\( U_m^T u_{m+1} = 0, \) and \( \beta_m \in \mathbb{R}_+ \). Moreover, the matrix

\[
T_m := \begin{bmatrix}
\alpha_1 & \beta_1 & 0 \\
\beta_1 & \alpha_2 & \beta_2 \\
& \ddots & \ddots \\
\beta_{m-2} & \alpha_{m-1} & \beta_{m-1} \\
0 & \beta_{m-1} & \alpha_m
\end{bmatrix}
\in \mathbb{R}^{m \times m}
\]

is symmetric and tridiagonal. Throughout this chapter \( I_m \) denotes the \( m \times m \) identity matrix and \( e_j = [0, \ldots, 0, 1, 0, \ldots, 0]^T \) the \( j \)th axis vector of appropriate dimension. This matrix (3.13) can be computed by the symmetric Lanczos algorithm, which is described by Algorithm 1. Each step with this algorithm requires the evaluation of one matrix-vector product with the matrix \( A \). Iteration \( j \) determines the entries \( \alpha_j \) and \( \beta_j \) of the matrix (3.13).

In applications, \( m \ll n \). Moreover, we assume that \( m \) is small enough so that breakdown in Algorithm 1 does not occur within the first \( m \) steps and, therefore, the matrix (3.13) exists. Breakdown is rare, but fortuitous, because in case of breakdown the Gauss rule is exact for all polynomials. We will not dwell on this unusual situation further.

The relation (3.12) between the columns \( u_j \) of \( U_m \) shows that for certain polynomials \( p_{j-1} \) of degree \( j - 1 \),

\[
u_j = p_{j-1}(A)v, \quad 1 \leq j \leq m + 1.
\]

It follows from the orthonormality of the vectors \( u_j \) that

\[
(p_{j-1}, p_{k-1}) = \int_{-\infty}^{\infty} p_{j-1}(t)p_{k-1}(t)d\nu(t) = v^T S p_{j-1}(A)p_{k-1}(A)S^T v
\]

\[
= v^T p_{j-1}(A)p_{k-1}(A)v = u_j^T u_k = \delta_{j,k}.
\]
Algorithm 1 The symmetric Lanczos process.

1: **Input:** symmetric matrix $A \in \mathbb{R}^{n \times n}$, initial unit vector $v \in \mathbb{R}^n$
2: number of steps $m$.
3: $v_0 := 0 \in \mathbb{R}^n$, $\beta_0 := 0$, $v_1 := v$
4: for $j = 1$ to $m$
5: $w := Av_j - v_{j-1}\beta_{j-1}$
6: $\alpha_j := v_j^Tw$
7: $w := w - v_j\alpha_j$
8: $\beta_j := \|w\|; v_{j+1} := w/\beta_j$
9: end for
10: **Output:** Entries $\alpha_1, \alpha_2, \ldots, \alpha_m$ and $\beta_1, \beta_2, \ldots, \beta_m$ of the matrix (3.13).

Thus, the polynomials $p_j$ are orthogonal with respect to the inner product (3.8). The relation (3.12) for the columns $u_j$ of $U_m$ yields, in view of (3.14), the recurrence relation

$$
\beta_1 p_1(t) = (t - \alpha_1)p_0(t), \quad p_0(t) = 1,
$$

$$
\beta_j p_j(t) = (t - \alpha_j)p_{j-1}(t) - \beta_{j-1}p_{j-2}(t), \quad 2 \leq j \leq m,
$$

where

$$
\alpha_j = (p_{j-1}, tp_{j-1})
$$

and the $\beta_j > 0$ are determined by the requirement that $(p_j, p_j) = 1$.

The recurrence relation (3.15) can be written in the form

$$
[p_0(t), p_1(t), \ldots, p_{m-1}(t)]T_m = t[p_0(t), p_1(t), \ldots, p_{m-1}(t)]
$$

$$
-\beta_m[0, \ldots, 0, p_m(t)],
$$

which shows that the zeros of $p_m$ are the eigenvalues of $T_m$.

Introduce the spectral factorization

$$
T_m = Q_m D_m Q_m^T,
$$

where $D_m = \text{diag}[\theta_1, \theta_2, \ldots, \theta_m]$ and $Q_m^T Q_m = I_m$. It is well known that the weights of the Gauss rule (3.6) are given by $\gamma_j^2 = (e_1^T Q_m e_j)^2$, $1 \leq j \leq m$. It follows that the Gauss rule
(3.6) can be written in the form

\[(3.17) \quad G_m f = e_1^T Q_m f(D_m)Q_m^T e_1 = e_1^T f(T_m)e_1;\]

see, e.g., [46] for details. Thus, the value $G_m f$ can be determined by first computing the Lanczos decomposition (3.12) and then evaluating one of the expressions in (3.17). Since the matrix $T_m$ required in applications generally is quite small, the right-hand side of (3.17) can be conveniently computed by methods described in, e.g., [52].

We turn to the anti-Gauss rules proposed by Laurie [55] associated with the measure $d\nu$ in (3.5). These rules can be applied to help determine estimates of the error in Gauss rules, similarly as Gauss-Kronrod rules. Anti-Gauss rules have the advantage of existing also when Gauss-Kronrod rules do not; see Notaris [61] for a nice recent survey about error estimation methods for Gauss rules.

It follows from (3.9) and (3.11) that

\[(3.18) \quad \tilde{G}_{m+1} f = (2I - G_m) f \quad \forall f \in P_{2m+1},\]

i.e., $\tilde{G}_{m+1}$ is an $(m + 1)$-point Gauss rule for the functional

\[(3.19) \quad \mathcal{J} f := (2I - G_m) f.\]

Introduce the bilinear form

\[\langle f, g \rangle := \mathcal{J}(fg),\]

and let $\{\tilde{p}_j\}_{j=0}^{m+1}$ be the first $m + 2$ orthogonal polynomials with respect to $\langle \cdot, \cdot \rangle$ normalized so that $\langle \tilde{p}_j, \tilde{p}_j \rangle = 1$ for all $j$. These polynomials satisfy recurrence relations of the form

\[(3.20) \quad \tilde{\beta}_1 \tilde{p}_1(t) = (t - \tilde{\alpha}_1)\tilde{p}_0(t), \quad \tilde{p}_0(t) = 1,\]

\[\tilde{\beta}_j \tilde{p}_j(t) = (t - \tilde{\alpha}_j)\tilde{p}_{j-1}(t) - \tilde{\beta}_{j-1} \tilde{p}_{j-2}(t), \quad 2 \leq j \leq m + 1,\]

where

\[(3.21) \quad \tilde{\alpha}_j = \langle \tilde{p}_{j-1}, t\tilde{p}_{j-1} \rangle\]
and $\tilde{\beta}_j > 0$ is defined by the requirement that $\langle \tilde{p}_j, \tilde{p}_j \rangle = 1$. It follows from (3.18) and (3.7) that for polynomials $f$ and $g$ such that $fg \in \mathbb{P}^{2m-1}$, we have

$$\langle f, g \rangle = (f, g) = I(fg).$$

Application of this equality to (3.15), (3.16), (3.20), and (3.21) yields

$$\tilde{\alpha}_j = \alpha_j, \quad 1 \leq j \leq m,$$
$$\tilde{\beta}_j = \beta_j, \quad 1 \leq j < m.$$

Therefore, $\tilde{p}_j = p_j$ for $0 \leq j < m$.

Consider the polynomial

(3.22)  $\check{p}_m(t) := (t - \tilde{\alpha}_m)\check{p}_{m-1}(t) - \check{\beta}_{m-1}\check{p}_{m-2}(t),$

which also can be expressed as

(3.23)  $\check{p}_m(t) = (t - \alpha_m)p_{m-1}(t) - \beta_{m-1}p_{m-2}(t).$

This polynomial is a multiple of $p_m$ and, therefore, $G_m\check{p}_m^2 = 0$. It follows from (3.16), (3.21), (3.22), and (3.23) that

$$\check{\beta}^2_m = \langle \check{p}_m, \check{p}_m \rangle = 2I(\check{p}_m^2) - G_m(\check{p}_m^2) = 2I(\check{p}_m^2) = 2(\check{p}_m, \check{p}_m) = 2\beta^2_m.$$

We may choose $\check{\beta}_m = \sqrt{2}\beta_m$. The diagonal coefficient $\alpha_{m+1}$ is determined by carrying out one more step of the symmetric Lanczos process. Then the symmetric tridiagonal matrix associated with the anti-Gauss rule $\check{G}_{m+1}$ is given by

(3.24)  $\check{T}_{m+1} := \begin{bmatrix}
\alpha_1 & \beta_1 & & & \\
\beta_1 & \alpha_2 & \beta_2 & & \\
& \beta_2 & \alpha_3 & & \\
& & & \ddots & \ddots \\
& & & & \beta_{m-1} \\
\beta_{m-1} & \alpha_m & \sqrt{2}\beta_m & & \\
& \beta_{m-1} & \sqrt{2}\beta_m & \alpha_{m+1} & \\
0 & & & & \\
\end{bmatrix} \in \mathbb{R}^{(m+1) \times (m+1)}.$
The entries of $\tilde{T}_{m+1}$ can be computed by carrying out $m + 1$ steps of the symmetric Lanczos process (3.12). Analogously to the formula (3.17), the anti-Gauss rule (3.10) can be evaluated according to

$$
(3.25) \quad \tilde{G}_{m+1} f = e_1^T f(\tilde{T}_{m+1}) e_1.
$$

We remark that the pair of quadrature rules $G_m f$ and $\tilde{G}_{m+1} f$ may yield tighter error bounds than pairs of $G_m f$ and an associated $(m + 1)$-point Gauss-Radau rule. This is illustrated in [20, Examples 1 and 2].

The computation of $\tilde{G}_{m+1} f$ requires one more step of the symmetric Lanczos process and, hence, one more matrix-vector product evaluation with the matrix $A$, than the computation of $G_m f$. When the matrix $A$ is large, the computational effort required for evaluating each matrix-vector product may be significant. We therefore would like to modify the anti-Gauss rule $\tilde{G}_{m+1} f$ so that its computation does not require more matrix-vector product evaluations than the computation of $G_m f$. Note that the last subdiagonal entry of the matrix (3.24) is available after $m$ steps of the symmetric Lanczos process; the $(m + 1)$st step is only required to determine the last diagonal element of the matrix. We propose to replace the last diagonal entry of the matrix (3.24) by an available scalar $\hat{\alpha}_{m+1}$ and denote the matrix so obtained by $\hat{T}_{m+1}$. The associated quadrature rule is given by

$$
(3.26) \quad \hat{G}_{m+1} f = e_1^T f(\hat{T}_{m+1}) e_1.
$$

We refer to this quadrature rule as a simplified anti-Gauss rule. The results developed in this section hold for an arbitrary last diagonal entry $\hat{\alpha}_{m+1} \in \mathbb{R}$ of $\hat{T}_{m+1}$ and Section 3.5 reports computed results for several choices of $\hat{\alpha}_{m+1}$. We found the choice $\hat{\alpha}_{m+1} := \alpha_m$ to perform well.

The evaluation of the simplified anti-Gauss rule (3.26) is inexpensive when the Gauss rule $G_m f$ already has been computed. In particular, no matrix-vector product evaluations,
in addition to those required to determine $G_m f$, are required. The remainder of this section explores some properties of the quadrature rule (3.26).

Assume for the moment that we can carry out $n$ steps of the Lanczos process without breakdown. This yields an orthonormal basis $\{u_j\}_{j=1}^n$ for $\mathbb{R}^n$ and an associated sequence of orthonormal polynomials $\{p_j\}_{j=0}^{n-1}$ defined by (3.14). We have

$$(3.27) \quad f(t) = \sum_{j=0}^{n-1} \eta_j p_j(t), \quad t \in \lambda(A),$$

where $\lambda(A)$ denotes the spectrum of $A$. Since $I p_0 = 1$ and $I p_j = 0$ for $j > 0$, application of the operator $I$ to (3.27) gives $I f = \eta_0$. Therefore, applying the Gauss rule $G_m$ to (3.27), using the property (3.7), yields

$$(3.28) \quad G_m f = I f + \sum_{j=2m}^{n-1} \eta_j G_m p_j.$$

**Theorem 7.** The quadrature rule (3.26) satisfies

$$(3.29) \quad \tilde{G}_{m+1} f = I f \quad \forall f \in P_{2m-1},$$

$$(3.30) \quad \tilde{G}_{m+1} f = (2I - G_m) f \quad \forall f \in P_{2m}.$$

Assume that the coefficients $\eta_j$ in (3.27) satisfy

$$(3.31) \quad |\eta_{2m} G_m p_{2m}| \geq \max \left\{ \left| \sum_{j=2m+1}^{n-1} \eta_j G_m p_j \right|, \left| \sum_{j=2m+1}^{n-1} \eta_j \tilde{G}_{m+1} p_j \right| \right\}.$$

Then the quadrature rules $G_m f$ and $\tilde{G}_{m+1} f$ bracket $I f$.

**Proof.** Property (3.29) follows from the fact that the matrix $\tilde{T}_{m+1}$ has the leading principal submatrix (3.13). Turning to (3.30), we observe that the entries of the matrix (3.24) can be determined by the Chebyshev algorithm applied to the functional $2I - G_m$ in the order $\alpha_1, \beta_1, \alpha_2, \ldots, \sqrt{2} \beta_m, \alpha_{m+1}$. A description of the modified Chebyshev algorithm is provided by Gautschi [44, Section 2.1.7]; the (standard) Chebyshev algorithm is a special case. Each coefficient increases the degree of precision of the quadrature rule defined by (3.25) by one.
In particular, the coefficients $\alpha_1$ through $\sqrt{2} \beta_m$ define the quadrature rule (3.26) for (3.19) that is exact at least for all $f \in \mathbb{P}^{2m}$.

To show (3.31), we apply the simplified anti-Gauss rule (3.26) to the expansion (3.27). Using (3.30), we obtain

$$\tilde{G}_{m+1}f = \sum_{j=0}^{n-1} \eta_j \tilde{G}_{m+1}p_j = \sum_{j=0}^{2m} \eta_j (2I - G_m)p_j + \sum_{j=2m+1}^{n-1} \eta_j \tilde{G}_{m+1}p_j$$

(3.32)

$$= If - \eta_{2m} G_{m} p_{2m} + \sum_{j=2m+1}^{n-1} \eta_j \tilde{G}_{m+1}p_j.$$ 

Property (3.31) now follows by combining (3.28) and (3.32). □

Theorem 7 shows that when the coefficients $\eta_j$ in the expansion (3.27) converge to zero sufficiently quickly with increasing index $j$, the Gauss rule $G_m f$ and the associated simplified anti-Gauss rule $\tilde{G}_{m+1} f$ yield quadrature errors of opposite sign. It is therefore natural to consider the average quadrature rule

$$A_{2m+1}f := \frac{1}{2}(G_m + \tilde{G}_{m+1})f.$$ 

(3.33)

It follows from (3.27) and (3.32) that

$$A_{2m+1}f = If + \sum_{j=2m+1}^{n-1} \eta_j A_{2m+1}p_j,$$

which shows that the quadrature rule (3.33) is exact for all $f \in \mathbb{P}^{2m}$. Laurie [55] introduced an analogous average rule defined with the aid of the anti-Gauss rule (3.10).

3.2 Simplified Anti-Gauss Rules for Nonsymmetric Matrix Functionals

We extend the discussion of the previous section to matrix functionals of the form (3.1) with a large, possibly nonsymmetric, matrix $A \in \mathbb{R}^{n \times n}$, and vectors $v, w \in \mathbb{R}^n$ such that $v^T w = 1$. Assume that $A$ is diagonalizable, i.e., that $A$ can be factored according to (3.3). The computation and application of the quadrature rules to be discussed in this section do
not require $A$ to be diagonalizable, but this assumption simplifies the presentation. When $A$ is nonsymmetric, the eigenvector matrix $S \in \mathbb{C}^{n \times n}$ generally is not orthogonal.

Let $f(A)$ be defined by (3.4). Note that some eigenvalues may be complex-valued and the eigenvector matrix $S$ in (3.3) may have complex-valued entries. Introduce the vectors

$$[\nu_1, \nu_2, \ldots, \nu_n] := w^T S, \quad [\nu'_1, \nu'_2, \ldots, \nu'_n] := (S^{-1} v)^T.$$

Then (3.1) can be expressed as

$$F(A) = w^T S f(\Lambda) S^{-1} v = \sum_{j=1}^{n} f(\lambda_j) \nu_j \nu'_j.$$

We can represent the right-hand side of (3.34) by an integral with a measure $d\omega(z)$ with support in the complex plane, i.e.,

$$I f := \int f(z) d\omega(z).$$

It follows from $v^T w = 1$ that $\int d\omega(z) = 1$.

The $m$-point Gauss quadrature rule associated with the measure $d\omega$, if it exists, is of the form

$$G_m f := \sum_{j=1}^{m} f(\zeta_j) \gamma_j \gamma'_j, \quad \gamma_j, \gamma'_j \in \mathbb{C},$$

and is characterized by

$$I f = G_m f \quad \forall f \in \mathbb{P}^{2m-1}.$$

The nodes $\zeta_j$ of the quadrature rule (3.35) are known to be the zeros of an $m$th degree orthogonal polynomial with respect to the bilinear form

$$[f, g] := I(fg);$$

see [23, 35, 50] for discussions. The weights $\gamma_j \gamma'_j$ are defined below.

Introduce the $(m + 1)$-point anti-Gauss rule

$$\tilde{G}_{m+1} f := \sum_{j=1}^{m+1} f(\tilde{\zeta}_j) \tilde{\gamma}_j \tilde{\gamma}'_j, \quad \tilde{\gamma}_j, \tilde{\gamma}'_j \in \mathbb{C},$$

27
associated with the measure \(d\omega(z)\). It is determined by the requirement that

\[
(I - \tilde{G}_{m+1})f = -(I - G_m)f \quad \forall f \in \mathbb{P}^{2m+1}.
\]

Gauss and anti-Gauss quadrature rules with respect to the measure \(d\omega(z)\) can be computed conveniently by the nonsymmetric Lanczos algorithm applied to the matrix \(A\) with initial vectors \(v\) and \(w\); see, e.g., Fenu et al. [35]. It is well known that the nonsymmetric Lanczos algorithm may break down. However, in the present section we assume that \(m\) is small enough for this not to take place within \(m + 1\) steps. Insightful discussions on how to handle breakdowns are provided by Bai et al. [5], Brezinski et al. [13, 14], and Ye [71].

Application of \(m\) steps of the nonsymmetric Lanczos process to \(A\) with initial vectors \(v\) and \(w\) yields the decompositions

\[
AU_m = U_m T_m + \beta_m u_{m+1} e_m^T,
\]

\[
A^T V_m = V_m T_m^T + \beta'_m v_{m+1} e_m^T,
\]

where the matrices \(U_m = [u_1, u_2, \ldots, u_m] \in \mathbb{R}^{n \times m}, V_m = [v_1, v_2, \ldots, v_m] \in \mathbb{R}^{n \times m}\) and the vectors \(u_{m+1}, v_{n+1} \in \mathbb{R}^n\) satisfy \(U_m^T V_m = I_m\), \(u_{m+1}^T v_{m+1} = 1\), \(U_m^T v_{m+1} = 0\), \(V_m^T u_{m+1} = 0\), \(u_1 := v\), \(v_1 := w\), and \(\beta_m, \beta'_m \in \mathbb{R}_+\). Moreover,

\[
T_m := \begin{bmatrix}
\alpha_1 & \beta'_1 & 0 \\
\beta_1 & \alpha_2 & \beta'_2 \\
& \beta_2 & \alpha_3 \\
& & \ddots & \ddots \ \\
& & & \ddots & \ddots & \beta'_{m-2} \\
& & & & \beta_{m-2} & \alpha_{m-1} & \beta'_{m-1} \\
& & & & & \beta_{m-1} & \alpha_m \\
0 & & & & & & 0
\end{bmatrix} \in \mathbb{R}^{m \times m}
\]

is a (generally nonsymmetric) tridiagonal matrix. We assume that \(m\) is small enough so that the decompositions (3.39) exist.
The relations (3.39) between the columns $u_j$ of $U_m$ and $v_j$ of $V_m$ show that

\begin{align}
(3.41) \quad u_j &= p_{j-1}(A)v, \quad 1 \leq j \leq m + 1, \\
v_j &= q_{j-1}(A^T)w,
\end{align}

for certain polynomials $p_{j-1}$ and $q_{j-1}$ of degree $j - 1$. It follows from the biorthonormality of the vectors $u_j$ and $v_k$

\[ [q_{k-1}, p_{j-1}] = \int q_{k-1}(z)p_{j-1}(z)d\omega(z) = w^T S q_{k-1}(A)p_{j-1}(A) S^{-1} v \]

\[ = w^T q_{k-1}(A)p_{j-1}(A) v = v_k^T u_j = \delta_{j,k}. \]

Thus, the polynomials $q_k$ and $p_j$ are biorthonormal with respect to the bilinear form (3.36). They will be implicitly defined by the nonsymmetric Lanczos process described by Algorithm 2. In particular, this process determines how the $p_j$ and $q_j$ are scaled; see (3.41). We remark that it is not a priori known how many pairs $\{p_i, q_i\}$ of biorthonormal polynomials exist.

**Algorithm 2** The nonsymmetric Lanczos process.

1: **Input**: nonsymmetric matrix $A \in \mathbb{R}^{n \times n}$, initial vectors $V, W \in \mathbb{R}^n$ such that $V^T W = 1$, number of steps $m$.
2: $w_0 := 0 \in \mathbb{R}^n$, $v_0 := 0 \in \mathbb{R}^n$, $w_1 := W$, $v_1 := V$, $\beta_0 := 0$, $\beta'_0 := 0$
3: **for** $j = 1$ to $m$
4: \hspace{1em} $\alpha_j := v_j^T (A w_j - w_{j-1} \beta_j)$
5: \hspace{1em} $r := A w_j - \alpha_{j-1} w_j + w_{j-1} \beta_{j-1}$
6: \hspace{1em} $s := A^T v_j - \alpha_{j-1} v_j + v_{j-1} \beta'_{j-1}$
7: \hspace{1em} $\beta_j := \|r\|$, $w_{j+1} := r/\beta_j$
8: \hspace{1em} $\beta'_j := \|s\|$, $v_{j+1} := s/\beta'_j$
9: **end for**
10: **Output**: Entries $\alpha_1, \alpha_2, \ldots, \alpha_m$, $\beta_1, \beta_2, \ldots, \beta_m$, and $\beta'_1, \beta'_2, \ldots, \beta'_m$ of the matrix (3.40).

The relations between the columns $u_j$ of $U_m$ and $v_k$ of $V_m$ expressed by (3.39) yield
recurrence relations for the polynomials $p_j$ and $q_j$. They are given by

\[
[p_0(z), p_1(z), \ldots, p_{m-1}(z)] T_m = z [p_0(z), p_1(z), \ldots, p_{m-1}(z)] - \beta_m [0, \ldots, 0, p_m(z)],
\]

\[
[q_0(z), q_1(z), \ldots, q_{m-1}(z)] T_m^T = z [q_0(z), q_1(z), \ldots, q_{m-1}(z)] - \beta'_m [0, \ldots, 0, q_m(z)],
\]

which shows that the eigenvalues of $T_m$ are zeros of $p_m$ (and of $q_m$). We will assume that the matrix $T_m$ has $m$ distinct eigenvalues $\zeta_1, \zeta_2, \ldots, \zeta_m$.

**Proposition 8.** The Gauss rule (3.35) can be expressed as

\[
G_m f = e^T_1 f(T_m)e_1,
\]

where $T_m$ is defined by (3.40). Substituting the spectral factorization $T_m = Z_mD_mD_m^{-1}$ into (3.42) yields

\[
G_m f = e^T_1 Z_m f(D_m) Z_m^{-1} e_1,
\]

which shows that the nodes of $G_m$ are the eigenvalues of $T_m$ and the weights are the products of entries of the first row of the eigenvector matrix, $e^T_1 Z_m = [\gamma_1, \gamma_2, \ldots, \gamma_m]$ and the entries of the first column of its inverse $Z_m^{-1} e_1 = [\gamma'_1, \gamma'_2, \ldots, \gamma'_m]^T$.

**Proof.** Proofs can be found in [23, 35, 50]. Commonly the form of (3.43) is shown first, from which the expression (3.42) easily follows. These proofs assume that the matrix $T_m$ is diagonalizable. This is the generic situation. The case when $T_m$ is defective is discussed by Pozza et al. [64].

We turn to anti-Gauss rules (3.37) associated with the measure $d\omega(z)$. It follows from (3.38) that

\[
\tilde{G}_{m+1} f = (2I - G_m) f \quad \forall f \in \mathbb{P}^{2m+1},
\]
i.e., $\tilde{G}_{m+1}$ is the $(m+1)$-point Gauss rule for the functional

$$Jf := (2I - G_m)f.$$ 

Introduce the bilinear form

$$\{f, g\} := J(fg),$$

and let $\tilde{p}_0, \tilde{p}_1, \tilde{p}_2, \ldots$ and $\tilde{q}_0, \tilde{q}_1, \tilde{q}_2, \ldots$ be families of biorthonormal polynomials with respect to $\{\cdot, \cdot\}$, i.e., $\{\tilde{p}_i, \tilde{q}_j\} = 0$ for $i \neq j$, and $\{\tilde{p}_i, \tilde{q}_i\} = 1$ for all $i$.

The polynomials $\tilde{p}_i$ and $\tilde{q}_j$ satisfy three-term recurrence relations and one can show that the recursion coefficients are the entries of the tridiagonal matrix

$$\tilde{T}_{m+1} := \begin{bmatrix}
\alpha_1 & \beta'_1 & 0 \\
\beta_1 & \alpha_2 & \beta'_2 \\
\beta_2 & \alpha_3 & \\
& \ddots & \ddots & \beta'_{m-1} \\
& \ddots & \ddots & \\
0 & \beta_{m-1} & \alpha_m & \sqrt{2}\beta'_m \\
& & \sqrt{2}\beta_m & \alpha_{m+1}
\end{bmatrix} \in \mathbb{R}^{(m+1) \times (m+1)}.$$ 

The derivation of these recurrence coefficients uses the fact that for polynomials $f$ and $g$, we have

$$\{f, g\} = [f, g] = I(fg) \quad \forall fg \in \mathbb{P}^{2m-1}.$$ 

The last off-diagonal entries of the matrix $\tilde{T}_{m+1}$ are not uniquely determined; only their product is required to be $2\beta'_m\beta_m$. Details can be found in [23, 35].

The entries of $\tilde{T}_{m+1}$ can be computed by $m + 1$ steps of the nonsymmetric Lanczos process, cf. (3.39). This requires the evaluation of $m + 1$ matrix-vector products with both the matrices $A$ and $A^T$. Analogously to formula (3.42), the anti-Gauss quadrature rule (3.37) can be evaluated according to

$$\tilde{G}_{m+1}f = c_1^T f(\tilde{T}_{m+1})e_1.$$ 

31
A simplified anti-Gauss rule can be defined similarly as in Section 3.1. Let the matrix \( \tilde{T}_{m+1} \) be determined by replacing the last diagonal entry of \( \tilde{T}_{m+1} \) by an arbitrary real scalar \( \tilde{\alpha}_{m+1} \). Then \( \tilde{T}_{m+1} \) can be determined by only \( m \) steps of the nonsymmetric Lanczos process and defines the quadrature rule

\[
\hat{G}_{m+1} f = e_1^T f(\tilde{T}_{m+1}) e_1,
\]

which we refer to as a simplified anti-Gauss rule. The computation of \( \tilde{T}_{m+1} \) only requires the evaluation of \( m \) matrix-vector products with each one of the matrices \( A \) and \( A^T \). We will assume that the matrix \( \tilde{T}_{m+1} \) has a complete set of linearly independent eigenvectors.

For a discussion of the unusual situation when this is not the case, we refer to [64].

The following result holds for \( \tilde{\alpha}_{m+1} \) being an arbitrary real scalar. We found in computations the choice \( \tilde{\alpha}_{m+1} = \alpha_m \) to yield good results. We will comment on the choice of \( \tilde{\alpha}_{m+1} \) further in Section 3.5.

**Theorem 9.** The simplified anti-Gauss rule (3.44) satisfies

\[
\hat{G}_{m+1} f = I f \quad \forall f \in \mathbb{P}^{2m-1}, \quad \hat{G}_{m+1} f = (2 I - G_m) f \quad \forall f \in \mathbb{P}^{2m},
\]

where \( G_m \) is the Gauss rule (3.35). Consider the expansion of the integrand

\[
f(t) = \sum_{j=0}^{n-1} \eta_j p_j(t), \quad t \in \lambda(A),
\]

in terms of the polynomials \( p_{j-1} \) determined by (3.41) and assume that the coefficients \( \eta_j \) in (3.45) are such that

\[
|\eta_{2m} G_m p_{2m}| \geq \max \left\{ \left| \sum_{j=2m+1}^{n-1} \eta_j G_m p_j \right|, \left| \sum_{j=2m+1}^{n-1} \eta_j \hat{G}_{m+1} p_j \right| \right\}.
\]

Then the Gauss rule (3.35) and the simplified anti-Gauss rule (3.44) bracket \( I f \).

**Proof.** The proof is analogous to that of Theorem 7. A Chebyshev algorithm applicable to the situation when the measure is supported in the complex plane has to be used. Such an
algorithm is described in [18]. Specifically, this reference discusses a modified Chebyshev algorithm; the Chebyshev algorithm is a special case. The bracketing follows in the same manner as in the proof of Theorem 7.

Similarly as in Section 3.1, we define the average quadrature rule

$$A_{2m+1} := \frac{1}{2}(G_m + \hat{G}_{m+1}).$$

It follows from Theorem 9 that this rule satisfies

$$A_{2m+1}f = I f \quad \forall f \in \mathbb{R}^{2m}.$$

3.3 Simplified Block Anti-Gauss Rules for Functions of a Symmetric Matrix

This section describes analogues of the quadrature rules of Section 3.1 that can be applied to determine element-wise estimates of upper and lower bounds for matrix functions of the form (3.2) with a symmetric matrix $A \in \mathbb{R}^{n \times n}$ and block vectors $W = V \in \mathbb{R}^{n \times k}$ with $2 \leq k \ll n$ orthonormal columns. Block Gauss rules of the kind considered also are discussed in [35, 45, 46], while block anti-Gauss rules have been described in [35].

Substituting the spectral factorization (3.3) into (3.2) yields

\begin{equation}
W^T f(A)W = \tilde{W}f(\Lambda)\tilde{W}^T = \sum_{i=1}^{n} f(\lambda_i)\tilde{w}_i\tilde{w}_i = \int f(t)d\tilde{w}(t) =: I f,
\end{equation}

where $\tilde{W} = [\tilde{w}_1, \ldots, \tilde{w}_n] := W^T S \in \mathbb{R}^{k \times n}$ and $\tilde{w} : \mathbb{R} \to \mathbb{R}^{k \times k}$ is a piece-wise constant matrix-valued distribution with jumps $\tilde{w}_i\tilde{w}_i^T$ at the eigenvalues $\lambda_i$ of $A$. There is a sequence of orthonormal polynomials $p_0, p_1, p_2, \ldots$ with respect to the bilinear form defined by $d\tilde{w}$, i.e.,

$$(p_i, p_j) := I(p_i p_j) = \delta_{i,j}.$$  

The polynomials satisfy a three-term recursion relation of the form

\begin{equation}
t p_{j-1}(t) = p_j(t)\Gamma_j + p_{j-1}(t)\Omega_j + p_{j-2}(t)\Gamma_j^T, \quad j = 1, 2, \ldots,
\end{equation}

$$p_0(t) := I_k, \quad p_{-1}(t) := O_k.$$
with matrix-valued coefficients $\Gamma_j, \Omega_j \in \mathbb{R}^{k \times k}$, where $\Omega_j$ is symmetric and $\Gamma_j$ can be chosen upper triangular; see, e.g., [35, 45, 46] for details. Here $O_k \in \mathbb{R}^{k \times k}$ denotes the zero matrix. Using the matrix

$$P_m(t) := [p_0(t), \ldots, p_{m-1}(t)] \in \mathbb{R}^{k \times km},$$

the recursion relation (3.47) can be expressed as

$$tP_m(t) = P_m(t)T_m + p_m(t)\Gamma_mE_m^T,$$

where the matrix

$$T_m := \begin{bmatrix}
\Omega_1 & \Gamma_1^T & 0 \\
\Gamma_1 & \Omega_2 & \Gamma_2^T \\
\vdots & \ddots & \ddots \\
\Gamma_{m-2} & \Omega_{m-1} & \Gamma_{m-1}^T \\
0 & \Gamma_{m-1} & \Omega_m
\end{bmatrix} \in \mathbb{R}^{km \times km}$$

is symmetric, block-tridiagonal, and has bandwidth $2k + 1$. This matrix is determined by $m$ steps of the symmetric block Lanczos process applied to $A$ with initial block vector $V$. Algorithm 3 describes this process under the assumption that no breakdown takes place. In line 8 of the algorithm, one computes a QR factorization of the matrix $R_j$, i.e., $V_{j+1} \in \mathbb{R}^{n \times k}$ has orthonormal columns and $\Gamma_j \in \mathbb{R}^{k \times k}$ is upper triangular with positive diagonal entries. See [5, 35, 45, 46] for discussions on the block Lanczos process. Further, $E_m$ denotes a "block axis vector" with $k \times k$ blocks; the $m$th block is $I_k$ and all other blocks vanish.

Let $T_m = Y_m \Theta_m Y_m^T$ be a spectral factorization of $T_m$, where

$$Y_m = [y_1^{(m)}, \ldots, y_{km}^{(m)}], \quad \Theta_m = \text{diag} [\theta_1^{(m)}, \ldots, \theta_{km}^{(m)}],$$

$Y_m$ satisfies $Y_m^TY_m = I_{km}$, and $\theta_1^{(m)} \leq \cdots \leq \theta_{km}^{(m)}$. Consider the quadrature rule

$$G_m f := \sum_{i=1}^{km} f(\theta_i^{(m)}) u_i^{(m)} (u_i^{(m)})^T$$
Algorithm 3 The symmetric block Lanczos process.

1: Input: symmetric matrix $A \in \mathbb{R}^{n \times n}$, initial block vector $V \in \mathbb{R}^{n \times k}$ with orthonormal columns, number of steps $m$.

3: $V_0 := O \in \mathbb{R}^{n \times k}$, $\Gamma_0 := O_k$, $V_1 := V$

4: for $j = 1$ to $m$

5: $B := AV_j - V_{j-1}\Gamma_j^T$

6: $\Omega_j := V_j^T B$

7: $R_j := B - V_j \Omega_j$

8: $V_{j+1}\Gamma_j := R_j$

9: end for

10: Output: Blocks $\Omega_1, \ldots, \Omega_m$ and $\Gamma_1, \ldots, \Gamma_m$ of the matrix (3.48).

for the integral from (3.46), where each vector $u_i^{(m)}$ consists of the first $k$ elements of $y_i^{(m)}$.

It is shown in [35, 45, 46] that

(3.51) \[ G_m f = E_1^T f(T_m)E_1 \]

is an $m$-block Gauss quadrature rule associated with a bilinear form determined by the matrix measure $d\bar{w}$, i.e.,

(3.52) \[ G_m f = \mathcal{I} f \quad \forall f \in \mathbb{P}^{2m-1} \]

with $\mathcal{I}$ defined by (3.46).

We turn to block anti-Gauss rules. Proceeding similarly as Laurie [55] for the case of a real-valued positive measure, we define the $(m+1)$-block anti-Gauss quadrature rule $\tilde{G}_{m+1}$ to be an $(m+1)$-block quadrature rule such that

\[ (\mathcal{I} - \tilde{G}_{m+1}) f = -(\mathcal{I} - G_m) f \quad \forall f \in \mathbb{P}^{2m+1}. \]

This implies that

\[ \tilde{G}_{m+1} f = (2\mathcal{I} - G_m) f \quad \forall f \in \mathbb{P}^{2m+1}. \]

Hence, $\tilde{G}_{m+1}$ is the (ordinary) $(m+1)$-block Gauss quadrature rule with respect to the bilinear form determined by the matrix-valued function $2\mathcal{I} - G_m$. 

35
Similarly as above, the block anti-Gauss rule associated with the block Gauss rule (3.51) can be expressed as
\[ \tilde{G}_{m+1} f = E_1^T f(\tilde{T}_{m+1}) E_1, \]
where
\[ \tilde{T}_{m+1} := \begin{bmatrix}
\Omega_1 & \Gamma_1^T & 0 \\
\Gamma_1 & \Omega_2 & \Gamma_2^T \\
& \ddots & \ddots & \ddots \\
0 & \cdots & \Omega_m & \sqrt{2} \Gamma_m^T \\
& & \Gamma_{m-1} & \Omega_{m+1} & \sqrt{2} \Gamma_{m+1}
\end{bmatrix} \in \mathbb{R}^{k(m+1) \times k(m+1)} \]
(3.53)
is a symmetric block tridiagonal matrix with one more block row and block column than the matrix (3.48); see [35] for details.

The evaluation of the block matrices that make up \( \tilde{T}_{m+1} \) requires the application of \( m + 1 \) steps of a symmetric block Lanczos process to the matrix \( A \) with initial block vector \( V \) and, therefore, demands the calculation of \( m + 1 \) matrix-block-vector products with the matrix \( A \). This is one more matrix-block-vector product than is required for determining the matrix (3.48). Analogously as in Section 3.1, we can reduce the number of matrix-block-vector products by one by replacing the last diagonal block, \( \Omega_{m+1} \), of (3.53) by an arbitrary symmetric matrix \( \hat{\Omega}_{m+1} \in \mathbb{R}^{k \times k} \). We refer to the block tridiagonal \( k(m+1) \times k(m+1) \) matrix so obtained as \( \hat{T}_{m+1} \). This matrix defines the simplified block anti-Gauss quadrature rule
\[ \hat{G}_{m+1} f = E_1^T f(\hat{T}_{m+1}) E_1, \]
(3.54)
In calculations reported in Section 3.5, we let \( \hat{\Omega}_{m+1} := \Omega_{m} \), but this choice is not required for the following results.

**Theorem 10.** The quadrature rule (3.54) is exact for all \( f \in P^{2m-1} \). If the last diagonal block, \( \hat{\Omega}_{m+1} \), of the block tridiagonal matrix \( \hat{T}_{m+1} \) agrees with the last diagonal block of
Then the rule (3.54) is exact for all \( f \in \mathbb{P}^{2m} \). This is the case, e.g., when the measure \( \mathcal{w} \) is such that all diagonal blocks of \( \tilde{T}_{m+1} \) vanish, and we let \( \widehat{\Omega}_{m+1} = O_k \).

**Proof.** A proof for the case when \( \widehat{\Omega}_{m+1} \) equals \( \Omega_{m+1} \) is provided in [35, Sections 3 and 6]. The proof for the situation when \( \widehat{\Omega}_{m+1} \in \mathbb{R}^{k \times k} \) is a general symmetric matrix proceeds analogously as the proof of Theorem 7 with the Chebyshev algorithm replaced by the symmetric block Chebyshev algorithm. A symmetric modified block Chebyshev algorithm is described in [22]; the symmetric block Chebyshev algorithm is a special case.

We conclude this section with a result that shows why pairs of \( m \)-block Gauss rules and simplified \((m+1)\)-block anti-Gauss rules can provide element-wise upper and lower bounds for (3.2).

**Theorem 11.** Consider the expansion (3.27) with the \( p_j \) orthonormal polynomials associated with (3.46) and assume that for some \( 1 \leq q,r \leq k \),

\[
|\eta_{[2m+1]}|_{q,r} \geq \max \left\{ \left| \sum_{j=2m+1}^{n-1} \eta_j \mathcal{G}_m p_j \right|_{q,r}, \left| \sum_{j=2m+1}^{n-1} \eta_j \hat{\mathcal{G}}_{m+1} p_j \right|_{q,r} \right\},
\]

where \( [M]_{q,r} \) denotes the \((q,r)\)-entry of the matrix \( M \in \mathbb{R}^{k \times k} \). Then \( [\mathcal{G}_m f]_{q,r} \) and \( [\hat{\mathcal{G}}_{m+1} f]_{q,r} \) bracket \([\mathcal{I} f]_{q,r}\).

**Proof.** It follows from (3.52) that

\[
\mathcal{G}_m f = \mathcal{I} f + \sum_{j=2m}^{n-1} \eta_j \mathcal{G}_m p_j.
\]

Moreover,

\[
\hat{\mathcal{G}}_{m+1} f = \sum_{j=0}^{n-1} \eta_j \hat{\mathcal{G}}_{m+1} p_j = \sum_{j=0}^{2m} \eta_j (2\mathcal{I} - \mathcal{G}_m) p_j + \sum_{j=2m+1}^{n-1} \eta_j \hat{\mathcal{G}}_{m+1} p_j
\]

(3.56)

\[
= \mathcal{I} f - \eta_{2m} \mathcal{G}_m p_{2m} + \sum_{j=2m+1}^{n-1} \eta_j \hat{\mathcal{G}}_{m+1} p_j,
\]

and the desired result follows. \( \square \)
Corollary 12. The average rule

\[ A_{m+1} := \frac{1}{2}(G_m + \tilde{G}_{m+1}) \]

is exact for all \( f \in \mathbb{P}^{2m} \). The rule is exact for all \( f \in \mathbb{P}^{2m+1} \) if the last diagonal block \( \hat{\Omega}_{m+1} \) of the block tridiagonal matrix \( \hat{T}_{m+1} \) agrees with the last diagonal block of \( \tilde{T}_{m+1} \).

Proof. The exactness of the rule (4.30) follows by adding the expansions (3.55) and (3.56). When the last diagonal block \( \hat{\Omega}_{m+1} \) of \( \hat{T}_{m+1} \) agrees with the last diagonal block of \( \tilde{T}_{m+1} \), then the simplified block anti-Gauss rule \( \hat{G}_{m+1} \) is the block anti-Gauss rule \( \tilde{G}_{m+1} \) and the result is shown in [35].

3.4 Simplified Block Anti-Gauss Rules for Functions of a Nonsymmetric Matrix

We consider functions (3.2) with a large, possibly nonsymmetric, matrix \( A \in \mathbb{R}^{n \times n} \) and block vectors \( V, W \in \mathbb{R}^{n \times k} \), \( 2 \leq k \ll n \), such that \( V^T W = I_k \). Our derivation assumes that \( A \) is diagonalizable, but this is not necessary for the application of the block quadrature rules considered; this is similar to the situation of Section 3.2. Substituting the spectral factorization (3.3) with the possibly complex eigenvector matrix \( S \) into (3.2) and letting

\[ [w_1, \ldots, w_n] := W^T S \in \mathbb{C}^{k \times n}, \quad [w'_1, \ldots, w'_n] := (S^{-1} V)^T \in \mathbb{C}^{k \times n}, \]

we obtain

\[ W^T f(A)V = \sum_{i=1}^{n} f(\lambda_i) w_i w'_i =: \mathcal{I} f, \]

Thus, \( \mathcal{I} \) can be thought of as an integral operator associated with a complex matrix-valued measure with support in the complex plane. There are two sequences of polynomials \( p_j \) and \( q_j, j = 0, 1, \ldots \), that are biorthonormal with respect to the bilinear form \( (f, g) := \mathcal{I}(fg) \) determined by the matrix-valued function \( \mathcal{I} \) in (3.58). Define

\[ P_m(\lambda) := [p_0(\lambda), \ldots, p_{m-1}(\lambda)] \in \mathbb{R}^{k \times km}, \]

\[ Q_m(\lambda) := [q_0(\lambda), \ldots, q_{m-1}(\lambda)] \in \mathbb{R}^{k \times km}. \]
The recursion relations for the polynomials $p_j$ and $q_j$ can be expressed as

\begin{equation}
\lambda P_m(\lambda) = P_m(\lambda)T_m + p_m(\lambda)\Gamma_mE_m^T, \tag{3.59}
\end{equation}

\begin{equation}
\lambda Q_m(\lambda) = Q_m(\lambda)T_m^T + q_m(\lambda)\Delta_mE_m^T,
\end{equation}

where

\begin{equation}
T_m := \begin{bmatrix}
\Omega_1 & \Delta_1^T & 0 \\
\Gamma_1 & \Omega_2 & \Delta_2^T \\
\ddots & \ddots & \ddots \\
\Gamma_{m-2} & \Omega_{m-1} & \Delta_{m-1}^T \\
0 & \Gamma_{m-1} & \Omega_m
\end{bmatrix} \in \mathbb{R}^{km \times km}
\end{equation}

is a (generally nonsymmetric) block tridiagonal matrix with $k \times k$ blocks. The off-diagonal blocks can be chosen to be triangular; see [35] for details. The matrix (3.60) is a block tridiagonal matrix. It can be determined by $m$ steps of the nonsymmetric block Lanczos process applied to $A \in \mathbb{R}^{n \times n}$ with initial block vectors $V, W \in \mathbb{R}^{n \times k}$. The implementation described by Algorithm 4 is proposed by Bai et al. [5]. Line 9 of the algorithm computes QR factorizations of the matrices $R_j$ and $S_j$, and in line 10 the singular value decomposition of $Q_j^TQ_R$ is evaluated. Thus, $\Sigma \in \mathbb{R}^{k \times k}$ in line 10 is a diagonal matrix, whose diagonal entries are assumed to be positive, and the matrices $U$ and $Z$ in the same line are orthogonal.

We define the quadrature rule

\begin{equation}
G_m f = E_1^Tf(T_m)E_1. \tag{3.61}
\end{equation}

Under the assumption that the matrix (3.60) has distinct eigenvalues, it is shown in [35] that

\begin{equation}
G_m f = \mathcal{I} f \quad \forall f \in \mathbb{P}^{2m-1}.
\end{equation}

We therefore refer to (3.61) as an $m$-block nonsymmetric Gauss quadrature rule associated with $\mathcal{I}$ given by (3.58). Similarly as in Section 3.3, we seek to determine an associated
Algorithm 4 The nonsymmetric block Lanczos process.

1: **Input:** nonsymmetric matrix $A \in \mathbb{R}^{n \times n}$, initial block vectors $W, V \in \mathbb{R}^{n \times k}$ such that $V^T W = I_k$, number of steps $m$.
2: $W_0 := V_0 := O \in \mathbb{R}^{n \times k}$, $\Delta_0 := \Gamma_0 := O_k$, $V_1 := V$, $W_1 := W$
3: for $j = 1$ to $m$
4: $B := AV_j - V_j - 1 \Delta_j^{T} - 1$
5: $\Omega_j := W_j^T B$
6: $R_j := B - V_j \Omega_j$
7: $S_j := A^T W_j - W_j \Omega_j^T - W_j - 1 \Gamma_j^{T} - 1$
8: $Q_R R_R := R_j$, $Q_S R_S := S_j$
9: $U \Sigma Z^T := R_j$, $Q_S U \Sigma Z^T := S_j$
10: $\Gamma_j := \Sigma^T U \Sigma Z^T R_R$, $\Delta_j := \Sigma^T U \Sigma Z^T R_S$
11: $V_{j+1} := Q_R Z \Sigma^{-\frac{1}{2}}$, $W_{j+1} := Q_S U \Sigma^{-\frac{1}{2}}$
12: end for
13: Output: Blocks $\Omega_1, \Omega_2, \ldots, \Omega_m$, $\Gamma_1, \Gamma_2, \ldots, \Gamma_{m-1}$, and $\Delta_1^T, \Delta_2^T, \ldots, \Delta_{m-1}^T$
14: of the matrix (3.60).
15: 

$(m + 1)$-block quadrature rule $\hat{G}_{m+1}$ such that

$$(3.62) \quad (I - \hat{G}_{m+1}) f = -(I - G_m) f \quad \forall f \in \mathbb{P}^{2m}. $$

It can be shown analogously as in Section 3.3 that the $(m + 1)$-block nonsymmetric quadrature rule

$$(3.63) \quad \hat{G}_{m+1} f := E_1^T f (\hat{T}_{m+1}) E_1, $$

where

$$(3.64) \quad \hat{T}_{m+1} := \begin{bmatrix}
\Omega_1 & \Delta_1^T & 0 \\
\Gamma_1 & \Omega_2 & \Delta_2^T \\
\ldots & \ldots & \ldots \\
\Gamma_{m-1} & \Omega_m & \Delta_m^T \\
0 & \Gamma_m & \hat{\Omega}_{m+1}
\end{bmatrix} \in \mathbb{R}^{k(m+1) \times k(m+1)}, $$

with the last diagonal block $\hat{\Omega}_m \in \mathbb{R}^{k \times k}$ arbitrary, satisfies (3.62). The computation of the matrix (3.64) requires the application of $m$ steps of the nonsymmetric block Lanczos process.
Table 3.1: Example 13: $F(A) = u^T(I + A^2)^{-1}u$, $A$ a symmetric Toeplitz matrix.

<table>
<thead>
<tr>
<th>Errors</th>
<th>$m = 10$</th>
<th>$m = 11$</th>
<th>$m = 12$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_m f - F(A)$</td>
<td>$-7.6 \cdot 10^{-11}$</td>
<td>$-3.4 \cdot 10^{-11}$</td>
<td>$-1.5 \cdot 10^{-12}$</td>
</tr>
<tr>
<td>$G_{m+1} f - F(A)$</td>
<td>$9.9 \cdot 10^{-11}$</td>
<td>$4.1 \cdot 10^{-11}$</td>
<td>$1.8 \cdot 10^{-12}$</td>
</tr>
<tr>
<td>$A_{2m+1} f - F(A)$</td>
<td>$1.1 \cdot 10^{-11}$</td>
<td>$3.4 \cdot 10^{-12}$</td>
<td>$1.1 \cdot 10^{-13}$</td>
</tr>
</tbody>
</table>

to $A$ with initial block vectors $V$ and $W$, just like the determination of the matrix (3.60). Similarly as in Section 3.3, one can show that under suitable conditions the block quadrature rules (3.61) and (3.63) bracket (3.58) component-wise. The average of the quadrature rules (3.51) and (3.63) defines an average rule that is exact for all $f \in \mathbb{P}^{2m}$. The proofs of these statements are analogous to the proofs of the corresponding results in Section 3.3 and therefore are omitted. Numerous computed examples indicate the choice $\hat{\Omega}_{m+1} = \Omega_m$ to be suitable.

3.5 Computed Examples

This section presents a few computed examples that illustrate the performance of the simplified anti-Gauss rules. The computations for all examples were carried out in MATLAB with approximately 15 significant decimal digits. Unless specified otherwise, the calculations were done on a MacBook Pro laptop computer with a 2.6 GHz Intel Core i5 processor and 8 GB 1600 MHz DDR3 memory.

The matrices in the first few examples are not large. The purpose of these examples is to illustrate that pairs of Gauss and simplified anti-Gauss rules provide upper and lower bounds for the expressions (3.1) and (3.2) in a variety of situations. We compare the computed approximations obtained with quadrature rules to values determined by explicitly evaluating the functions (3.1) and (3.2). The last few examples discuss applications to network analysis and are too large to allow the evaluation of (3.1) and (3.2) on the MacBook Pro laptop computer.
Example 13. We would like to compute approximations of the functional

$$ F(A) := u^T (I + A^2)^{-1} u, $$

with a symmetric Toeplitz matrix $A \in \mathbb{R}^{200 \times 200}$ with first row $[1, 1/2, 1/3, \ldots, 1/200]$. The vector $u$ has normally distributed entries with zero mean and is normalized to be of unit norm. Table 3.1 displays the errors in approximations obtained by Gauss, simplified anti-Gauss, and average quadrature rules with the integrand $f(t) := 1/(1+t^2)$ with $\hat{\alpha}_{m+1} = \alpha_m$. The exact value is $F(A) \approx 1.358 \cdot 10^{-2}$. The table shows the errors in $\hat{G}_{m+1} f$ and $G_m f$ to have opposite sign and about the same magnitude for each value of $m$. The average rules are seen to be most accurate.

We remark that the results achieved with the rule $\hat{G}_{m+1}$ are not very sensitive to the choice of the last diagonal entry $\hat{\alpha}_{m+1}$ of the symmetric tridiagonal matrix $\hat{T}_{m+1}$ in (3.26), as long as this entry is roughly of the same size as the unknown last diagonal entry of the tridiagonal matrix $T_{m+1}$, which is defined by (3.13) with $m$ replaced by $m+1$. For instance, $\hat{\alpha}_{m+1} = (\alpha_m + \alpha_{m-1})/2$ gives essentially the same results as those shown in Table 3.1.

It is interesting compare the error of the (standard) anti-Gauss rules $\tilde{G}_{m+1} f$ to the error of the simplified anti-Gauss rules $\hat{G}_{m+1} f$. We have

$$ \tilde{G}_{11} f - F(A) = 1.2 \cdot 10^{-10}, \quad \tilde{G}_{12} f - F(A) = 4.4 \cdot 10^{-11}, \quad \tilde{G}_{13} f - F(A) = 1.6 \cdot 10^{-12}. $$

Thus, the errors in $\tilde{G}_{11} f$ and $\tilde{G}_{12} f$ are slightly smaller than those in the corresponding anti-Gauss rules, and so are the errors in the corresponding average rules. The error in $\tilde{G}_{13} f$ is seen to be slightly smaller than the error in $\hat{G}_{13} f$. We conclude that the simplified anti-Gauss rules give about the same errors as the (standard) anti-Gauss rules.

There are other ways to estimate the error in the computed approximations of $F(A) := u^T (I + A^2)^{-1} u$ than using pairs of Gauss and simplified anti-Gauss rules or pairs of Gauss and anti-Gauss rules. One may, for instance, evaluate pairs of Gauss and Gauss–Radau
Table 3.2: Example 14: $F(A) = u^T(I + A^2)^{-1}v$, $A$ a nonsymmetric Toeplitz matrix.

<table>
<thead>
<tr>
<th>Errors</th>
<th>$m = 14$</th>
<th>$m = 15$</th>
<th>$m = 17$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_m f - F(A)$</td>
<td>$-2.4 \cdot 10^{-5}$</td>
<td>$1.9 \cdot 10^{-3}$</td>
<td>$6.0 \cdot 10^{-4}$</td>
</tr>
<tr>
<td>$\hat{G}_{m+1} f - F(A)$</td>
<td>$2.1 \cdot 10^{-3}$</td>
<td>$-2.0 \cdot 10^{-3}$</td>
<td>$-5.3 \cdot 10^{-4}$</td>
</tr>
<tr>
<td>$A_{2m+1} f - F(A)$</td>
<td>$-1.4 \cdot 10^{-4}$</td>
<td>$-3.9 \cdot 10^{-5}$</td>
<td>$3.7 \cdot 10^{-5}$</td>
</tr>
</tbody>
</table>

rules with the latter having a prescribed node at the origin. All eigenvalues of $A$ are positive. Therefore such Gauss–Radau rules are well defined. Let $R_{m+1} f$ denote this kind of $(m+1)$-node Gauss–Radau rule. The evaluation of $R_{m+1} f$ requires the calculation of $m$ matrix-vector products with the matrix $A$, just like the evaluation of $\hat{G}_{m+1} f$. We find for $m = 12$ that $R_{m+1} f - F(A) = -3.9 \cdot 10^{-12}$. Comparison with results of Table 3.1 shows that the pair of rules $G_m f$ and $R_{m+1} f$ do not bracket $F(A)$, though $|G_m f - R_{m+1} f| = 2.4 \cdot 10^{-12}$ provides the correct order of the error in $G_m f$. The error of the average $\frac{1}{2}(G_m f + R_{m+1} f)$ is $-2.7 \cdot 10^{-12}$, which is larger in magnitude than the error in $A_{2m+1} f$; cf. Table 3.1.

We turn to the use of the difference of two Gauss rules as an error estimator and obtain for $m = 10$ the estimate $G_{m+1} f - G_m f = -4.1 \cdot 10^{-11}$ for the error in $G_m f$. A comparison with Table 3.1 shows this estimate to be of too small magnitude. It is well known that $G_{m+1} f - G_m f$ may provide a poor estimate of the error in $G_m f$; see [25]. Another reason for not using this error estimator is that its evaluation requires the same number of matrix-vector product evaluations as the computation of $G_{11} f$, $\hat{G}_{11} f$, and $A_{23} f$. Table 3.1 shows $G_{11} f$ and $\hat{G}_{11} f$ to bracket $F(A)$, and the rule $A_{23} f$ to yield significantly higher accuracy than $G_{10} f$ and $G_{11} f$. □

**Example 14.** We determine approximations of the functional

$$F(A) := u^T(I + A^2)^{-1}v,$$

where $A$ is a $200 \times 200$ real nonsymmetric Toeplitz matrix with first row and column $[1, 1/2, 1/3, \ldots, 1/200]$ and $[1, 1, \ldots, 1]^T$, respectively. The vectors $u$ and $v$ have normally
distributed random entries with zero mean; they are scaled so that $u^T v = 1$. Table 3.2 shows the errors in approximations determined by Gauss, simplified anti-Gauss, and average quadrature rules for $m = 14, 15, 17$. We report results for $\hat{\alpha}_{m+1} = \alpha_m$, but found the performance of the rules $\hat{G}_{m+1} f$ not to be sensitive to small changes of $\hat{\alpha}_{m+1}$. The exact value $F(A)$ is very close to zero and lies between the values $G_m f$ and $\hat{G}_{m+1} f$ for all $m$. The average rules are the most accurate quadrature rules.

We note that the errors in Table 3.2 are much larger than the corresponding errors in Table 3.1. This depends on the matrix $A$. In Example 13 it is symmetric with its eigenvalues in the interval $[0.38, 8.94]$, while the matrix $A$ in the present example is nonsymmetric with eigenvalues in a large area of the complex plane; see Figure 3.1.

The following two examples are concerned with matrix-valued functions of the form (3.2), where $V$ and $W$ are block vectors with two columns each. The approach to bracket matrix functionals described in [45, 46] is not guaranteed to yield error-bounds for these examples.

**Example 15.** We would like to compute an approximation of the function

$$F(A) := U^T \exp(A) U,$$

where $A$ is a $100 \times 100$ real symmetric matrix with randomly generated uniformly distributed
Table 3.3: Example 15: $F(A) = U^T \exp(A)U$, $A$ symmetric indefinite.

<table>
<thead>
<tr>
<th>Errors</th>
<th>$m = 13$</th>
<th>$m = 14$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_m f - F(A)$</td>
<td>$[-0.267, -0.046] \cdot 10^{-3}$</td>
<td>$[-0.150, -0.047] \cdot 10^{-4}$</td>
</tr>
<tr>
<td>$\hat{G}_{m+1} f - F(A)$</td>
<td>$[0.272, 0.040] \cdot 10^{-3}$</td>
<td>$[0.130, 0.036] \cdot 10^{-4}$</td>
</tr>
<tr>
<td>$A_{m+1} f - F(A)$</td>
<td>$[-0.270, -0.309] \cdot 10^{-5}$</td>
<td>$[-0.966, -0.564] \cdot 10^{-6}$</td>
</tr>
</tbody>
</table>

Eigenvalues in the interval $[-5, 5]$ and $U \in \mathbb{R}^{100 \times 2}$ is a block vector with random orthonormal columns. Table 3.3 shows the difference between the exact value $F(A) \approx \begin{bmatrix} 1.491 & 0.840 \\ 0.840 & 1.575 \end{bmatrix} \cdot 10^4$ and the approximations determined by block Gauss, simplified block anti-Gauss, and average block quadrature rules. We observe that the component-wise errors of $G_m f$ and $\hat{G}_{m+1} f$ are of opposite sign and of about the same magnitude. The choice $m = 13$ yields about 3 correct decimal digits in the block Gauss and simplified block anti-Gauss rules. Smaller errors are obtained for $m = 14$. The average block rules yield the smallest component-wise errors. The exact value of $F(A)$ is evaluated with the MATLAB function `expm`.

Table 3.3 displays results for $\hat{\Omega}_{m+1} = \Omega_m$. Also the choices $\hat{\Omega}_{m+1} = O_k$ and $\hat{\Omega}_{m+1} = I_k$ give quadrature rules with component-wise errors of the same sign and of about the same size as the errors shown in the table. □

Example 16. Consider the matrix function

$$F(A) := U^T \exp(A) V,$$

where $A$ is a $300 \times 300$ real nonsymmetric matrix with uniformly distributed random entries in the interval $[0, 1/300]$. The matrix is generated with the MATLAB command
Table 3.4: Example 16: \( F(A) = U^T \exp(A)V \), \( A \) nonsymmetric.

<table>
<thead>
<tr>
<th>Errors</th>
<th>( m = 2 )</th>
<th>( m = 3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( G_m f - F(A) )</td>
<td>\begin{bmatrix} 0.039 &amp; 0.036 \ 0.200 &amp; 0.186 \end{bmatrix} \cdot 10^{-3}</td>
<td>\begin{bmatrix} 0.019 &amp; 0.019 \ 0.099 &amp; 0.102 \end{bmatrix} \cdot 10^{-7}</td>
</tr>
<tr>
<td>( \hat{G}_{m+1} f - F(A) )</td>
<td>\begin{bmatrix} -0.039 &amp; -0.036 \ -0.201 &amp; -0.187 \end{bmatrix} \cdot 10^{-3}</td>
<td>\begin{bmatrix} -0.018 &amp; -0.019 \ -0.095 &amp; -0.098 \end{bmatrix} \cdot 10^{-7}</td>
</tr>
<tr>
<td>( A_{m+1} f - F(A) )</td>
<td>\begin{bmatrix} -0.077 &amp; -0.007 \ -0.422 &amp; -0.050 \end{bmatrix} \cdot 10^{-6}</td>
<td>\begin{bmatrix} 0.035 &amp; 0.037 \ 0.184 &amp; 0.193 \end{bmatrix} \cdot 10^{-9}</td>
</tr>
</tbody>
</table>

\( A = \text{rand}(300)/300 \). The random block vectors \( U, V \in \mathbb{R}^{300 \times 2} \) satisfy \( U^T V = I \). We have

\[
F(A) \approx \begin{bmatrix} 1.010 & 0.002 \\ 0.041 & 1.018 \end{bmatrix}.
\]

Table 3.4 shows the component-wise errors in computed approximations of \( F(A) \) determined by block Gauss, simplified block anti-Gauss with \( \hat{\Omega}_{m+1} = \Omega_m \), and average block quadrature rules. The approximations of \( F(A) \) obtained by block Gauss and simplified block anti-Gauss rules have about 3 and 7 correct decimal digits for \( m = 2 \) and \( m = 3 \), respectively. The average block rules are more accurate. \( \square \)

The remaining computed examples are concerned with the analysis of large networks. We first briefly review the application of matrix functions to network analysis. Further details can be found in, e.g., [9, 11, 31, 35]. A network is identified by a graph \( G = \{\mathcal{V}, \mathcal{E}\} \) that is defined by a set of vertices \( \mathcal{V} \) and a set of edges \( \mathcal{E} \). We assume \( G \) to be a large unweighted graph with \( n \) nodes without self-loops and multiple edges. Both undirected graphs, in which travel can occur in both directions along each edge, and directed graphs, in which some or all edges are “one way streets” will be considered. The adjacency matrix \( A = [A_{ij}]_{i,j=1}^n \) associated with \( G \) has the entry \( A_{ij} = 1 \) if there is an edge from node \( i \) to node \( j \), and \( A_{ij} = 0 \) otherwise. The adjacency matrix is symmetric if and only if \( G \) is undirected.
Typically, the number of edges is much smaller than \( n^2 \). Therefore the adjacency matrix generally is sparse.

A walk of length \( k \) is a sequence of vertices \( v_1, v_2, \ldots, v_k \) such that there is an edge from vertex \( v_i \) to vertex \( v_{i+1} \) for \( i = 1, 2, \ldots, k - 1 \). Vertices and edges may be repeated. The entry \( [A^\ell]_{ij} \) of the matrix \( A^\ell \) is equal to the number of walks of length \( \ell \) starting at node \( i \) and ending at node \( j \). Thus, given a function

\[
(3.65) \quad f(A) = \sum_{\ell=0}^{\infty} c_\ell A^\ell
\]

with positive coefficients \( c_\ell \) chosen to guarantee convergence, the entry \( [f(A)]_{ij} \) can be interpreted as a measure of the ease of traveling from node \( i \) to node \( j \) within the network. The term \( c_0 I_m \) has no specific meaning and is introduced for convenience. The coefficients \( c_\ell \) are generally chosen as decreasing functions of \( \ell \). This reduces the contribution to \( f(A) \) of long walks. A popular choice is \( c_\ell = 1/\ell! \) for all \( \ell \geq 0 \), which yields \( f(A) = \exp(A) \) and will be used in the computed examples below; see, e.g., Estrada and Higham [31] for a discussion of this and other matrix functions for network analysis. Vertex \( i \) is considered important when the entry \( [f(A)]_{ii} = e^T_i f(A) e_i \) is (relatively) large. This entry is referred to as the subgraph centrality of node \( i \). We will use the quadrature rules discussed in this chapter to approximate the subgraph centrality of several nodes.

It is easy to communicate between the vertices \( i \) and \( j \) if \( [f(A)]_{ij} = e^T_i f(A) e_j \) is (relatively) large. Let \( e = [1, 1, \ldots, 1]^T \). The quantity \( e^T f(A) e \) is referred as the total communicability of a network. A large value indicates that it is easy to communicate within the network; see Benzi and Klymko [11] for details. We will approximate the total communicability of several networks by using quadrature rules.

**Example 17.** We consider the network Yeast, which is represented by an undirected graph with 2114 vertices and 4480 edges. It describes the protein interaction of yeast. Each edge represents an interaction between two proteins [53, 67]. The data set was originally
Table 3.5: Subgraph centrality for the vertices 100, 224, and 1000 of the Yeast network.

<table>
<thead>
<tr>
<th>Errors</th>
<th>$i = 100$</th>
<th>$i = 224$</th>
<th>$i = 1000$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f(A)_{ii}$</td>
<td>$3.9 \cdot 10^0$</td>
<td>$9.1 \cdot 10^4$</td>
<td>$1.0 \cdot 10^4$</td>
</tr>
<tr>
<td>$G_m f - [f(A)]_{ii}$</td>
<td>$3.5 \cdot 10^{-11}$</td>
<td>$4.1 \cdot 10^{-9}$</td>
<td>$3.5 \cdot 10^{-10}$</td>
</tr>
<tr>
<td>$\tilde{G}<em>{m+1} f - [f(A)]</em>{ii}$</td>
<td>$-3.8 \cdot 10^{-11}$</td>
<td>$-3.6 \cdot 10^{-9}$</td>
<td>$-3.9 \cdot 10^{-10}$</td>
</tr>
<tr>
<td>$A_{2m+1} f - [f(A)]_{ii}$</td>
<td>$-1.4 \cdot 10^{-12}$</td>
<td>$2.5 \cdot 10^{-10}$</td>
<td>$-2.7 \cdot 10^{-11}$</td>
</tr>
</tbody>
</table>

Table 3.6: Example 17: $F(A) = u^T \exp(A)u$, $A$ is the symmetric adjacency matrix for the Yeast network, $u = [1, 1, \ldots, 1]^T$.

<table>
<thead>
<tr>
<th>Errors</th>
<th>$m = 7$</th>
<th>$m = 9$</th>
<th>$m = 12$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_m f - F(A)$</td>
<td>$-5.8 \cdot 10^{-2}$</td>
<td>$-2.0 \cdot 10^{-4}$</td>
<td>$-5.1 \cdot 10^{-9}$</td>
</tr>
<tr>
<td>$\tilde{G}_{m+1} f - F(A)$</td>
<td>$5.5 \cdot 10^{-2}$</td>
<td>$2.1 \cdot 10^{-4}$</td>
<td>$5.4 \cdot 10^{-9}$</td>
</tr>
<tr>
<td>$A_{2m+1} f - F(A)$</td>
<td>$-1.7 \cdot 10^{-3}$</td>
<td>$2.2 \cdot 10^{-6}$</td>
<td>$1.7 \cdot 10^{-10}$</td>
</tr>
</tbody>
</table>

included in the Notre Dame Networks Database and is now available at [7]. We determine the subgraph centrality $[f(A)]_{ii}$ for the vertices $i = 100$, $i = 224$, and $i = 1000$. Table 3.5 shows the exact values as well as the errors in approximations calculated by 12-node Gauss and simplified anti-Gauss rules. The computation of these rules requires 12 steps of the symmetric Lanczos process applied to $A$ with initial vector $e_i$. In this manner, we obtain quadrature rules that yield very accurate approximations of $[f(A)]_{ii}$ that bracket the exact values for all vertices; vertex 224 has the largest subgraph centrality among the vertices considered. A preprocessing technique for determining which ones of all the vertices in the network may have the largest subgraph centrality is described in [34].

We remark that upper and lower bounds for the subgraph centrality of a vertex can be determined by evaluating pairs of Gauss and Gauss-Radau rules as proposed by Golub and Meurant [45, 46]. This approach is illustrated by Benzi and Boito [9]. It requires that a lower bound for the eigenvalues of the adjacency matrix $A$ be available. The adjacency matrix for the Yeast network is symmetric and all derivatives of the integrand are positive.
Table 3.7: Example 18: Subgraph centrality for the vertices 50, 250, and 400 of the Air500 network.

<table>
<thead>
<tr>
<th>Errors</th>
<th>( i = 50 )</th>
<th>( i = 250 )</th>
<th>( i = 400 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f(A)_{ii} )</td>
<td>6.7 ( \cdot 10^{34} )</td>
<td>1.8 ( \cdot 10^{43} )</td>
<td>7.9 ( \cdot 10^{41} )</td>
</tr>
<tr>
<td>( \mathcal{G}<em>m f - f(A)</em>{ii} )</td>
<td>3.2 ( \cdot 10^{22} )</td>
<td>1.4 ( \cdot 10^{23} )</td>
<td>2.7 ( \cdot 10^{23} )</td>
</tr>
<tr>
<td>( \mathcal{G}<em>m f - f(A)</em>{ii} )</td>
<td>-7.0 ( \cdot 10^{22} )</td>
<td>-9.8 ( \cdot 10^{22} )</td>
<td>-2.7 ( \cdot 10^{22} )</td>
</tr>
<tr>
<td>( \mathcal{A}<em>{2m+1} f - f(A)</em>{ii} )</td>
<td>1.9 ( \cdot 10^{22} )</td>
<td>2.3 ( \cdot 10^{22} )</td>
<td>-1.6 ( \cdot 10^{21} )</td>
</tr>
</tbody>
</table>

Table 3.8: Example 18: \( F(A) = u^T \exp(A)u \), \( A \) is the nonsymmetric adjacency matrix for the Air500 network, \( u = [1, 1, \ldots, 1]^T \).

<table>
<thead>
<tr>
<th>Errors</th>
<th>( m = 7 )</th>
<th>( m = 9 )</th>
<th>( m = 12 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \mathcal{G}_m f - F(A) )</td>
<td>-1.8 ( \cdot 10^{32} )</td>
<td>-9.3 ( \cdot 10^{28} )</td>
<td>-7.3 ( \cdot 10^{24} )</td>
</tr>
<tr>
<td>( \mathcal{G}_{m+1} f - F(A) )</td>
<td>2.1 ( \cdot 10^{32} )</td>
<td>9.9 ( \cdot 10^{28} )</td>
<td>7.4 ( \cdot 10^{23} )</td>
</tr>
<tr>
<td>( \mathcal{A}_{2m+1} f - F(A) )</td>
<td>1.6 ( \cdot 10^{31} )</td>
<td>2.7 ( \cdot 10^{27} )</td>
<td>5.2 ( \cdot 10^{21} )</td>
</tr>
</tbody>
</table>

Therefore, the technique of Golub and Meurant [46], based on evaluating pairs of a Gauss rule and a suitable Gauss–Radau rule, can be applied to compute upper and lower bounds for \( F(A) \). This technique requires that an upper bound for the largest eigenvalue of \( A \) be available in order to determine a suitable Gauss–Radau rule. The application of pairs of Gauss and generalized anti-Gauss rules or simplified generalized anti-Gauss rules does not require eigenvalue bounds. We therefore believe the application of these rules may be of interest also when the technique by Golub and Meurant [46] can be used.

The total communicability of \( F(A) = e^T \exp(A)e \), where \( e = [1, 1, \ldots, 1]^T \), is approximately 221.35. Table 3.6 shows the errors in computed approximations of the total communicability determined by Gauss, simplified anti-Gauss, and average quadrature rules. Quadrature rules with only a few nodes are required to yield approximations with higher accuracy than what typically is required in applications. The approximations determined by Gauss and simplified anti-Gauss rules bracket the exact value. □

**Example 18.** We consider the Air500 network with 500 vertices and 24009 edges. This
Table 3.9: Example 19. $F(A) = u^T \exp(A)u$, $A$ is the nonsymmetric adjacency matrix for the Wikivote network, $u = [1, 1, \ldots, 1]^T$.

<table>
<thead>
<tr>
<th>Errors</th>
<th>$m = 7$</th>
<th>$m = 10$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_m f - F(A)$</td>
<td>$4.5 \cdot 10^{14}$</td>
<td>$2.1 \cdot 10^{10}$</td>
</tr>
<tr>
<td>$\tilde{G}_{m+1} f - F(A)$</td>
<td>$-8.7 \cdot 10^{14}$</td>
<td>$-1.3 \cdot 10^9$</td>
</tr>
<tr>
<td>$A_{2m+1} f - F(A)$</td>
<td>$-2.1 \cdot 10^{14}$</td>
<td>$4.1 \cdot 10^9$</td>
</tr>
</tbody>
</table>

network describes flight connections between the top 500 airports based on total passenger volume worldwide within one year from July 1, 2007, to June 30, 2008; see [12, 57]. The network is directed, i.e., the adjacency matrix is nonsymmetric.

Table 3.7 displays the subgraph centrality for the vertices 50, 250, and 400. Gauss and simplified anti-Gauss quadrature rules with $m = 12$ nodes are seen to bracket the subgraph centralities and yield very high relative accuracy.

Table 3.8 shows the errors of computed approximations of the total communicability by Gauss, simplified anti-Gauss, and average quadrature rules. The exact value is $F(A) = 3.8328 \cdot 10^{35}$. Thus, the errors shown in Table 3.8 are of small relative magnitude. The approximations determined by Gauss and simplified anti-Gauss rules bracket the exact values. \qed

**Example 19.** Our last example computes approximations of the total communicability for the directed Wikivote network with 8297 vertices and 103689 edges. The network is for administrator elections and provides vote history data: a directed edge from vertex $i$ to vertex $j$ indicates that voter $i$ voted for candidate $j$. The data set is available at SNAP (Stanford Network Analysis Platform) Network Data Sets [68]. This network yields a matrix $A$ that is much larger than in the previous examples. We are able to evaluate the approximations of the total communicability reported in Table 3.9 on the MacBook pro laptop computer used for all previous examples; however, the evaluation of the exact value $F(A) \approx 1.0796 \cdot 10^{19}$ determined by computing the matrix exponential of the adjacency matrix requires a more
powerful computer. Table 3.9 shows the errors in computed approximations determined by Gauss, simplified anti-Gauss, and average quadrature rules. □
CHAPTER 4

Generalized Block Anti-Gauss Quadrature Rules

This chapter discusses new methods for computing bounds or estimates of bounds for matrix functions of the form

\[(4.1) \quad F(A) := V^T f(A)V,\]

where \(A \in \mathbb{R}^{n \times n}\) is a large symmetric matrix and the “block vector” \(V \in \mathbb{R}^{n \times k}\) consists of \(1 \leq k \ll n\) orthonormal columns. The superscript \(T\) denotes transposition and the function \(f\) is assumed to be such that \(f(A)\) is well defined; it suffices that \(f\) is analytic in an open simply connected set in the complex plane that contains the spectrum of \(A\) in its interior, see Higham [52] for several ways to define \(f(A)\). Matrix functions of the form (4.1) arise, e.g., in network analysis, when solving linear discrete ill-posed problems by Tikhonov regularization, and in certain solution methods for partial differential equations; see, e.g., [4, 9, 22, 35, 46, 54] and references therein.

Recall the discussion of upper and lower bounds for the expression (4.1) from Section 3.1. We note that it may be difficult to determine error estimates except in very special situations, and, therefore, it can be difficult to decide how many steps of the symmetric block Lanczos algorithm to carry out to determine an approximation of (4.1) of desired accuracy.

Fenu et al. [35] describe block anti-Gauss quadrature rules that, when paired with a suitable block Gauss rule, give component-wise upper and lower bounds for (4.1) for certain functions \(f\), matrices \(A\), and block vectors \(V\). These matrix-valued block anti-Gauss rules generalize the real-valued anti-Gauss rules introduced by Laurie [55] for the approximation of integrals with respect to a real-valued positive measure on the real axis.
While it is possible to provide conditions that secure that pairs of suitable block Gauss and anti-Gauss rules yield component-wise upper and lower bounds for (4.1), it is difficult to verify whether these conditions hold for a given matrix function. Therefore, for block size $k = 1$, a generalization of real-valued anti-Gauss quadrature rules was presented in [63] that, together with a suitable real-valued Gauss rule, may bracket the functional (4.1) also when pairs of Gauss and (standard) anti-Gauss rules do not. It is one of the aims of this chapter to describe block analogues of the generalized real-valued anti-Gauss rules discussed in [63].

We propose to include the last subdiagonal block in the quadrature rule and “guess” or estimate a suitable last diagonal block to obtain a square block tridiagonal matrix. This approach yields quadrature rules that are exact for polynomials of higher degree than the corresponding rules obtained when the last subdiagonal block is discarded. We refer to the block quadrature rules determined in this manner as “simplified”. Simplified anti-Gauss and block anti-Gauss quadrature rules are described in chapter 3 and in [2]. The present chapter introduces simplified generalized block anti-Gauss rules. The latter rules are simpler to derive and use than the corresponding unsimplified generalized block anti-Gauss rules, which also are described in this chapter.

So far, we have discussed expressions (4.1) with a symmetric matrix $A$. This chapter also considers functions of the form

$$(4.2) \quad F(A) := W^T f(A)V,$$

where the matrix $A \in \mathbb{R}^{n \times n}$ is allowed to be nonsymmetric, and the block vectors $V, W \in \mathbb{R}^{n \times k}$ are biorthonormal, i.e., $W^T V = I_k$, where $I_k$ denotes the identity matrix of order $k$. An approximation of the function (4.2) can be computed by applying a few steps of the nonsymmetric block Lanczos algorithm to $A$ with initial block vectors $V$ and $W$. The approximation so obtained can be interpreted as a block Gauss quadrature rule. Fenu et al.
discuss this approach of approximating (4.2) and describe associated block anti-Gauss
quadrature rules. This chapter introduces generalized anti-Gauss rules that can be applied
to bracket the expression (4.2) in a fairly inexpensive manner. These rules generalize those
described in [63]. Also simplified versions of these quadrature rules are discussed.

This chapter is organized as follows. Section 4.1 considers the approximation of the
functions (4.1) and (4.2) when the block size \( k \) is one. A modification of the generalized
anti-Gauss rules described in [63] for (4.1) with a symmetric matrix \( A \) is described. It
allows one more moment to be matched for the same number of steps by the symmetric
Lanczos algorithm than the method in [63] and, therefore, may yield higher accuracy for
about the same computational effort. Analogous generalized anti-Gauss quadrature rules
associated with the matrix function (4.2) with \( A \) nonsymmetric, and \( V \) and \( W \) vectors, also
are described. Section 4.2 discusses block analogues of the quadrature rules considered in
Section 4.1. An analysis of when pairs of block Gauss and generalized block anti-Gauss
rules bracket the exact value is presented in Section 4.3, and a few computed examples are
described in Section 4.4, including applications to network analysis.

4.1 Scalar-valued Generalized Anti-Gauss Rules

This section first considers approximation of the functional (4.1) when \( k = 1 \) and the
matrix \( A \) is symmetric, and subsequently discusses approximation of the functional (4.2)
with \( k = 1 \) and a nonsymmetric matrix \( A \). In this section \( V \) in (4.1) is a unit vector, and
\( V \) and \( W \) in (4.2) are biorthonormal vectors.

4.1.1 Scalar-valued Functionals with a Symmetric Matrix

Let the symmetric matrix \( A \in \mathbb{R}^{n \times n} \) have the spectral factorization

\[
(4.3) \quad A = SAS^T, \quad \Lambda = \text{diag}[\lambda_1, \lambda_2, \ldots, \lambda_n],
\]
similar to Section 3.1. We introduce the Gauss quadrature rules for symmetric matrix functions, just as in that section, so we omit the details.

The polynomials $p_j$ satisfy a three-term recursion relations, which for the polynomials $p_0, p_1, \ldots, p_m$ can be expressed as

$$
\begin{bmatrix}
p_0(x) \\
p_1(x) \\
\vdots \\
p_{m-1}(x)
\end{bmatrix}
= T_m
\begin{bmatrix}
p_0(x) \\
p_1(x) \\
\vdots \\
p_{m-1}(x)
\end{bmatrix}
+ \beta_m
\begin{bmatrix}
0 \\
\vdots \\
0
\end{bmatrix},
$$

where $T_m$ (3.13) is a symmetric tridiagonal matrix defined by the recurrence coefficients for the orthonormal polynomials $p_j$.

This matrix can be computed by the symmetric Lanczos algorithm, which is described by Algorithm 1.

Let the function $f$ be continuous on the convex hull of the spectrum of $A$. Recall the $m$-point Gauss quadrature rule (3.17) associated with the integral operator (3.5). The Gauss rule (3.17) also can be expressed as

$$
\mathcal{G}_m f = \sum_{i=1}^m f(x_i^{(G)})w_i^{(G)},
$$

where the nodes $x_i^{(G)}$ are the eigenvalues of $T_m$ and the weights $w_i^{(G)}$ are the squares of the first components of normalized eigenvectors of $T_m$; it is well known that the eigenvalues of $T_m$ are pairwise distinct, see, e.g., [44, 46]. The equivalence of the expressions (3.17) and (4.4) follows by substituting the spectral factorization of $T_m$ into (3.17). The nodes and weights in (4.4) can be computed efficiently with the Golub–Welsch algorithm [43, 44, 46] or by a method described by Laurie [56].

The following $(m + \ell)$-point quadrature rules for $\ell \geq 2$ were defined in [63]. They generalize the $(m + 1)$-point anti-Gauss rule introduced by Laurie [55], which is obtained when $\ell = 1$. 

55
Definition 20. The generalized anti-Gauss quadrature rule

\( \tilde{G}^{(\ell)}_{m+\ell} f := \sum_{i=1}^{m+\ell} f(\tilde{x}_i^{(\ell)}) \tilde{w}_i^{(\ell)} \)

is an \((m + \ell)\)-point quadrature rule such that

\( (\mathcal{I} - \tilde{G}^{(\ell)}_{m+\ell}) f = -(\mathcal{I} - G_m) f \quad \forall f \in P^{2m+2\ell-1}. \)

It follows from (3.7) and (4.6) that

\( \tilde{G}^{(\ell)}_{m+\ell} f = \mathcal{I} f \quad \forall f \in P^{2m-1}. \)

We can express (4.6) as

\( \tilde{G}^{(\ell)}_{m+\ell} f = (2 \mathcal{I} - G_m) f \quad \forall f \in P^{2m+2\ell-1}, \)

which shows that \( \tilde{G}^{(\ell)}_{m+\ell} \) may be considered an \((m + \ell)\)-point Gauss quadrature rule associated with the functional \(2 \mathcal{I} - G_m\). When this functional is quasi-definite, there is a unique family of orthonormal polynomials \(\{\tilde{p}_i\}_{i=0,1,2,...} \) associated with the bilinear form

\( \langle f, g \rangle := (2 \mathcal{I} - G_m)(fg); \)

see, e.g., [24] for a discussion on bilinear forms defined by a quasi-definite functional. Thus,

\( \langle \tilde{p}_j, \tilde{p}_i \rangle = \delta_{j,i}. \)

Assuming that the polynomials \(\tilde{p}_0, \tilde{p}_1, \ldots, \tilde{p}_{m+\ell}\) exist, they satisfy a three-term recursion relation, which can be written as

\[
\begin{bmatrix}
\tilde{p}_0(x) \\
\tilde{p}_1(x) \\
\vdots \\
\tilde{p}_{m+\ell-1}(x)
\end{bmatrix}
= \tilde{T}^{(\ell)}_{m+\ell}
\begin{bmatrix}
\tilde{p}_0(x) \\
\tilde{p}_1(x) \\
\vdots \\
\tilde{p}_{m+\ell-1}(x)
\end{bmatrix}
+ \beta_m
\begin{bmatrix}
0 \\
\vdots \\
0
\end{bmatrix},
\]
where

\[
\mathcal{T}_{m+\ell}^{(\ell)} = \begin{bmatrix}
\tilde{\alpha}_1 & \tilde{\beta}_1 \\
\tilde{\beta}_1 & \tilde{\alpha}_2 & \tilde{\beta}_2 \\
\tilde{\beta}_2 & \tilde{\alpha}_3 & \ddots \\
\ddots & \ddots & \ddots & \tilde{\beta}_{m-2} \\
\tilde{\beta}_{m-2} & \tilde{\alpha}_{m-1} & \tilde{\beta}_{m-1} & \ddots \\
\tilde{\beta}_{m-1} & \ddots & \ddots & \ddots & \tilde{\beta}_{m+\ell-1} \\
0 & \ddots & \ddots & \ddots & \tilde{\beta}_{m+\ell-1} \\
& & & \tilde{\alpha}_{m+\ell} & 0
\end{bmatrix}
\]

is a tridiagonal matrix of order \(m + \ell\). Its nontrivial entries are recursion coefficients for the polynomials \(\tilde{p}_j\). The matrix (4.9) can be chosen to be real and symmetric when the functional \(2I - \mathcal{G}_m\) is positive definite. In this case the nodes and weights of the generalized anti-Gauss rule (4.5) are the eigenvalues and the squares of the first components of normalized eigenvectors, respectively, of the matrix (4.9). When \(2I - \mathcal{G}_m\) is quasi-definite, the tridiagonal matrix (4.9) may be chosen to be real nonsymmetric. In any case, the \((m + \ell)\)-point generalized anti-Gauss rule can be evaluated as

\[
\tilde{\mathcal{G}}_{m+\ell}^{(\ell)} f = e^T_1 f (\mathcal{T}_{m+\ell}^{(\ell)}) e_1,
\]

which may not require explicit computation of the nodes and weights.

Introduce the average quadrature rule

\[
\tilde{\mathcal{A}}_{m+\ell}^{(\ell)} := \frac{1}{2} (\mathcal{G}_m + \tilde{\mathcal{G}}_{m+\ell}^{(\ell)}),
\]

which generalizes the average rule \(\tilde{\mathcal{A}}_{m+1}^{(1)}\) considered by Laurie [55]. We obtain from (4.8) that

\[
\tilde{\mathcal{A}}_{m+\ell}^{(\ell)} f = \mathcal{I} f \quad \forall f \in \mathbb{P}^{2m+2\ell-1}.
\]
We turn to the determination of the entries of the matrix (4.9) and start with the case \( \ell = 1 \). The quadrature rule associated with the matrix \( \widetilde{T}_m^{(1)} \) is the anti-Gauss rule described by Laurie [55]. It follows from (3.7) and (4.7) that

\[ \widetilde{\alpha}_j = \alpha_j, \quad j = 1, \ldots, m, \]

and

\[ \widetilde{\beta}_j = \beta_j, \quad j = 1, \ldots, m - 1. \]

Therefore,

\[ \widetilde{p}_j = p_j, \quad j = 1, \ldots, m - 1. \]

Further,

\[ \widetilde{\alpha}_{m+1} = \alpha_{m+1}, \quad \widetilde{\beta}_m = \sqrt{2} \beta_m, \quad \widetilde{p}_m = \frac{1}{\sqrt{2}} p_m, \]

and it follows that

\[ \widetilde{p}_{m+1} = \frac{1}{\sqrt{2} \beta_{m+1}} (\beta_{m+1} p_{m+1} - \beta_m p_{m-1}); \]

see [55] for details. These coefficients have been introduced in Section 3.1 but for convenience of the reader we define them here again.

For every positive integer \( \ell \), the leading \((m + \ell - 1) \times (m + \ell - 1)\) principal submatrix of \( \widetilde{T}_m^{(\ell)} \) is \( \widetilde{T}_m^{(\ell-1)} \). In particular, when \( \ell = 2 \) and \( \widetilde{T}_m^{(1)} \) is known, we only have to derive expressions for the entries \( \widetilde{\alpha}_{m+2} \) and \( \widetilde{\beta}_{m+1} \) to determine \( \widetilde{T}_m^{(2)} \). These expressions are shown in the following theorem.

**Theorem 21.** Assume that \( \widetilde{\beta}_{m+1} \neq 0 \). Then the nontrivial entries in the last row of the matrix \( \widetilde{T}_m^{(2)} \), defined by (4.9), satisfy

\[
\begin{align*}
\widetilde{\alpha}_{m+2} &= \frac{\alpha_{m+2} \beta_{m+1}^2 - \alpha_m \beta_m^2}{\beta_{m+1}^2}, \\
\widetilde{\beta}_{m+1}^2 &= \beta_{m+1}^2 - \beta_m^2.
\end{align*}
\]
Proof. An equivalent result is shown in [63], where the orthogonal polynomials are monic, i.e., have leading coefficient one. The analogues of the tridiagonal matrices (3.13) and (4.9) associated with monic orthogonal polynomials have all superdiagonal entries equal to one, and the formulas corresponding to (4.12) and (4.13) differ, but are equivalent.

When the right-hand side of (4.13) is positive, then it is natural to define

$$\tilde{\beta}_{m+1} = \sqrt{\beta_{m+1}^2 - \beta_m^2}.$$  

This yields a real symmetric matrix $\tilde{T}_{m+2}^{(2)}$. If, on the other hand, the right-hand side of (4.13) is negative, then we can let $\tilde{T}_{m+2}^{(2)}$ be nonsymmetric and such that the product of the last off-diagonal entries equals the right-hand side of (4.13). For instance, we may choose the last superdiagonal entry to be unity and the last subdiagonal entry to be equal to the right-hand side of (4.13). The quadrature rule determined by the nonsymmetric tridiagonal matrix so obtained gives the desired value. This is a consequence of the following result.

**Proposition 22.** Let the matrix $C \in \mathbb{C}^{(m+2) \times (m+2)}$ be nonsingular and such that all entries but the diagonal one vanish in the first row and column. Assume that the spectral factorization

$$\tilde{T}_{m+2}^{(2)} = \tilde{S}_{m+2}^{(2)} \tilde{\Lambda}_{m+2}^{(2)} (\tilde{S}_{m+2}^{(2)})^{-1}$$

exists. Then

$$\tilde{G}_{m+2}^{(2)} f = e_1^T f (CT_{m+2}^{(2)} C^{-1}) e_1.$$  

Proof. We have to show that the value of the quadrature rule (4.10) with $\ell = 2$ is independent of the matrix $C$. This quadrature rule can be written as

$$\tilde{G}_{m+2}^{(2)} f = e_1^T \tilde{S}_{m+2}^{(2)} f (\tilde{\Lambda}_{m+2}^{(2)} (\tilde{S}_{m+2}^{(2)})^{-1} e_1,$$

see, e.g., Gragg [50] or Fenu et al. [35] for proofs. Due to the structure of the matrix $C$, we have

$$\tilde{G}_{m+2}^{(2)} f = e_1^T C \tilde{S}_{m+2}^{(2)} f (\tilde{\Lambda}_{m+2}^{(2)} (\tilde{S}_{m+2}^{(2)})^{-1} C^{-1} e_1 = e_1^T f (CT_{m+2}^{(2)} C^{-1}) e_1.$$
The different choices of the last sub- and super-diagonal entries of $\tilde{T}_m^{(2)}$ correspond to different diagonal matrices $C$. 

Using results of Pozza et al. [64], the requirement that $\tilde{T}_m^{(2)}$ has a spectral factorization (4.14) can be removed.

The computation of the entries of $\tilde{T}_m^{(2)}$ requires that $m + 2$ steps with Algorithm 1 be carried out. The matrix $\tilde{T}_m^{(2)}$ defines the quadrature rule $\tilde{G}_m^{(2)}$. Analogously as above, we may carry out only $m + 1$ steps of Algorithm 1 and use an estimate $\tilde{\alpha}_m$ of $\tilde{\alpha}_m$. We denote the matrix obtained by replacing $\tilde{\alpha}_m$ by $\tilde{\alpha}_m$ in $\tilde{T}_m^{(2)}$ by $\tilde{T}_m^{(2)}$. For instance, we may let $\tilde{\alpha}_m := \tilde{\alpha}_m$. The matrix $\tilde{T}_m^{(2)}$ defines the quadrature rule $\tilde{G}_m^{(2)}$. We have

$$ \tilde{G}_m^{(2)} f = \tilde{G}_m^{(2)} f \quad \forall f \in \mathbb{P}^{2m+2}; $$

see Theorem 30 below for a proof.

We define the average rule

$$ \tilde{A}_m^{(2)} := \frac{1}{2}(G_m + \tilde{G}_m^{(2)}) $$

and obtain analogously to (3.33) that

$$ \tilde{A}_m^{(2)} f = I f \quad \forall f \in \mathbb{P}^{2m+2}. $$

We turn to $\ell = 3$, and provide expressions for the entries in the last row of the matrix $\tilde{T}_m^{(3)}$, assuming that division by zero does not prevent these expressions from being evaluated. Analogous formulas for a nonsymmetric tridiagonal matrix that is analogous to $\tilde{T}_m^{(3)}$ and is associated with monic orthogonal polynomials are derived in [63]. We therefore omit the proof.

**Theorem 23.** Assume that $\tilde{\beta}_m^{(3)} \tilde{\beta}_m^{(3)} \neq 0$. Then the nontrivial entries in the last row of the matrix $\tilde{T}_m^{(3)}$ defined by (4.9) satisfy

$$ \tilde{\beta}_m^{(3)} = \frac{\beta_m^{(3)}(\beta_m^{(3)} - \beta_m^{(3)}) - \beta_m^{(3)} \beta_m^{(3)}(\alpha_m - \alpha_m)^2}{\beta_m^{(3)} \beta_m^{(3)}} $$
and

\[
\tilde{\alpha}_{m+3} = \frac{\beta_{m+2}^2 \beta_{m+1}^2 (\alpha_{m+3} + 2\alpha_{m+2} - 2\tilde{\alpha}_{m+2}) + \beta_{m}^2 \beta_{m-1}^2 (2\tilde{\alpha}_{m+2} - 2\alpha_{m} - \alpha_{m-1})}{\beta_{m+2}^2 \beta_{m+1}^2} + \frac{\alpha_{m+2}^2 \beta_{m+1}^2 (\alpha_{m+2} - \tilde{\alpha}_{m+2})^2 - \alpha_{m}^2 \beta_{m}^2 (\alpha_{m} - \tilde{\alpha}_{m+2})^2}{\beta_{m+2}^2 \beta_{m+1}^2}.
\]

We remark that if the right-hand side of (4.15) is positive, then we let \(\tilde{\beta}_{m+2}^{(2)}\) be the square root of this expression. If, instead, the right-hand side of (4.15) is negative, then we may choose \(\tilde{T}_{m+3}^{(3)}\) to be nonsymmetric with the last superdiagonal entry one and the last subdiagonal entry equal to the right-hand side of (4.15).

The matrix \(\tilde{T}_{m+3}^{(3)}\) determines the quadrature rule \(\tilde{G}_{m+3}^{(3)}\). The computation of the entries of \(\tilde{T}_{m+3}^{(3)}\) requires \(m + 3\) steps with Algorithm 1, which demands the evaluation of \(m + 3\) matrix-vector products with the matrix \(A\). Similarly as above, we may, in order to reduce the number of matrix-vector product evaluations with \(A\) by one, replace the entry \(\tilde{\alpha}_{m+3}\) by an estimate \(\hat{\alpha}_{m+3}\). For instance, we may use \(\tilde{\alpha}_{m+2}\) as an estimate of \(\tilde{\alpha}_{m+3}\). We denote the symmetric tridiagonal matrix that is obtained by replacing the entry \(\tilde{\alpha}_{m+3}\) of \(\tilde{T}_{m+3}^{(3)}\) by an estimate \(\hat{\alpha}_{m+3}\) by \(\tilde{T}_{m+3}^{(3)}\), and denote the quadrature rule associated with the latter matrix by \(\tilde{G}_{m+3}^{(3)}\). We remark that \(\tilde{\alpha}_{m+3}\) may be replaced by some \(\hat{\alpha}_{m+3} \in \mathbb{R}\) also when the product \(\tilde{\beta}_{m+2} \tilde{\alpha}_{m+1}\) vanishes. Thus, the quadrature rule \(\tilde{G}_{m+3}^{(3)}\) can be defined also when the rule \(\tilde{G}_{m+3}^{(3)}\) does not exist. The properties of the rule \(\tilde{G}_{m+3}^{(3)}\) are analogous to those of \(\tilde{G}_{m+2}^{(2)}\); see Theorem 30 below for details for the more general situation when the measure \(dw\) is matrix-valued. An average rule \(\tilde{A}_{m+3}^{(3)}\) can be defined similarly as \(\tilde{A}_{m+2}^{(2)}\); cf. (4.11).

4.1.2 Real-valued Functionals with a Nonsymmetric Matrix

We extend the discussion of the previous section to matrix functionals with a large, possibly nonsymmetric, matrix \(A \in \mathbb{R}^{n \times n}\), and vectors \(V, W \in \mathbb{R}^n\) such that \(V^TW = 1\). We recall the discussion of Section 3.2 for Gauss and anti-Gauss quadrature rules with respect to the measure \(d\nu\). Where there are two families of biorthonormal polynomials \(\{\tilde{p}_j\}_{j=0,1,2,...}\).
and \( \{ \tilde{q}_i \}_{i=0,1,2,...} \) associated with this bilinear form. Thus,

\[
\langle \tilde{p}_j, \tilde{q}_i \rangle = \delta_{j,i}.
\]

Assuming that the polynomials \( \tilde{p}_0, \tilde{p}_1, \ldots, \tilde{p}_{m+\ell} \) and \( \tilde{q}_0, \tilde{q}_1, \ldots, \tilde{q}_{m+\ell} \) exist, they satisfy three-term recursion relations, that can be written as

\[
x [\tilde{p}_0(x), \tilde{p}_1(x), \ldots, \tilde{p}_{m+\ell-1}(x)] = [\tilde{p}_0(x), \tilde{p}_1(x), \ldots, \tilde{p}_{m+\ell-1}(x)] \tilde{T}^{(\ell)}_{m+\ell}
\]

\[
+ \tilde{\beta}_{m+\ell} [0, \ldots, 0, \tilde{p}_{m+\ell}(x)],
\]

\[
x [\tilde{q}_0(x), \tilde{q}_1(x), \ldots, \tilde{q}_{m+\ell-1}(x)] = [\tilde{q}_0(x), \tilde{q}_1(x), \ldots, \tilde{q}_{m+k-1}(x)] (\tilde{T}^{(\ell)}_{m+\ell})^T
\]

\[
+ \tilde{\beta}'_{m+\ell} [0, \ldots, 0, \tilde{q}_{m+\ell}(z)],
\]

where

\[
\tilde{T}^{(\ell)}_{m+\ell} = \begin{bmatrix}
\tilde{\alpha}_1 & \tilde{\beta}'_1 & 0 \\
\tilde{\beta}'_1 & \tilde{\alpha}_2 & \tilde{\beta}'_2 \\
\tilde{\beta}'_2 & \tilde{\alpha}_3 & \ddots \\
& \ddots & \ddots & \ddots & \ddots \tilde{\beta}'_{m-2} \\
& & \ddots & \ddots & \ddots & \ddots \tilde{\beta}'_{m-1} \\
& & & \ddots & \ddots & \ddots \tilde{\beta}'_{m+\ell-1} \\
0 & \ldots & \ldots & \ldots & \ldots & \tilde{\beta}_{m+\ell} & \tilde{\alpha}_{m+\ell}
\end{bmatrix}
\]

is a tridiagonal matrix of order \( m + \ell \).

We turn to the computation of the generalized anti-Gauss rule

\[
\tilde{G}^{(\ell)}_{m+\ell} f = e_1^T f (\tilde{T}^{(\ell)}_{m+\ell}) e_1
\]

determined by the matrix (4.17). Thus, we discuss the evaluation of the entries of this matrix. The rule (4.18) satisfies (4.6) with \( \mathcal{I} \) given by (3.5) and the \( m \)-point Gauss rule \( \mathcal{G}_m \) defined by (3.6). We first consider the entries of the matrix (4.17) for the situation when
\( \ell = 1 \). It follow from section 3.2 but for more convenience of the reader and to use it in the coming proof we use (3.7) and (4.8) to show that

\[ \tilde{\alpha}_j = \alpha_j, \quad j = 1, \ldots, m, \]

and

\[ \tilde{\beta}_j = \beta_j, \quad \tilde{\beta}'_j = \beta'_j, \quad j = 1, \ldots, m - 1. \]

Therefore,

\[ \tilde{p}_j = p_j, \quad \tilde{q}_j = q_j, \quad j = 1, \ldots, m - 1, \]

and

\[ \tilde{\alpha}_{m+1} = \alpha_{m+1}, \quad \tilde{\beta}_m = \sqrt{2} \beta_m, \quad \tilde{\beta}'_m = \sqrt{2} \beta'_m, \]

\[ \tilde{p}_m = \frac{1}{\sqrt{2}} p_m, \quad \tilde{q}_m = \frac{1}{\sqrt{2}} q_m. \]

We obtain

\[ \tilde{p}_{m+1} = \frac{1}{\sqrt{2} \beta_{m+1}} (\beta_{m+1} p_{m+1} - \beta'_m p_{m-1}), \]

\[ \tilde{q}_{m+1} = \frac{1}{\sqrt{2} \beta'_m} (\beta'_m q_{m+1} - \beta_m q_{m-1}); \]  

(4.19)

see [23] for a proof. All entries of the matrix \( \tilde{T}^{(1)}_{m+1} \) can be determined by carrying out \( m + 1 \) steps with Algorithm 2. Alternatively, we carry out only \( m \) steps and estimate the entry \( \tilde{\alpha}_{m+1} \); for instance, we may set this entry to \( \tilde{\alpha}_m \). The latter approach is discussed and illustrated in [2].

We turn to the situation when \( \ell = 2 \). For every \( \ell \geq 1 \), the leading \( (m+\ell-1) \times (m+\ell-1) \) principal submatrix of \( \tilde{T}^{(\ell)}_{m+\ell} \) is \( \tilde{T}^{(\ell-1)}_{m+\ell-1} \). Assuming that the matrix \( \tilde{T}^{(1)}_{m+1} \) is known, we only have to determine expressions for the entries \( \tilde{\alpha}_{m+2}, \, \tilde{\beta}_{m+1}, \) and \( \tilde{\beta}'_{m+1} \) in the last row and column of \( \tilde{T}^{(2)}_{m+2} \), if they exist. It is convenient to sometimes write \( \mathcal{I}(f,g) = \mathcal{I}(f,g) \) and \( \mathcal{G}_m(f,g) = \mathcal{G}_m(f,g) \) for \( \mathcal{I} \) defined by (3.5) and \( \mathcal{G}_m \) by (3.6).
Theorem 24. Assume that $\tilde{\beta}_{m+1}\tilde{\beta}_{m+1}' \neq 0$. Then the nontrivial entries in the last row and column of the matrix $\tilde{T}^{(2)}_{m+2}$ defined by (4.17) satisfy

\begin{align}
\tilde{\alpha}_{m+2} &= \frac{\alpha_{m+2}\beta_{m+1}'\beta_{m+1} - \alpha_m\beta_m\beta_m'}{\beta_{m+1}'\beta_{m+1}}, \\
\tilde{\beta}_{m+1}\tilde{\beta}_{m+1}' &= \beta_{m+1}'\beta_{m+1} - \beta_m\beta_m'.
\end{align}

Proof. We first show (4.21). The relation (4.19) and the fact that $I(p_j, q) = 0$ for $q \in \mathbb{P}^{j-1}$ yield

\begin{align}
2I(x\tilde{p}_m, \tilde{q}_{m+1}) &= 2I\left(\frac{x}{\sqrt{2}}p_m, \frac{1}{\sqrt{2}\beta_{m+1}'}(\beta_{m+1}'q_{m+1} - \beta_mq_{m-1})\right) \\
&= \frac{1}{\beta_{m+1}'}I(xp_m, \beta_{m+1}'q_{m+1} - \beta_mq_{m-1}) - I(xp_m, \beta_mq_{m-1}) \\
&= \frac{1}{\beta_{m+1}'}(\beta_{m+1}'I(xp_m, q_{m+1}) - \beta_mI(xp_m, q_{m-1})) \\
&= \frac{1}{\beta_{m+1}'}(\beta_{m+1}'\beta_{m+1}' - \beta_m\beta_m').
\end{align}

Similarly, the three-term recurrence relations (4.16), property (3.7), and the observation that $G_m(p_m, q) = 0$ for all polynomials $q$, yield

\begin{align}
G_m(x\tilde{p}_m, \tilde{q}_{m+1}) &= G_m\left(\frac{x}{\sqrt{2}}p_m, \frac{1}{\sqrt{2}\beta_{m+1}'}(\beta_{m+1}'q_{m+1} - \beta_mq_{m-1})\right) \\
&= \frac{1}{2}G_m\left(xp_m, \frac{1}{\beta_{m+1}'}(\beta_{m+1}'q_{m+1} - \beta_mq_{m-1})\right) = 0.
\end{align}

The relations (4.22) and (4.23) now give

$$\tilde{\beta}_{m+1} = (2I - G_m)(x\tilde{p}_m, \tilde{q}_{m+1}) = \frac{1}{\beta_{m+1}'}(\beta_{m+1}'\beta_{m+1}' - \beta_m\beta_m').$$

This shows (4.21).
We turn to (4.20). Analogously as above, we obtain

\[
2\mathcal{I}(x\bar{p}_{m+1}, \bar{q}_{m+1})
\]

\[
= 2\mathcal{I}
\left(\frac{x}{\sqrt{2}m_{m+1}} (\beta_{m+1}x - \beta_{m+1}'m p_{m-1}), \frac{1}{\sqrt{2}m_{m+1}} (\beta_{m+1}q_{m+1} - \beta_{m+1}q_{m-1})\right)
\]

\[
= \frac{1}{\beta_{m+1}\beta_{m+1}'} \mathcal{I}
\left(x(\beta_{m+1}x - \beta_{m+1}'m p_{m-1}), (\beta_{m+1}q_{m+1} - \beta_{m+1}q_{m-1})\right)
\]

\[
= \frac{1}{\beta_{m+1}\beta_{m+1}'} \left(\beta_{m+1}'\beta_{m+1} x p_{m+1} - \beta_{m+1}'m q_{m+1}, \beta_{m+1}x p_{m+1} + \beta_{m+1}'m q_{m+1}\right)
\]

\[
= \frac{1}{\beta_{m+1}\beta_{m+1}'} \left(\beta_{m+1}'\beta_{m+1} x p_{m+1} + \beta_{m+1}'m q_{m+1}\right)
\]

and

\[
\mathcal{G}_{m}(x\bar{p}_{m+1}, \bar{q}_{m+1}) = \mathcal{G}_{m}
\left(\frac{x}{\beta_{m+1}} ((x - \alpha_{m+1})x - \beta_{m+1}'m p_{m-1}), \frac{1}{\beta_{m+1}} ((x - \alpha_{m+1})x - \beta_{m+1}'m q_{m-1})\right)
\]

\[
= \frac{\mathcal{G}_{m}
\left(x(\beta_{m+1}x - \beta_{m+1}'m p_{m-1}), (\beta_{m+1}q_{m+1} - \beta_{m+1}q_{m-1})\right)}{\beta_{m+1}\beta_{m+1}'}
\]

\[
= \frac{\mathcal{G}_{m}(x(-\sqrt{2}m_{m+1}'x p_{m-1}), -\sqrt{2}m_{m+1}'x q_{m-1})}{\beta_{m+1}\beta_{m+1}'}
\]

\[
= \frac{1}{\beta_{m+1}\beta_{m+1}'} \left(2\beta_{m+1}'\beta_{m+1} x p_{m+1}, \beta_{m+1}x p_{m+1} + \beta_{m+1}'m q_{m+1}\right)
\]

\[
= \frac{1}{\beta_{m+1}\beta_{m+1}'} \left(2\beta_{m+1}'\beta_{m+1} x p_{m+1}\right)
\]

The above relations yield

\[
\tilde{\alpha}_{m+2} = (2\mathcal{I} - \mathcal{G}_{m})(x\bar{p}_{m+1}, \bar{q}_{m+1})
\]

\[
= \frac{1}{\beta_{m+1}\beta_{m+1}'} \left(\beta_{m+1}'\beta_{m+1} x p_{m+1} + \beta_{m+1}'m q_{m+1} - 2\beta_{m+1}'\beta_{m+1} x p_{m+1}\right)
\]

and the right-hand side simplifies to (4.20).

The evaluation of the entries of the matrix \(\mathcal{T}_{m+2}^{(2)}\) requires that \(m+2\) steps with Algorithm 2 be carried out. If we estimate the value of the entry \(\tilde{\alpha}_{m+2}\), then it suffices to carry out \(m+1\) steps. Moreover, the condition \(\beta_{m+1}\beta_{m+1}' \neq 0\) may be violated, in which case \(\tilde{\alpha}_{m+2}\)
is not defined. It then is meaningful to replace \( \tilde{\alpha}_{m+2} \) in \( \tilde{T}_{m+2}^{(2)} \) by a well-defined entry \( \hat{\alpha}_{m+2} \).

We refer to the matrix so defined as \( \hat{T}_{m+2}^{(2)} \) and to the quadrature rule associated with \( \hat{T}_{m+2}^{(2)} \) as \( \hat{G}_{m+2}^{(2)} \). This rule is said to be a simplified generalized anti-Gauss rule. In computations, we choose the last subdiagonal and superdiagonal entries of \( \tilde{T}_{m+2}^{(2)} \) and \( \hat{T}_{m+2}^{(2)} \) to be \( \tilde{\beta}_{m+1} \) and unity, respectively. This choice can be justified by a result analogous to Proposition 22.

Now let \( \ell = 3 \). We assume that the matrix \( \tilde{T}_{m+2}^{(2)} \) is known and seek to determine the entries \( \tilde{\beta}_{m+2} \) and \( \tilde{\beta}'_{m+2} \) in the last row and column of the matrix \( \hat{T}_{m+3}^{(3)} \). The last diagonal element of \( \hat{T}_{m+3}^{(3)} \) is set to \( \hat{\alpha}_{m+3} := \tilde{\alpha}_{m+2} \). It is convenient not to use the matrix \( \tilde{T}_{m+3}^{(3)} \) because its last diagonal entry is tedious to evaluate. The quadrature rule associated with \( \hat{T}_{m+3}^{(3)} \) is denoted by \( \hat{G}_{m+3}^{(3)} \). It satisfies

\[
\hat{G}_{m+3}^{(3)} f = \tilde{G}_{m+3}^{(3)} f \quad \forall f \in \mathbb{P}^{2m+4},
\]

where \( \tilde{G}_{m+3} \) is the quadrature rule determined by the matrix \( \tilde{T}_{m+3}^{(3)} \); see Theorem 30 below.

The following result yields properties of the last off-diagonal entries of the matrices \( \tilde{T}_{m+3}^{(3)} \) and \( \hat{T}_{m+3}^{(3)} \).

**Theorem 25.** Assume that \( \tilde{\beta}_{m+1} \tilde{\beta}'_{m+1} \tilde{\beta}'_{m+2} \neq 0 \). Then the last off-diagonal entries of the matrix \( \hat{T}_{m+3}^{(3)} \) defined by (4.17) satisfy

\[
\tilde{\beta}_{m+2} \tilde{\beta}'_{m+2} = \frac{1}{\beta_{m+1} \beta'_{m+1}} \left( \tilde{\beta}_{m+1} \tilde{\beta}'_{m+1} (\beta_{m+2} \beta_{m+2} \beta_{m+2} \beta'_{m+1} \beta'_{m+1} - \beta_{m} \beta_{m} \beta_{m-1} \beta'_{m-1}) 
- \beta_{m+1} \beta'_{m+1} \beta_{m} \beta'_{m} (\alpha_{m+2} - \alpha_{m})^2 \right).
\]

**Proof.** The last off-diagonal entries of \( \tilde{T}_{m+3}^{(3)} \) are defined by

\[
\tilde{\beta}_{m+2} = (2I - \mathcal{G}_m)(x \tilde{\mathcal{G}}_{m+1}, \bar{q}_{m+2}), \quad \tilde{\beta}'_{m+2} = (2I - \mathcal{G}_m)(x \tilde{\mathcal{G}}_{m+2}, \bar{q}_{m+1}).
\]

Express \( \bar{q}_{m+2} \) in terms of polynomials \( q_j \). We obtain from

\[
\bar{q}_{m+2} = \frac{1}{\tilde{\beta}'_{m+2}} ((x - \tilde{\alpha}_{m+1}) \bar{q}_{m+1} - \tilde{\beta}_{m+1} \bar{q}_m)
\]
By using 4.21 and 4.19 we get

\[
\tilde{q}_{m+2} = \frac{1}{\beta'_{m+2}} ((x - \tilde{\alpha}_{m+1})(\rho_{m+1}^{\prime}q_{m+1} - \beta_m q_{m-1}))
- \frac{1}{\sqrt{2}\beta'_{m+1}}(\beta_{m+1}q_{m+1} - \beta_m \beta'_m) q_m
\]

\[
= \frac{1}{\sqrt{2}\beta'_{m+1}\beta'_{m+2}} ((x - \tilde{\alpha}_{m+1})(\beta'_{m+1}q_{m+1} - \beta_m q_{m-1}) - \beta_{m+1}q_{m+1} + \beta_m \beta'_m q_m)
\]

By substituting the value of \( q_m \) in the last term, we obtain

\[
\tilde{q}_{m+2} = \frac{1}{\sqrt{2}\beta'_{m+1}\beta'_{m+2}} ((x - \alpha_{m+1})(\beta'_{m+1}q_{m+1} - \beta_m q_{m-1}) - \beta_{m+1}q_{m+1} + \beta_m \beta'_m q_{m-1})
+ (x - \alpha_m)q_{m-1} - \beta_m \beta'_m q_{m-2})
\]

\[
= \frac{1}{\sqrt{2}\beta'_{m+1}\beta'_{m+2}} ((x - \alpha_{m+1})\beta'_{m+1}q_{m+1} + (\alpha_{m+1} - \alpha_m)\beta_{m+1} q_{m-1}
- \beta_{m+1}\beta'_{m+1}q_{m-1} - \beta_m \beta'_m q_{m-2})
\]

This expression can be simplified by using

\[
\beta'_{m+1}\beta'_{m+2}q_{m+2} = (x - \alpha_{m+2})\beta'_{m+1}q_{m+1} - \beta'_{m+1}\beta_{m+1} q_{m-1};
\]

This gives

\[
\tilde{q}_{m+2} = \frac{1}{\sqrt{2}\beta'_{m+1}\beta'_{m+2}} ((x - \alpha_{m+1})\beta'_{m+1}q_{m+1} + (\alpha_{m+1} - \alpha_m)\beta_{m+1} q_{m-1}
- \beta'_{m+1}\beta_{m+2} q_{m+2} - (x - \alpha_{m+2})\beta'_{m+1}q_{m+1} - \beta_{m+1} \beta'_m q_{m-2})
\]

Finally we get that

\[
(4.25) \quad \tilde{q}_{m+2} = \frac{1}{\sqrt{2}\beta'_{m+1}\beta'_{m+2}} (\beta'_{m+1}\beta'_{m+2}q_{m+2} + \beta'_{m+1}(\alpha_{m+2} - \tilde{\alpha}_{m+2}) q_{m+1}
+ \beta_{m} (\alpha_{m+2} - \alpha_m) q_{m-1} - \beta_{m} \beta'_m q_{m-2}).
\]

To show (4.24), the relations (4.19), and (4.25) with the fact that \( \mathcal{I}(p_j, q) = 0 \) for \( q \in \mathbb{P}^{j-1} \)
yield

\[ 2\mathcal{I}(x_{\bar{p}m+1}, \bar{q}_{m+2}) \]

\[ = 2\mathcal{I} \left( \frac{x}{\sqrt{2\beta_m+1}} (\beta_{m+1}p_m+1 - \beta_m p_{m-1}), \frac{1}{\sqrt{2\beta_{m+1}+2\beta_{m+1}}} (\beta_{m+1}^\prime \beta_{m+2}^\prime q_{m+2} + \beta_{m+1}^\prime (\alpha_{m+2} - \bar{\alpha}_{m+2}) q_{m+1} + \beta_{m}(\bar{\alpha}_{m+2} - \alpha_m) q_{m-1} - \beta_{m} \beta_{m-1} q_{m-2} \right) \]

\[ = \frac{1}{\beta_m \beta_m \beta_{m+1} \beta_{m+2}} \left( \mathcal{I}(x, \beta_{m+1} p_m+1, \beta_{m+1}^\prime \beta_{m+2}^\prime q_{m+2}) + \mathcal{I}(x, \beta_{m+1} p_{m+1}, \beta_{m+1}^\prime (\alpha_{m+2} - \bar{\alpha}_{m+2}) q_{m+1} + \mathcal{I}(x, \beta_{m}^\prime \beta_{m-1}, \beta_{m}^\prime \beta_{m-1} q_{m-2}) \right) \]

Moreover,

(4.26)

\[ 2\mathcal{I}(x_{\bar{p}m+1}, \bar{q}_{m+2}) = \frac{1}{\beta_m \beta_m \beta_{m+1} \beta_{m+2}} \left( \beta_{m+1}^\prime \beta_{m+2}^\prime \beta_{m+2} + \beta_{m+1}^\prime \beta_{m+2}^\prime (\alpha_{m+2} - \bar{\alpha}_{m+2}) \alpha_{m+2} \right. \]

\[ - \beta_{m}^\prime \beta_{m}(\bar{\alpha}_{m+2} - \alpha_m) \alpha_m + \beta_{m}^\prime \beta_{m} \beta_{m-1}^\prime \beta_{m-1}^\prime \left. \right) \]

and

\[ \mathcal{G}(x_{\bar{p}m+1}, \bar{q}_{m+2}) \]

\[ = \mathcal{G} \left( \frac{x}{\sqrt{2\beta_{m+1}} (\beta_{m+1} p_m+1 - \beta_m p_{m-1}), \frac{1}{\sqrt{2\beta_{m+1} \beta_{m+2}}} (\beta_{m+1}^\prime \beta_{m+2}^\prime q_{m+2} + \beta_{m+1}^\prime (\alpha_{m+2} - \bar{\alpha}_{m+2}) q_{m+1} + \beta_{m}(\bar{\alpha}_{m+2} - \alpha_m) q_{m-1} - \beta_{m} \beta_{m-1} q_{m-2} \right) \]
\[
\begin{align*}
&= \frac{1}{2\beta_{m+1}^2 \beta_1 + \beta_{m+2}^2} G \left( (x - \alpha_{m+1}) xp_m - x \beta_m^p p_{m-1} - x \beta_m^p p_{m-1}, \right. \\
&= \beta_{m+1} (x - \alpha_{m+2}) q_{m+1} - \beta_{m+1} \beta_{m+1} q_m + (\alpha_{m+2} - \bar{\alpha}_{m+2})(x - \alpha_m) q_m \\
&- (\alpha_{m+2} - \bar{\alpha}_{m+2}) \beta_m q_{m-1} + \beta_m (\bar{\alpha}_{m+2} - \alpha_m) q_{m-1} - \beta_m \beta_{m-1} q_{m-2} \\
&= \frac{1}{2\beta_{m+1}^2 \beta_1 + \beta_{m+2}^2} G \left( -2x \beta_m^p p_{m-1}, (x - \alpha_m+1)(x - \alpha_m) q_m - (x - \alpha_m+2) \right. \\
&\beta_m q_{m-1} - (\alpha_{m+2} - \bar{\alpha}_{m+2}) \beta_m q_{m-1} + \beta_m (\bar{\alpha}_{m+2} - \alpha_m) q_{m-1} - \beta_m \beta_{m-1} q_{m-2} \\
&= \frac{1}{2\beta_{m+1}^2 \beta_1 + \beta_{m+2}^2} G \left( -2x \beta_m^p p_{m-1}, -x \beta_m q_{m-1} + 2\bar{\alpha}_{m+2} \beta_m q_{m-1} - \alpha_m \beta_m q_{m-1} - \beta_m \beta_{m-1} q_{m-2} \right)
\end{align*}
\]

and

69
Then by adding (4.26) and (4.27) and using the value of (4.20) we obtain

$$
G_m(x, \beta_{m+1}, \beta_{m+2}, \gamma_{m+1}, \gamma_{m+2}) = \left(\frac{2\beta'_m\beta_m\alpha_m^2 - 2\beta'_m\beta_m\alpha_m + 2\beta'_m\beta_m\beta_m - \beta'_m\beta_m - 1}{\beta_m + 1}\right).
$$

By using the value of (4.21), we get

$$
\beta_m + \beta'_m = \frac{1}{\beta_m + 1}\left(\beta_{m+1}\beta_{m+2} - \beta_m + 1\right)
$$

This complete the proof of this theorem.
Assume that $\tilde{\beta}_{m+2}\tilde{\beta}'_{m+2} \neq 0$. Then, in computations, we choose the last subdiagonal and superdiagonal entries of $\tilde{T}_{m+3}^{(3)}$ to be $\tilde{\beta}_{m+2}\tilde{\beta}'_{m+2}$ and unity, respectively. This choice can be justified by a result analogous to Proposition 22.

We remark that it is possible, but tedious, to compute the entries of the last row and column of the nonsymmetric tridiagonal matrices $\tilde{T}_{m+\ell}^{(\ell)}$ for $\ell \geq 4$.

4.2 Generalized Block Anti-Gauss Rules

This sections generalizes the discussion of Section 4.1 to the situation when $V$ and $W$ in (4.1) and (4.2) are block vectors with block size $1 < k \ll n$.

4.2.1 Generalized Block Anti-Gauss Rules for Functions of a Symmetric Matrix

We consider the approximation of matrix functions of the form (4.1) with a large symmetric matrix $A \in \mathbb{R}^{n \times n}$ when $V \in \mathbb{R}^{n \times k}$ is a block vector with orthonormal columns. We omit the details for the $m$-block Gauss quadrature rules for symmetric functions since it was introduced in Section 3.3.

We turn to generalized block anti-Gauss rules. These rules are block generalizations of the generalized anti-Gauss rules described in Section 4.1. They also generalize the block anti-Gauss rules introduced in [35]. We say that $\tilde{G}_{m+\ell}^{(\ell)}$ is a generalized $(m + \ell)$-block anti-Gauss rule associated with the matrix-valued measure $d\tilde{w}$ in (3.46) if

$$\left( I - \tilde{G}_{m+\ell}^{(\ell)} \right) f = - (I - G_m) f \quad \forall f \in \mathbb{P}^{2m+2\ell-1},$$

where $I$ is defined in (3.46). It is convenient to extend $I$ and $G_m$ to allow functions $f$ affording representation as a series with $k \times k$ matrix coefficients. For later convenience, we define

$$I(f, g) := \sum_{i=1}^{n} f^T(\lambda_i) \tilde{v}_i \tilde{v}_i^T g(\lambda_i),$$

$$G_m(f, g) := \sum_{i=1}^{km} f^T(\theta_i^{(m)}) u_i^{(m)} \left( u_i^{(m)} \right)^T g(\theta_i^{(m)}).$$
The Gauss rule (3.51) can be extended similarly to allow arguments \( f, g \in \mathbb{R}^{k \times k} \), i.e., we may define \( \mathcal{G}_m(f, g) \); see [35] for details.

4.2.1.1 Generalized Block Anti-Gauss Rules, \( \ell = 1 \)

The block anti-Gauss rule \( \tilde{\mathcal{G}}^{(\ell)}_{m+\ell} \) for \( \ell = 1 \) has previously been discussed in [35]. This rule satisfies

\[
(I - \tilde{\mathcal{G}}^{(1)}_{m+1}) f = -(I - \mathcal{G}_m) f \quad \forall f \in \mathbb{P}^{2m+1},
\]

which can be written as

\[
(4.29) \quad \tilde{\mathcal{G}}^{(1)}_{m+1} f = (2I - \mathcal{G}_m) f \quad \forall f \in \mathbb{P}^{2m+1}.
\]

Hence, \( \tilde{\mathcal{G}}^{(1)}_{m+1} \) is the (standard) \((m + 1)\)-block Gauss quadrature rule with respect to the bilinear form defined by the matrix-valued function \( 2I - \mathcal{G}_m \). It follows from (4.29) that the average rule

\[
(4.30) \quad \mathcal{A}_{m+1} := \frac{1}{2} \left( \mathcal{G}_m + \tilde{\mathcal{G}}^{(1)}_{m+1} \right)
\]

is exact for all polynomials in \( \mathbb{P}^{2m+1} \).

The details of the derivation of the block anti-Gauss \( \tilde{\mathcal{G}}^{(1)}_{m+1} \) can be found in [35], which is the case \( \ell = 1 \) as presented in Section 3.2. We can evaluate the block entries of the matrix (3.53) with almost no work from the matrix \( T_{m+1} \) that is associated with the \((m + 1)\) block Gauss quadrature rule analogous to (3.51). It follows from (3.47) that the matrix recursion coefficients \( \Omega_i \) and \( \Gamma_i \) are given by

\[
\Omega_i = \mathcal{I}(p_{i-1}, \lambda p_{i-1}), \quad \Gamma_i = \mathcal{I}(p_i, \lambda p_{i-1}),
\]

and, similarly, we obtain from (3.47) that

\[
\tilde{\Omega}_i = (2I - \mathcal{G}_m)(\tilde{p}_{i-1}, \lambda \tilde{p}_{i-1}) \quad \tilde{\Gamma}_i = (2I - \mathcal{G}_m)(\tilde{p}_i, \lambda \tilde{p}_{i-1}).
\]
These relations can be used to show that

\[ \tilde{\Omega}_i = \Omega_i, \quad 1 \leq i \leq m, \]
\[ \tilde{\Gamma}_i = \Gamma_i, \quad 1 \leq i \leq m - 1, \]
\[ \tilde{p}_i = p_i, \quad 0 \leq i \leq m - 1; \]

Moreover, one can choose

\[ (4.31) \quad \tilde{\Gamma}_m = \sqrt{2}\Gamma_m, \quad \tilde{\Omega}_{m+1} = \Omega_{m+1}; \]

see [35] for details. Again we introduced the block coefficients in this section as well as in Section 3.3 for more convenience of the reader. In conclusion, the matrix \( \tilde{T}^{(1)}_{m+1} \) associated with the \((m + 1)\)-block anti-Gauss rule \( \tilde{G}^{(1)}_{m+1} \) can be determined from the matrix \( T_{m+1} \) associated with the \((m + 1)\)-block Gauss rule \( G_{m+1} \) by multiplying the block entries \( \Gamma_m \) and \( \Gamma^T_m \) of the latter by \( \sqrt{2} \). The quadrature rule \( \tilde{G}^{(1)}_{m+1} \) allows the representation

\[ \tilde{G}^{(1)}_{m+1} f = E_1^T f(\tilde{T}^{(1)}_{m+1}) E_1, \]

which is analogous to (3.51).

4.2.1.2 Generalized Block Anti-Gauss Rules, \( \ell = 2 \)

For every positive integer \( \ell \), the leading \( k(m + \ell - 1) \times k(m + \ell - 1) \) principal submatrix of \( \tilde{T}^{(\ell)}_{m+\ell} \) is \( \tilde{T}^{(\ell-1)}_{m+\ell-1} \). We derive explicit formulas for the coefficients \( \tilde{\Gamma}_{m+1} \) and \( \tilde{\Omega}_{m+2} \) in the last block row of \( \tilde{T}^{(2)}_{m+2} \), if they exist.

**Theorem 26.** Assume that the matrix \( \tilde{\Gamma}_{m+1} \) is nonsingular. Then the nontrivial block entries in the last block row of the matrix \( \tilde{T}^{(2)}_{m+2} \) satisfy

\[ (4.32) \quad \tilde{\Omega}_{m+2} = \tilde{\Gamma}^{-T}_{m+1}(\Gamma^T_{m+1}\Omega_{m+2}\Gamma_{m+1} - \Gamma_{m}\Omega_{m}\Gamma^T_{m})\tilde{\Gamma}^{-1}_{m+1}, \]
\[ (4.33) \quad \tilde{\Gamma}^T_{m+1}\tilde{\Gamma}_{m+1} = \Gamma^T_{m+1}\Gamma_{m+1} - \Gamma_{m}\Gamma^T_{m}. \]
Proof. We first show (4.33). The relation (3.47) with $j = m + 1$ reads

$$\tilde{p}_{m+1}\tilde{\Gamma}_{m+1} = \lambda \tilde{p}_m - \tilde{p}_m \tilde{\Omega}_{m+1} - \tilde{p}_{m-1}\tilde{\Gamma}_m^T,$$

and from (4.31) and $\tilde{p}_m = (1/\sqrt{2}) p_m$, we obtain

$$\tilde{p}_{m+1}\tilde{\Gamma}_{m+1} = \frac{\lambda}{\sqrt{2}} p_m - \frac{1}{\sqrt{2}} p_m \Omega_{m+1} - \sqrt{2} \tilde{p}_{m-1}\tilde{\Gamma}_m^T,$$

which yields

$$\tilde{p}_{m+1} = \frac{1}{\sqrt{2}} (\lambda p_m - p_m \Omega_{m+1} - 2 \tilde{p}_{m-1}\tilde{\Gamma}_m^T)\tilde{\Gamma}_{m+1}^{-1}$$

(4.34)

$$= \frac{1}{\sqrt{2}} (p_{m+1}\Gamma_{m+1} - p_m \Gamma_m)\tilde{\Gamma}_{m+1}^{-1}.$$

Hence,

$$\tilde{\Gamma}_{m+1} = (2I - \mathcal{G}_m) (\tilde{p}_{m+1}, \lambda \tilde{p}_m) = 2I (\tilde{p}_{m+1}, \lambda \tilde{p}_m) - \mathcal{G}_m (\tilde{p}_{m+1}, \lambda \tilde{p}_m),$$

where $\mathcal{G}_m (\tilde{p}_{m+1}, \lambda \tilde{p}_m) = O_k$ due to [35, Theorem 2]. Moreover, it follows from (4.34) that

$$\tilde{\Gamma}_{m+1} = 2I \left( \frac{1}{\sqrt{2}} (p_{m+1}\Gamma_{m+1} - p_m \Gamma_m)\tilde{\Gamma}_{m+1}^{-1}, \frac{1}{\sqrt{2}} \lambda p_m \right)$$

$$= \tilde{\Gamma}_{m+1} (I (p_{m+1}\Gamma_{m+1}, \lambda p_m) - I (p_m \Gamma_m, \lambda p_m)).$$

Therefore,

$$\Gamma_{m+1}^T \tilde{\Gamma}_{m+1} = \Gamma_{m+1}^T (p_{m+1}, \lambda p_m) - \Gamma_m (p_m, \lambda p_m)$$

$$= \Gamma_{m+1}^T - \Gamma_m \Gamma_m^T.$$

We turn to (4.32). By using the relation (4.34) as above, we obtain

$$2I (\tilde{p}_{m+1}, \lambda \tilde{p}_{m+1}) = 2I \left( \frac{1}{\sqrt{2}} (p_{m+1}\Gamma_{m+1} - p_m \Gamma_m^T)\tilde{\Gamma}_{m+1}^{-1}, \frac{\lambda}{\sqrt{2}} (p_{m+1}\Gamma_{m+1} - p_m \Gamma_m^T)\tilde{\Gamma}_{m+1}^{-1} \right)$$

$$= \tilde{\Gamma}_{m+1} (I (p_{m+1}\Gamma_{m+1}, \lambda p_{m+1}) \Gamma_{m+1})$$

$$+ I (p_m \Gamma_m, \lambda p_m \Gamma_m)\tilde{\Gamma}_{m+1}^{-1}$$

(4.35)

$$= \tilde{\Gamma}_{m+1} (\Gamma_{m+1}^T \Omega_{m+2} \Gamma_{m+1} + \Gamma_m \Omega_m \Gamma_m^T)\tilde{\Gamma}_{m+1}^{-1}.$$
and

\[
\mathcal{G}_m (\tilde{p}_{m+1}, \lambda \tilde{p}_{m+1}) = \mathcal{G}_m \left( (\lambda \tilde{p}_m - \tilde{p}_m \tilde{\Omega}_{m+1} - \tilde{p}_m - 1 \tilde{\Gamma}_m^T) \tilde{\Gamma}_{m+1}^{-1}, \right.
\]

\[
\left. \lambda (\lambda \tilde{p}_m - \tilde{p}_m \tilde{\Omega}_{m+1} - \tilde{p}_m - 1 \tilde{\Gamma}_m^T) \tilde{\Gamma}_{m+1}^{-1} \right),
\]

\[
(4.36)
\]

Finally, combining (4.35) and (4.36) yields

\[
\tilde{\Omega}_{m+2} = (2 \mathcal{I} - \mathcal{G}_m)(\tilde{p}_{m+1}, \tilde{p}_{m+1})
\]

\[
= \tilde{\Gamma}_{m+1}^{-T} (\Gamma_m^T \Omega_{m+1} + \Gamma_m \Gamma_m^T - 2 \Gamma_m \Omega_{m} \Gamma_m) \tilde{\Gamma}_{m+1}^{-1}.
\]

The right-hand side simplifies to (4.32).

When the right-hand side of (4.33) is symmetric positive definite, we let the matrix \( \tilde{\Gamma}_{m+1} \) be the a Cholesky factor of the right-hand side; otherwise we choose the last sub- and super-diagonal blocks of \( \tilde{\Gamma}_{m+2}^{(2)} \) to be the right-hand side of (4.33) and the identity, respectively. Different choices of the last sub- and super-diagonal blocks correspond to different similarity transformations of the matrix \( \tilde{\Gamma}_{m+2}^{(2)} \). The quadrature rule \( \tilde{\mathcal{G}}_{m+2}^{(2)} \) can be shown to be invariant under these transformations. This follows similarly as Proposition 22.
4.2.1.3 Generalized Block Anti-Gauss Rules, \( \ell = 3 \)

It is quite tedious to compute the last block entries \( \tilde{\Gamma}_{m+2} \) and \( \tilde{\Omega}_{m+3} \) of the matrix \( \tilde{T}_{m+3}^{(3)} \). We therefore only outline how to determine

\[
\tilde{\Gamma}_{m+2} = (2I - G_m) (\tilde{p}_{m+2}, \lambda \tilde{p}_{m+1})
\]

and propose to use a simplified generalized block anti-Gauss rule defined by a block tridiagonal matrix \( \tilde{T}_{m+3}^{(3)} \), which is obtained by replacing the last diagonal block entry \( \tilde{\Omega}_{m+3} \) of \( \tilde{T}_{m+3}^{(3)} \) by \( \tilde{\Omega}_{m+2} = \tilde{\Omega}_{m+2} \). The quadrature rule \( \tilde{G}_{m+3}^{(3)} \) determined by the matrix \( \tilde{T}_{m+3}^{(3)} \) satisfies

\[
\tilde{G}_{m+3}^{(3)} f = \tilde{G}_{m+3}^{(3)} f \quad \forall f \in \mathbb{P}^{2m+4},
\]

see Theorem 30 below.

**Theorem 27.** Assume that the matrix \( \tilde{\Gamma}_{m+1} \) is nonsingular. Then the last subdiagonal block entry \( \tilde{\Gamma}_{m+2} \) of the matrix \( \tilde{T}_{m+3}^{(3)} \) satisfies

\[
\tilde{\Gamma}_{m+2}^{T} \tilde{\Gamma}_{m+2} = \tilde{\Gamma}_{m+2}^{T} (\Gamma_{m+2}^{T})^{T} \Gamma_{m+2} - \Gamma_{m+1}^{T} \Gamma_{m+1} \Gamma_{m+2}^{T} \Gamma_{m+1}
\]

\[
- (\tilde{\Omega}_{m+2} - \Omega_{m+2}) \Gamma_{m+1}^{T} \Omega_{m+2} \Gamma_{m+1} + (\tilde{\Omega}_{m+2} - \Omega_{m}) \Gamma_{m} \Omega_{m} \Gamma_{m}^{T} \Gamma_{m+1}.
\]

**Proof.** We first express \( \tilde{p}_{m+2} \) in terms of polynomials \( \tilde{p}_j \). The relation

\[
\tilde{p}_{m+2} = (\lambda \tilde{p}_{m+1} - \tilde{p}_{m+1} \tilde{\Omega}_{m+2} - \tilde{p}_{m} \tilde{\Gamma}_{m+1}^{T} \tilde{\Gamma}_{m+2}^{-1}
\]

gives

\[
\tilde{p}_{m+2} = \frac{1}{\sqrt{2}} (p_{m+2} \Gamma_{m+2} \Gamma_{m+1} + p_{m+1} \Gamma_{m+1} (\Omega_{m+2} - \tilde{\Omega}_{m+2})
\]

\[
+ p_{m-1} \Gamma_{m}^{T} (\tilde{\Omega}_{m+2} - \Omega_{m}) - p_{m-2} \Gamma_{m}^{T} \Gamma_{m}^{T} \tilde{\Gamma}_{m+1}^{-1} \tilde{\Gamma}_{m+2}^{-1}.
\]

We obtain in the same manner as in Subsection 4.2.2 that

\[
2I (\tilde{p}_{m+2}, \lambda \tilde{p}_{m+1}) = \tilde{\Gamma}_{m+2}^{T} \tilde{\Gamma}_{m+2} \Gamma_{m+2}^{T} \Gamma_{m+2} \Gamma_{m+1} + \Gamma_{m+1}^{T} \Gamma_{m+1} \Gamma_{m+2} \Gamma_{m+1}
\]

\[
- (\tilde{\Omega}_{m+2} - \Omega_{m+2}) \Gamma_{m+1}^{T} \Omega_{m+2} \Gamma_{m+1} - (\tilde{\Omega}_{m+2} - \Omega_{m}) \Gamma_{m} \Omega_{m} \Gamma_{m}^{T} \Gamma_{m}
\]

\[
+ \Gamma_{m-1} \Gamma_{m}^{T} \Gamma_{m-1} \Gamma_{m}^{T} \tilde{\Gamma}_{m+1}^{-1},
\]

\[
G_m (\tilde{p}_{m+2}, \lambda \tilde{p}_{m+2}) = \tilde{\Gamma}_{m+2}^{T} \tilde{\Gamma}_{m+2} (2(\Omega_{m} - \tilde{\Omega}_{m+2}) \Gamma_{m} \Omega_{m} \Gamma_{m}^{T} + 2 \Gamma_{m-1} \Gamma_{m}^{T} \Gamma_{m-1} \Gamma_{m}^{T} \tilde{\Gamma}_{m+1}^{-1}.
\]
The theorem now follows after some extensive computations. The proof is similar to the proof of Theorem 29.

4.2.2 Generalized Block Anti-Gauss Rules for Functions of a Nonsymmetric Matrix

We extend the discussion of the previous subsection to matrix functions (4.2) with a large, possibly nonsymmetric, matrix $A \in \mathbb{R}^{n \times n}$ and block vectors $V, W \in \mathbb{R}^{n \times k}$ such that $V^T W = I_k$. We omit the details for the $m$-block Gauss quadrature rules for nonsymmetric functions since it was introduced in section 3.4.

4.2.2.1 Generalized Block Anti-Gauss Rules, $\ell = 1$

We seek to determine a matrix-valued $(m + 1)$-block anti-Gauss quadrature rule $\tilde{G}^{(1)}_{m+1}$ such that

$$
\left( I - \tilde{G}^{(1)}_{m+1} \right) f = - \left( I - G_m \right) f, \quad f \in \mathbb{P}^{2m+1}.
$$

This relation implies, analogously to the discussion following (4.29), that $\tilde{G}^{(1)}_{m+1}$ is an $(m+1)$-block Gauss quadrature rule with respect to a bilinear form determined by the matrix-valued function $2I - G_m$. The average rule (4.30) with $G_m$ and $\tilde{G}^{(1)}_{m+1}$ determined by (3.51) and (4.37), respectively, is exact for all $p \in \mathbb{P}^{2m+1}$. A derivation of the, generally nonsymmetric, block tridiagonal matrix (3.64) below associated with the quadrature rule $\tilde{G}^{(1)}_{m+1}$ can be found in [35]. We therefore here only outline the derivation with the purpose of introducing notation that subsequently will be used.

There are sequences of polynomials $\tilde{p}_j$ and $\tilde{q}_j$, $j = 0, 1, \ldots$, that are biorthonormal with respect to a bilinear form determined by the matrix-valued function $2I - G_m$. These polynomials satisfy recursion relations for $j = 1, 2, \ldots$ of the form

$$
\lambda \tilde{p}_{j-1}(\lambda) = \tilde{p}_j(\lambda) \tilde{\Gamma}_j + \tilde{p}_{j-1}(\lambda) \tilde{\Omega}_j + \tilde{p}_{j-2}(\lambda) \tilde{\Delta}_T^{j-1},
$$

$$
\lambda \tilde{q}_{j-1}(\lambda) = \tilde{q}_j(\lambda) \tilde{\Delta}_j + \tilde{q}_{j-1}(\lambda) \tilde{\Omega}_j + \tilde{q}_{j-2}(\lambda) \tilde{\Gamma}_T^{j-1},
$$

$$
\tilde{p}_0(\lambda) := I_k, \quad \tilde{q}_0(\lambda) := I_k, \quad \tilde{p}_{-1}(\lambda) := O_k, \quad \tilde{q}_{-1}(\lambda) := O_k.
$$
The recursion coefficients \( \tilde{\Omega}_j, \tilde{\Gamma}_j, \tilde{\Delta}_T \) are real \( k \times k \) matrices, that define the block tridiagonal matrix \( \tilde{T}^{(1)}_{m+1} \) which is equal to \( \tilde{T}_{m+1} \) (3.64) which determines the \((m+1)\)-block anti-Gauss rule

(4.39)

\[
\tilde{G}^{(1)}_{m+1} f = E_1^T f(\tilde{T}^{(1)}_{m+1}) E_1.
\]

We turn to the computation of the block entries of the matrix (3.64). The recursions (3.59) and (4.38), together with (4.37) and [35, Corollary 8], yield

(4.40)

\[
\begin{align*}
\tilde{\Omega}_i &= \Omega_i, & 1 \leq i \leq m + 1, \\
\tilde{\Gamma}_i &= \Gamma_i, & 1 \leq i \leq m - 1, \\
\tilde{\Delta}_i &= \Delta_i, & 1 \leq i \leq m - 1, \\
\tilde{p}_i &= p_i, & 0 \leq i \leq m - 1, \\
\tilde{q}_i &= q_i, & 0 \leq i \leq m - 1,
\end{align*}
\]

and \( \Delta_T^T \Gamma_m = 2 \Delta_T \Gamma_m \). Moreover, \( \tilde{\Omega}_{m+1} = \Omega_{m+1} \); see [35] for details. We choose

(4.41)

\[
\begin{align*}
\tilde{\Gamma}_m &:= \sqrt{2} \Gamma_m, & \tilde{\Delta}_m &:= \sqrt{2} \Delta_m.
\end{align*}
\]

Thus, similarly as in Subsection 4.2.1.1, the matrix \( \tilde{T}^{(1)}_{m+1} \) given by (3.64) can be determined from the matrix \( T_{m+1} \) associated with the \((m+1)\)-block Gauss rule (4.39) by multiplying the last off-diagonal blocks \( \Gamma_m \) and \( \Delta_m \) by \( \sqrt{2} \).

We note for future reference that the relations (4.40) yield

(4.42)

\[
\begin{align*}
\tilde{p}_m \tilde{\Gamma}_m &= \lambda \tilde{p}_{m-1} - \tilde{p}_{m-1} \tilde{\Omega}_m - \tilde{p}_{m-2} \tilde{\Delta}_T^T_{m-1} \\
&= \lambda p_{m-1} - p_{m-1} \Omega_m - p_{m-2} \Delta_T^T_{m-1} = p_m \Gamma_m, \\
\tilde{q}_m \tilde{\Delta}_m &= \lambda \tilde{q}_{m-1} - \tilde{q}_{m-1} \tilde{\Omega}_m - \tilde{q}_{m-2} \tilde{\Gamma}_T^T_{m-1} \\
&= \lambda q_{m-1} - q_{m-1} \Omega_T^T - q_{m-2} \Gamma_T^T_{m-1} = q_m \Delta_m;
\end{align*}
\]

see [35] for a detailed derivation.

4.2.2.2 Generalized Block Anti-Gauss Rules, \( \ell = 2 \)

We derive explicit formulas for the coefficients \( \tilde{\Gamma}_{m+1}, \tilde{\Delta}_{m+1} \) and \( \tilde{\Omega}_{m+2} \) in the last block row and column of \( \tilde{T}_{m+2}^{(2)} \), if they exist. The matrix \( \tilde{T}_{m+2}^{(2)} \) is defined analogously as (3.64), and has the latter as \( k(m+1) \times k(m+1) \) leading principal submatrix.
Theorem 28. Assume that $\tilde{\Gamma}_{m+1} \tilde{\Delta}_{m+1}$ is nonsingular. Then the nontrivial block entries in the last block row and column of the matrix $\tilde{T}^{(2)}_{m+2}$ satisfy

\begin{align}
\tilde{\Omega}_{m+2} &= \tilde{\Delta}^{-T}_{m+1}(\Delta^T_{m+1}\Omega_{m+2}\Gamma_{m+1} - \Delta_{m+1}^T\Omega_{m}\Delta^T_{m+1})\tilde{\Gamma}_{m+1}^{-1}, \\
\tilde{\Delta}^T_{m+1}\tilde{\Gamma}_{m+1} &= \Delta^T_{m+1}\Gamma_{m+1} - \Delta_{m+1} \Gamma_{m}^T.
\end{align}

Proof. The proof uses the definition (4.28) that allows the evaluation of the function $\mathcal{I}$ in (3.58) with two matrix-valued arguments. Similarly, a modification of the block Gauss rule (3.51) that allows two matrix-valued arguments is used.

We first show (4.44). From (4.42), we have

$$
\tilde{q}_m\tilde{\Delta}_m = \lambda\tilde{q}_{m-1} - \tilde{q}_{m-1}\tilde{\Omega}_m - \tilde{q}_{m-2}\tilde{\Gamma}_{m-1}^T,
$$

and by (4.41) and $\tilde{q}_m = (1/\sqrt{2})q_m$ it follows that

$$
\tilde{q}_{m+1}\tilde{\Delta}_{m+1} = \lambda\sqrt{\frac{1}{2}}q_m - \frac{1}{\sqrt{2}}q_m\Omega_{m+1} - \sqrt{2}q_{m-1}\Gamma_m^T
$$

and

$$
\tilde{q}_{m+1} = \frac{1}{\sqrt{2}}(\lambda q_m - q_m\Omega_{m+1} - 2q_{m-1}\Gamma_m^T)\tilde{\Delta}_{m+1}^{-1}
\quad = \frac{1}{\sqrt{2}}(q_{m+1}\Delta_{m+1} - q_{m-1}\Gamma_m^T)\tilde{\Delta}_{m+1}^{-1}.
$$

Hence,

$$
\tilde{\Gamma}_{m+1} = (2\mathcal{I} - \mathcal{G}_m)(\tilde{q}_{m+1}, \lambda\tilde{p}_m) = 2\mathcal{I}(\tilde{q}_{m+1}, \lambda\tilde{p}_m) - \mathcal{G}_m(\tilde{q}_{m+1}, \lambda\tilde{p}_m),
$$

where $\mathcal{G}_m(\tilde{q}_{m+1}, \lambda\tilde{p}_m) = O_k$ due to [35, Theorem 4]. In view of (4.45), we also have

$$
\tilde{\Gamma}_{m+1} = 2\mathcal{I}\left(\frac{1}{\sqrt{2}}(q_{m+1}\Delta_{m+1} - q_{m-1}\Gamma_m^T)\tilde{\Delta}_{m+1}^{-1}, \frac{1}{\sqrt{2}}\lambda p_m\right)
$$

and

$$
\tilde{\Delta}_{m+1}^T\tilde{\Gamma}_{m+1} = \mathcal{I}(q_{m+1}\Delta_{m+1}, \lambda p_m) - \mathcal{I}(q_{m-1}\Gamma_m^T, \lambda p_m)
\quad = \Delta^T_{m+1}\mathcal{I}(q_{m+1}, \lambda p_m) - \Gamma_m\mathcal{I}(q_{m-1}, \lambda p_m)
\quad = \Delta^T_{m+1}\Gamma_{m+1} - \Gamma_m\Delta^T_{m}.
$$
We turn to \((4.43)\). Using the relation \((4.45)\) as above, we obtain

\[
2 \mathcal{I}(\tilde{q}_{m+1}, \lambda \tilde{p}_{m+1}) = 2 \mathcal{I} \left( \frac{1}{\sqrt{2}} (q_{m+1} \Delta_{m+1} - q_{m-1} \Gamma_{m}^T) \tilde{\Delta}_{m+1}^{-1} \right)
\]

\[
\frac{\lambda}{\sqrt{2}} (p_{m+1} \Gamma_{m+1} - p_{m-1} \Delta_{m}^T) \tilde{\Gamma}_{m+1}^{-1}
\]

\[
= \tilde{\Delta}_{m+1}^{-1} \left[ \mathcal{I}(q_{m+1} \Delta_{m+1}, \lambda p_{m+1} \Gamma_{m+1}) + \mathcal{I}(q_{m-1} \Gamma_{m}^T, \lambda p_{m-1} \Delta_{m}^T) \right] \tilde{\Gamma}_{m+1}^{-1}
\]

\[
= \tilde{\Delta}_{m+1}^{-1} \left[ \Delta_{m+1}^T \mathcal{I}(q_{m+1}, \lambda p_{m+1}) \Gamma_{m+1} + \Gamma_{m} \mathcal{I}(q_{m-1}, \lambda p_{m-1}) \Delta_{m}^T \right] \tilde{\Gamma}_{m+1}^{-1}
\]

\[
= \tilde{\Delta}_{m+1}^{-1} \left[ \Delta_{m+1}^T \Omega_{m+2} \Gamma_{m+1} + \Gamma_{m} \Omega_{m} \Delta_{m}^T \right] \tilde{\Gamma}_{m+1}^{-1}
\]

(4.46)

and

\[
\mathcal{G}_m(\tilde{q}_{m+1}, \lambda \tilde{p}_{m+1}) = \mathcal{G}_m \left( \lambda \tilde{q}_m - \tilde{q}_m \tilde{\Omega}_{m+1} - \tilde{q}_{m-1} \tilde{\Gamma}_m^T \right) \tilde{\Delta}_{m+1}^{-1}
\]

\[
\frac{\lambda}{\sqrt{2}} (p_{m+1} \tilde{\Omega}_{m+1} - p_{m-1} \tilde{\Delta}_m^T) \tilde{\Gamma}_{m+1}^{-1}
\]

\[
= \tilde{\Delta}_{m+1}^{-1} \mathcal{G}_m \left( \lambda \sqrt{2} q_m - \frac{1}{\sqrt{2}} q_m \Omega_{m+1} - \sqrt{2} q_{m-1} \Gamma_{m}^T \right),
\]

\[
\frac{\lambda}{\sqrt{2}} (p_{m+1} \Omega_{m+1} - \sqrt{2} p_{m-1} \Delta_{m}^T) \tilde{\Gamma}_{m+1}^{-1}
\]

\[
= \tilde{\Delta}_{m+1}^{-1} \mathcal{G}_m \left( - \sqrt{2} q_{m-1} \Gamma_{m}^T, \lambda \left( - \sqrt{2} p_{m-1} \Delta_{m}^T \right) \right) \tilde{\Gamma}_{m+1}^{-1}
\]

\[
= \tilde{\Delta}_{m+1}^{-1} \left[ 2 \Gamma_{m} \mathcal{I}(q_{m-1}, \lambda p_{m-1}) \Delta_{m}^T \right] \tilde{\Gamma}_{m+1}^{-1}
\]

(4.47)

It now follows from (4.46) and (4.47) that

\[
\tilde{\Omega}_{m+2} = (2 \mathcal{I} - \mathcal{G}_m)(\tilde{q}_{m+1}, \lambda \tilde{p}_{m+1})
\]

\[
= \tilde{\Delta}_{m+1}^{-1} \left[ \Delta_{m+1}^T \Omega_{m+2} \Gamma_{m+1} + \Gamma_{m} \Omega_{m} \Delta_{m}^T - 2 \Gamma_{m} \Omega_{m} \Delta_{m}^T \right] \tilde{\Gamma}_{m+1}^{-1}
\]

where the right-hand side simplifies to \((4.43)\).

The last off-diagonal block entries of \(\tilde{T}_{m+2}^{(2)}\) can be chosen arbitrarily so that \((4.44)\) holds. The different choices correspond to similarity transformations of \(\tilde{T}_{m+2}^{(2)}\) under which the quadrature rule \(\tilde{\mathcal{G}}_{m+2}^{(2)}\) is invariant. The invariance can be shown analogously as Proposition 22.
4.2.2.3 Generalized Block Anti-Gauss Rules, \( \ell = 3 \)

We derive a relation for the last subdiagonal and superdiagonal block entries \( \tilde{\Gamma}_{m+2} \) and \( \tilde{\Delta}_{m+2} \) of \( \tilde{T}^{(3)}_{m+3} \).

**Theorem 29.** Assume that the matrices \( \tilde{\Gamma}_{m+1} \) and \( \tilde{\Delta}_{m+1} \tilde{\Delta}_{m+2} \) are nonsingular. Then the last subdiagonal block entries \( \tilde{\Gamma}_{m+2} \) and \( \tilde{\Delta}_{m+2} \) of \( \tilde{T}^{(3)}_{m+3} \) satisfy

\[
\tilde{\Delta}_{m+2}^T \tilde{\Gamma}_{m+2} = \tilde{\Delta}_{m+1}^T \tilde{\Delta}_{m+2} \Gamma_{m+2} \Gamma_{m+1} - \Gamma_{m-1} \Gamma_{m} \tilde{\Delta}_{m-1} \tilde{\Delta}_{m} +
\]

\[
(\tilde{\Omega}_{m+2} - \Omega_{m+2})^T \Delta_{m+1} \Omega_{m+2} \Gamma_{m+1} + (\tilde{\Omega}_{m+2} - \Omega_{m})^T \Gamma_{m} \Omega_{m} \Delta_{m} \tilde{\Gamma}_{m+1}^{-1}.
\]

**Proof.** We first express \( \tilde{q}_{m+2} \) in terms of polynomials \( q_j \). It follows from

\[
\tilde{q}_{m+2} = (\lambda \tilde{q}_{m+1} - \tilde{q}_{m+1} \tilde{\Omega}_{m+2} - \tilde{q}_{m} \tilde{\Gamma}_{m+1}) \tilde{\Delta}_{m+2}^{-1}
\]

By using (4.44) and (4.45) we get

\[
\tilde{q}_{m+2} \tilde{\Delta}_{m+2} = \frac{1}{\sqrt{2}} \left( \lambda (q_{m+1} \Delta_{m+1} - q_{m-1} \Gamma_{m}^T) - (q_{m+1} \Delta_{m+1} - q_{m-1} \Gamma_{m}^T) \tilde{\Omega}_{m+2} \tilde{\Delta}_{m+1}^{-1} \right)
\]

\[-\frac{1}{\sqrt{2}} (q_{m} \Delta_{m+1} \Gamma_{m+1} \Gamma_{m}^T - q_{m} \Gamma_{m} \tilde{\Delta}_{m+1}^{-1})
\]

\[= \frac{1}{\sqrt{2}} (\lambda (q_{m+1} \Delta_{m+1} - q_{m-1} \Gamma_{m}^T) - (q_{m+1} \Delta_{m+1} - q_{m-1} \Gamma_{m}^T) \tilde{\Omega}_{m+2} - q_{m} \Delta_{m+1} \Gamma_{m+1} \Gamma_{m}^T \tilde{\Delta}_{m+1}^{-1}.
\]

By substituting the value of \( q_{m} \) which is analogous of (3.15) in the above expression

\[
\tilde{q}_{m+2} \tilde{\Delta}_{m+2} = \frac{1}{\sqrt{2}} \left( \lambda (q_{m+1} \Delta_{m+1} - q_{m-1} \Gamma_{m}^T) - (q_{m+1} \Delta_{m+1} - q_{m-1} \Gamma_{m}^T) \tilde{\Omega}_{m+2} - q_{m} \Delta_{m+1} \Gamma_{m+1} \Gamma_{m}^T \tilde{\Delta}_{m+1}^{-1}.
\]

Since

\[
q_{m+2} \Delta_{m+2} \Delta_{m+1} = (\lambda q_{m+1} \Delta_{m+1} - q_{m+1} \Delta_{m+1} \Omega_{m+2} - q_{m} \Delta_{m+1} \Gamma_{m}^T)\,.
\]
we obtain

\begin{align*}
\bar{q}_{m+2} \Delta_{m+2} &= \frac{1}{\sqrt{2}} (\lambda q_{m+1} \Delta_{m+1} - q_{m+1} \Delta_{m+1} \tilde{\Omega}_{m+2} + q_{m-1} \Gamma_{m}^{T} (\tilde{\Omega}_{m+2} - \Omega_{m}) + q_{m+2} \\
\Delta_{m+2} \Delta_{m+1} - \lambda q_{m+1} \Delta_{m+1} + q_{m+1} \Delta_{m+1} \Omega_{m+2} - q_{m-2} \Gamma_{m-1}^{T} \Gamma_{m}^{T}) \tilde{\Delta}_{m+1}^{-1} \\
= \frac{1}{\sqrt{2}} (q_{m+2} \Delta_{m+2} \Delta_{m+1} + q_{m+1} \Delta_{m+1} (\Omega_{m+2} - \tilde{\Omega}_{m+2}) \\
+ q_{m-1} \Gamma_{m}^{T} (\tilde{\Omega}_{m+2} - \Omega_{m}) - q_{m-2} \Gamma_{m-1}^{T} \Gamma_{m}^{T}) \tilde{\Delta}_{m+1}^{-1},
\end{align*}

and

\begin{align*}
(4.48) \quad \bar{q}_{m+2} &= \frac{1}{\sqrt{2}} \left( q_{m+2} \Delta_{m+2} \Delta_{m+1} - q_{m+1} \Delta_{m+1} (\tilde{\Omega}_{m+2} - \Omega_{m+2}) \\
+ q_{m-1} \Gamma_{m}^{T} (\tilde{\Omega}_{m+2} - \Omega_{m}) - q_{m-2} \Gamma_{m-1}^{T} \Gamma_{m}^{T} \right) \tilde{\Delta}_{m+1}^{-1} \tilde{\Delta}_{m+2}^{-1},
\end{align*}

The relations (4.42) and (4.48), and the fact that \( \mathcal{I}(q_j, p) = O_k \) for \( p \in \mathbb{F}^{j-1} \) yield

\begin{align*}
2\mathcal{I}(\bar{q}_{m+2}, \lambda \bar{\rho}_{m+1}) \\
= 2\mathcal{I} \left( \frac{1}{\sqrt{2}} \left( q_{m+2} \Delta_{m+2} \Delta_{m+1} - q_{m+1} \Delta_{m+1} (\tilde{\Omega}_{m+2} - \Omega_{m+2}) + q_{m-1} \Gamma_{m}^{T} \\
(\tilde{\Omega}_{m+2} - \Omega_{m}) - q_{m-2} \Gamma_{m-1}^{T} \Gamma_{m}^{T} \right) \tilde{\Delta}_{m+1}^{-1} \tilde{\Delta}_{m+2}^{-1} \right), \frac{1}{\sqrt{2}} (\lambda p_{m+1} \Gamma_{m+1} - p_{m-1} \Delta_{m} \Gamma_{m+1}^{-1}) \right) \\
= \tilde{\Delta}_{m+1}^{-1} \tilde{\Delta}_{m+2}^{-1} \left( \mathcal{I}(q_{m+2} \Delta_{m+2} \Delta_{m+1}, \lambda p_{m+1} \Gamma_{m+1}) - \mathcal{I}(q_{m+1} \Delta_{m+1} (\tilde{\Omega}_{m+2} - \Omega_{m+2}) \\
, \lambda p_{m+1} \Gamma_{m+1}) + \mathcal{I}(q_{m-2} \Gamma_{m-1}^{T} \Gamma_{m}^{T}, \lambda p_{m-1} \Delta_{m} \Gamma_{m+1}^{-1}) \right) \Gamma_{m+1}^{-1} \\
= \tilde{\Delta}_{m+1}^{-1} \tilde{\Delta}_{m+2}^{-1} \left( \Delta_{m+2}^{T} \Delta_{m+1} \mathcal{I}(q_{m+2}, \lambda p_{m+1}) \Gamma_{m+1} - (\tilde{\Omega}_{m+2} - \Omega_{m+2}) \Gamma_{m+1}^{T} \mathcal{I}(q_{m} + 1 \\
, \lambda p_{m+1}) \Gamma_{m+1} - (\tilde{\Omega}_{m+2} - \Omega_{m}) \Gamma_{m} \mathcal{I}(q_{m-1}, \lambda p_{m-1}) \Delta_{m} \Gamma_{m+1} + \Gamma_{m+1} \Gamma_{m} \mathcal{I}(q_{m-2}, \lambda p_{m-1}) \Delta_{m} \Gamma_{m+1} \right) \Gamma_{m+1}^{-1} \\
= \tilde{\Delta}_{m+1}^{-1} \tilde{\Delta}_{m+2}^{-1} \left( \Delta_{m+2}^{T} \Delta_{m+1} \Gamma_{m+1}^{T} \mathcal{I}(q_{m+2}, \lambda p_{m+1}) \Gamma_{m+1} - (\tilde{\Omega}_{m+2} - \Omega_{m+2}) \Gamma_{m+1}^{T} \mathcal{I}(q_{m+1} \\
, \lambda p_{m+1}) \Gamma_{m+1} - (\tilde{\Omega}_{m+2} - \Omega_{m}) \Gamma_{m} \mathcal{I}(q_{m-1}, \lambda p_{m-1}) \Delta_{m} \Gamma_{m+1} + \Gamma_{m+1} \Gamma_{m} \mathcal{I}(q_{m-2}, \lambda p_{m-1}) \Delta_{m} \Gamma_{m+1} \right) \Gamma_{m+1}^{-1},
\end{align*}

82
and we obtain

\[
2 \mathcal{I}(\tilde{q}_{m+2}, \lambda \tilde{p}_{m+1}) = \tilde{\Delta}_{m+2}^{-T} \tilde{\Delta}_{m+1}^{-T} \left( \Delta_{m+1}^T \Delta_{m+2}^T \Gamma_{m+2} \Gamma_{m+1} + (\tilde{\Omega}_{m+2} - \Omega_{m+2})^T \Delta_{m+1}^T \right) \tilde{\Gamma}_{m+1}^{-1}
\]

(4.49)

\[
\Omega_{m+2} \Gamma_{m+1} - (\tilde{\Omega}_{m+2} - \Omega_{m})^T \Gamma_{m} \Omega_{m} \Delta_{m}^T + \Gamma_{m-1} \Gamma_{m} \Delta_{m-1}^T \Delta_{m}^T \tilde{\Gamma}_{m+1}^{-1}
\]

and

\[
\mathcal{G}(\tilde{q}_{m+2}, \lambda \tilde{p}_{m+1})
\]

\[
= \mathcal{G} \left( \frac{1}{\sqrt{2}} \left( (q_{m+2} \Delta_{m+2} \Delta_{m+1} - q_{m+1} \Delta_{m+1} \Omega_{m+2} - q_{m} \Delta_{m+1} \Gamma_{m+1}^T \Delta_{m+1} - \lambda q_{m} (\tilde{\Omega}_{m+2} + \Omega_{m+2}))
\right.
\]

\[
+ q_{m} \Omega_{m+1} (\tilde{\Omega}_{m+2} - \Omega_{m+2}) + q_{m-1} \Gamma_{m}^T (\tilde{\Omega}_{m+2} - \Omega_{m+2}) + q_{m-1} \Gamma_{m} \Omega_{m+2} - \Omega_{m})
\]

\[
- q_{m-2} \Gamma_{m-1}^T \Gamma_{m} \left( \tilde{\Delta}_{m+1}^{-1} \tilde{\Delta}_{m+2}^{-1} \frac{\lambda}{\sqrt{2}} \left( \lambda p_{m} - p_{m} \Omega_{m+1} - p_{m-1} \Delta_{m}^T\Delta_{m} - p_{m-1} \Delta_{m}^T \tilde{\Gamma}_{m+1}^{-1} \right)
\]

\[
= \frac{1}{2} \mathcal{G} \left( \left( \lambda q_{m+1} \Delta_{m+1} - q_{m+1} \Delta_{m+1} \Omega_{m+2} + q_{m+1} \Gamma_{m}^T (\tilde{\Omega}_{m+2} - \Omega_{m+2}) + q_{m+1} \Gamma_{m} \right)
\]

\[
- q_{m-2} \Gamma_{m-1}^T \Gamma_{m} \left( \tilde{\Delta}_{m+1}^{-1} \tilde{\Delta}_{m+2}^{-1} \frac{\lambda}{\sqrt{2}} \left( \lambda p_{m} - p_{m} \Omega_{m+1} - p_{m-1} \Delta_{m}^T\Delta_{m} - p_{m-1} \Delta_{m}^T \tilde{\Gamma}_{m+1}^{-1} \right)
\]

\[
= \frac{1}{2} \mathcal{G} \left( \left( \lambda (\lambda q_{m} - q_{m} \Omega_{m+1} - q_{m-1} \Gamma_{m}^T) - (\lambda q_{m} - q_{m} \Omega_{m+1} - q_{m-1} \Gamma_{m}^T) \Omega_{m+2} + q_{m-1} \Gamma_{m} \right)
\]

\[
- q_{m-2} \Gamma_{m-1}^T \Gamma_{m} \left( \tilde{\Delta}_{m+1}^{-1} \tilde{\Delta}_{m+2}^{-1} \frac{\lambda}{\sqrt{2}} \left( \lambda p_{m} - p_{m} \Omega_{m+1} - p_{m-1} \Delta_{m}^T\Delta_{m} - p_{m-1} \Delta_{m}^T \tilde{\Gamma}_{m+1}^{-1} \right)
\]

\[
= \mathcal{G} \left( \left( - \lambda q_{m-1} \Gamma_{m}^T + q_{m-1} \Gamma_{m} \Omega_{m+2} + q_{m-1} \Gamma_{m} \Omega_{m+2} - \Omega_{m+2} + q_{m-1} \Gamma_{m} \right)
\]

\[
- q_{m-2} \Gamma_{m-1}^T \Gamma_{m} \left( \tilde{\Delta}_{m+1}^{-1} \tilde{\Delta}_{m+2}^{-1} \frac{\lambda}{\sqrt{2}} \left( \lambda p_{m} - p_{m} \Omega_{m+1} - p_{m-1} \Delta_{m}^T\Delta_{m} - p_{m-1} \Delta_{m}^T \tilde{\Gamma}_{m+1}^{-1} \right)
\]

By using

\[
q_{m} \Delta_{m} \Gamma_{m}^T = (\lambda q_{m-1} \Gamma_{m}^T - q_{m-1} \Gamma_{m} \Omega_{m} - q_{m-2} \Gamma_{m-1}^T \Gamma_{m})
\]

83
we can simplify
\[ \mathcal{G}(\tilde{q}_{m+2}, \lambda \tilde{p}_{m+1}) \]
\[ = \mathcal{I} \left( 2q_{m-1} \Gamma_m^T (\tilde{\Omega}_{m+2} - \Omega_m) - 2q_{m-2} \Gamma_m^T \Gamma_m^T \right) \Delta_{m+1}^{-1} \Delta_{m+2}^{-1}, \lambda \left( - p_{m-1} \Delta_m^T \right) \Gamma_m^{-1} \]
\[ = \Delta_{m+2}^{-1} \Delta_{m+1}^{-1} \left( 2q_{m-1} \Gamma_m^T (\tilde{\Omega}_{m+2} - \Omega_m), - \lambda p_{m-1} \Delta_m^T \right) - \mathcal{I} \left( 2q_{m-2} \Gamma_m^T \Gamma_m^T, - \lambda p_{m-1} \Delta_m^T \right) \right) \Gamma_m^{-1} \]
\[ = \Delta_{m+2}^{-1} \Delta_{m+1}^{-1} \left( 2(\Omega_m - \tilde{\Omega}_{m+2}) \Gamma_m \Omega_m \Delta_m^T + 2\Gamma_m^{-1} \Gamma_m \Delta_m^T \Delta_{m-1} \right) \Gamma_m^{-1} \]

(4.50) \[ \mathcal{G}_m(q_{m+2}, \lambda \tilde{p}_{m+1}) = \Delta_{m+2}^{-1} \Delta_{m+1}^{-1} \left( 2(\Omega_m - \tilde{\Omega}_{m+2}) \Gamma_m \Omega_m \Delta_m^T + 2\Gamma_m^{-1} \Gamma_m \Delta_m^T \Delta_{m-1} \right) \Gamma_m^{-1}. \]

Then by adding (4.49) and (4.50) we complete the proof of Theorem 29. \hfill \Box

The formula for the last diagonal block, \( \tilde{\Omega}_{m+3} \), of \( \tilde{T}_{m+3}^{(3)} \) is quite complicated. We therefore set this block to \( \tilde{\Omega}_{m+2} \). This defines the block tridiagonal matrix \( \tilde{T}_{m+3}^{(3)} \) and the associated quadrature rule \( \tilde{\mathcal{G}}_{m+3}^{(3)} \). This rule satisfies

(4.51) \[ \tilde{\mathcal{G}}_{m+3}^{(3)} f = \tilde{\mathcal{G}}_{m+3}^{(3)} f \quad \forall f \in \mathbb{P}^{2m+4}. \]

We conclude this section with a proof of (4.51). Let \( \mathcal{G}_m f \) denote the \( m \)-block Gauss rule (3.51) associated with the operator \( \mathcal{I} \) defined by (3.58). This quadrature rule can be expressed with the, generally nonsymmetric, block tridiagonal matrix \( T_m \), which is defined by (3.60); cf. (3.51). Consider the associated \( (m + \ell) \)-block generalized anti-Gauss rule \( \tilde{\mathcal{G}}_{m+\ell}^{(\ell)} f \).
It is defined by a block tridiagonal matrix

\[
\tilde{T}_{m+\ell}^{(\ell)} = \begin{bmatrix}
\tilde{\Omega}_1 & \tilde{\Delta}_1^T & & \\
\tilde{\Gamma}_1 & \tilde{\Omega}_2 & \tilde{\Delta}_2^T & \\
& \ddots & \ddots & \ddots \\
& & \tilde{\Gamma}_{m-1} & \tilde{\Omega}_m & \tilde{\Delta}_{m+\ell-1}^T \\
& & & \tilde{\Gamma}_{m+\ell-1} & \tilde{\Omega}_{m+\ell}
\end{bmatrix} \in \mathbb{R}^{k(m+\ell)\times k(m+\ell)},
\]

i.e.,

\[
\tilde{G}_{m+\ell}^{(\ell)} f = E_1^T f(\tilde{T}_{m+\ell}^{(\ell)}) E_1.
\]

The matrix (4.52) is a block analogue of (4.9). We assume that \( m \) and \( \ell \) are small enough so that the matrix (4.52) exists. Its leading \( km \times km \) principal submatrix is the block tridiagonal matrix \( T_m \) defined by (3.60). The other blocks of \( \tilde{T}_{m+\ell}^{(\ell)} \) are determined so that the quadrature rule \( \tilde{G}_{m+\ell}^{(\ell)} f \) satisfies

\[
(I - \tilde{G}_{m+\ell}^{(\ell)}) f = -(I - G_m) f \quad \forall f \in P_{2m+2\ell-1}.
\]

This is the block analogue of equation (4.6).

When the last diagonal block entry \( \tilde{\Omega}_{m+\ell} \in \mathbb{R}^{k\times k} \) of (4.52) is replaced by another matrix \( \hat{\Omega}_{m+\ell} \in \mathbb{R}^{k\times k} \), we obtain the block tridiagonal matrix \( \hat{T}_{m+\ell}^{(\ell)} \) and the associated quadrature rule

\[
\hat{G}_{m+\ell}^{(\ell)} f = E_1^T f(\hat{T}_{m+\ell}^{(\ell)}) E_1.
\]

The following theorem collects some results for this quadrature rule, which is also analogous to the Theorem 7.

**Theorem 30.** The quadrature rule (4.54) satisfies

\[
\hat{G}_{m+\ell}^{(\ell)} f = \mathcal{I} f \quad \forall f \in \mathbb{P}^{2m-1},
\]

(4.55)

\[
\hat{G}_{m+\ell}^{(\ell)} f = \hat{G}_{m+\ell}^{(\ell)} f \quad \forall f \in \mathbb{P}^{2m+2\ell-2}.
\]

(4.56)
Proof. The nonsymmetric Lanczos process (Algorithm 4) generates the block entries of the matrix (3.60) in the order \( \Omega_1, \Gamma_1, \Delta_1, \Omega_2, \Gamma_2, \Delta_2, \Omega_3, \ldots \). Each diagonal block \( \Omega_j \) and each pair of off-diagonal blocks \( \{ \Gamma_j, \Delta_j \} \) in the matrix (3.60) increase the degree of the polynomials that are integrated exactly by the quadrature rule (3.51) by one. The matrix (3.60) has \( m \) diagonal blocks and \( m - 1 \) pairs of off-diagonal blocks and, therefore, the rule (3.51) is exact for all polynomials in \( \mathbb{P}^{2m-1} \). If we would replace the diagonal block \( \Omega_m \) by an arbitrary \( k \times k \) matrix, then the quadrature rule defined by the matrix so obtained would be exact for all polynomials in \( \mathbb{P}^{2m-2} \).

We turn to (4.55). Since the matrix \( \tilde{T}^{(\ell)}_{m+\ell} \) associated with the rule \( \tilde{G}^{(\ell)}_{m+\ell} \) is block tridiagonal and has the matrix (3.60) as its \( km \times km \) leading principal submatrix, it integrates all polynomials exactly that the rule (3.51) integrates exactly. This shows (4.55).

Consider the rule \( \tilde{G}^{(\ell)}_{m+\ell} \) exact. It is defined by the block tridiagonal matrix \( \tilde{T}^{(\ell)}_{m+\ell} \). The rule \( \tilde{G}^{(\ell)}_{m+\ell} \) is associated with a block tridiagonal matrix \( \tilde{T}^{(\ell)}_{m+\ell} \), which differs from \( \tilde{T}^{(\ell)}_{m+\ell} \) only in the last diagonal block. Thus, the matrices \( \tilde{T}^{(\ell)}_{m+\ell} \) and \( \tilde{T}^{(\ell)}_{m+\ell} \) share the first \( m + \ell - 1 \) diagonal block entries and all \( m + \ell - 1 \) pairs of off-diagonal block entries. They therefore integrate all polynomials in \( \mathbb{P}^{2m+2\ell-2} \) in the same way. This shows (4.56). \( \square \)

**Corollary 31.** Let the block quadrature rules \( G_m \) and \( \tilde{G}^{(\ell)}_{m+\ell} \) be defined by (3.51) and (4.54), respectively. Then the average rule

\[
\tilde{A}^{(\ell)}_{m+\ell} := \frac{1}{2}(G_m + \tilde{G}^{(\ell)}_{m+\ell})
\]

satisfies

\[
\tilde{A}^{(\ell)}_{m+\ell} f = \mathcal{I} f \quad \forall f \in \mathbb{P}^{2m+2\ell-2}.
\]

Hence, the average rule is exact for polynomials of higher degree than the Gauss and simplified generalized anti-Gauss rules that determine it.
Proof. The rule $\tilde{G}^{(\ell)}_{m+\ell}$ is constructed to satisfy

$$\tilde{G}^{(\ell)}_{m+\ell} f = (2I - G_m) f \quad \forall f \in \mathbb{P}^{2m+2\ell-1}.$$ 

Since the block tridiagonal matrices that are associated with the block quadrature rules $\tilde{G}^{(\ell)}_{m+\ell}$ and $\hat{G}^{(\ell)}_{m+\ell}$ only differ in the last block diagonal entry, we obtain similarly as in the proof of Theorem 30 that

$$\hat{G}^{(\ell)}_{m+\ell} f = (2I - G_m) f \quad \forall f \in \mathbb{P}^{2m+2\ell-2}.$$ 

Substitution into (4.57) shows (4.58).

We remark that it is straightforward to show an analogue of Corollary 31 for the average rule $\tilde{A}^{(\ell)}_{m+\ell} := \frac{1}{2}(G_m + \tilde{G}^{(\ell)}_{m+\ell})$. We therefore omit the details.

4.3 Bracketing with Gauss and Generalized Anti-Gauss Rules

Let $\{s_j\}_{j=0}^\infty$ be a family of polynomials with each polynomial $s_j$ of exactly degree $j$, and such that the expansion

$$(4.59) \quad f(x) = \sum_{j=0}^{\infty} \eta_j s_j(x)$$

converges on the convex hull of the support of the measure $d\bar{w}$ defined in (3.58). We also would like the $s_j$ to be such that the terms $\eta_j s_j$ converge to zero. For instance, when the convex hull of the support of the measure is an interval, then the $s_j$ can be chosen to be Chebyshev polynomials for this interval. Alternatively, we may choose the $s_j$ to be Chebyshev or Faber polynomials for the field of values of $A$ or for some $\varepsilon$-pseudospectrum of $A$; see Trefethen and Embree [69] for discussions on the latter. We obtain

$$(I - G_m) f = \sum_{j=0}^{\infty} \eta_j (I - G_m)s_j = \sum_{j=2m}^{\infty} \eta_j (I - G_m)s_j = \eta_{2m}(I - G_m)s_{2m} + \ldots$$

$$(4.60) \quad + \eta_{2m+2\ell-2}(I - G_m)s_{2m+2\ell-2} + \sum_{j=2m+2\ell-1}^{\infty} \eta_j (I - G_m)s_j,$$
where we have used (3.7). Similarly,

\[
(I - \hat{G}_{m+\ell}^{(f)}) f = \sum_{j=0}^{\infty} \eta_j (I - \hat{G}_{m+\ell}^{(f)}) s_j = \sum_{j=2m}^{\infty} \eta_j (I - \hat{G}_{m+\ell}^{(f)}) s_j \\
= \eta_{2m} (I - \hat{G}_{m+\ell}^{(f)}) s_{2m} + \ldots + \eta_{2m+2\ell-2} (I - \hat{G}_{m+\ell}^{(f)}) s_{2m+2\ell-2} \\
+ \sum_{j=2m+2\ell-1}^{\infty} \eta_j (I - \hat{G}_{m+\ell}^{(f)}) s_j \\
= -\eta_{2m} (I - G_m) s_{2m} - \ldots - \eta_{2m+2\ell-2} (I - G_m) s_{2m+2\ell-2} \\
(4.61) + \sum_{j=2m+2\ell-1}^{\infty} \eta_j (I - \hat{G}_{m+\ell}^{(f)}) s_j.
\]

Here we have applied (4.55), (4.53), and (4.56) in order.

If the terms \(\eta_j (I - G_m) s_j\) decay in norm sufficiently rapidly with increasing \(j\), then the right-hand sides of

\[
(I - G_m) f \approx \eta_{2m} (I - G_m) s_{2m} + \ldots + \eta_{2m+2\ell-2} (I - G_m) s_{2m+2\ell-2}, \\
(I - \hat{G}_{m+\ell}^{(f)}) f \approx -\eta_{2m} (I - G_m) s_{2m} - \ldots - \eta_{2m+2\ell-2} (I - G_m) s_{2m+2\ell-2}
\]

are accurate approximations of the left-hand sides. Hence,

\[
G_m f \approx I f - \eta_{2m} (I - G_m) s_{2m} - \ldots - \eta_{2m+2\ell-2} (I - G_m) s_{2m+2\ell-2}, \\
\hat{G}_{m+\ell}^{(f)} f \approx I f + \eta_{2m} (I - G_m) s_{2m} + \ldots + \eta_{2m+2\ell-2} (I - G_m) s_{2m+2\ell-2}.
\]

This suggests that the component-wise errors of the quadrature rules \(G_m f\) and \(\hat{G}_{m+\ell}^{(f)} f\) are roughly equal in magnitude and of opposite sign, and then the components of \(G_m f\) and \(\hat{G}_{m+\ell}^{(f)} f\) bracket the components of \(I f\). The magnitudes of the entries of \(\eta_j s_j\) decay quickly to zero when \(j\) increases if \(f\) is analytic in a large simply connected region in the complex plane that contains the support of the measure \(d\tilde{w}\) and has its boundary far away from the support. The following theorem provides sufficient conditions for \(G_m f\) and \(\hat{G}_{m+\ell}^{(f)} f\) to bracket \(I f\).
Theorem 32. Consider the expansion (4.59) in terms of the polynomials $s_j$ and assume that for some $1 \leq q, r \leq k,$

$$
\left| \begin{bmatrix}
\sum_{j=2m}^{2m+2\ell-2} \eta_j (I - G_m)s_j
\end{bmatrix}
\right|_{q,r} \geq \max \left\{ \left| \begin{bmatrix}
\sum_{j=2m+2\ell-1}^{\infty} \eta_j (I - G_m)s_j
\end{bmatrix}
\right|_{q,r}, \left| \begin{bmatrix}
\sum_{j=2m+2\ell-1}^{\infty} \eta_j (I - \hat{G}_{m+\ell})s_j
\end{bmatrix}
\right|_{q,r} \right\},
$$

(4.62)

where $[M]_{q,r}$ denotes the $(q,r)$-entry of the matrix $M \in \mathbb{R}^{k \times k}$. Then $[G_m f]_{q,r}$ and $[\hat{G}_{m+\ell} f]_{q,r}$ bracket $[I f]_{q,r}$.

Proof. It follows from the expansions (4.60) and (4.61) that

$$
G_m f = I f - \sum_{j=2m}^{2m+2\ell-2} \eta_j (I - G_m)s_j - \sum_{j=2m+2\ell-1}^{\infty} \eta_j (I - G_m)s_j,
$$

(4.63)

$$
\hat{G}_{m+\ell} f = I f + \sum_{j=2m}^{2m+2\ell-2} \eta_j (I - G_m)s_j - \sum_{j=2m+2\ell-1}^{\infty} \eta_j (I - \hat{G}_{m+\ell})s_j.
$$

(4.64)

This shows (4.62).

We remark that bounds analogous to those of Theorem 32 can be established for the pair $\{G_m f, \hat{G}_{m+\ell} f\}$ of quadrature rule values. It is difficult to assess whether the bound (4.62) holds for a given function $f$, matrix $A$, and pair of quadrature rules $\{G_m, \hat{G}_{m+\ell}\}$. Nevertheless, Theorem 32 indicates that if the terms in the expansions (4.60) and (4.61) converge to zero quickly, then the pairs of quadrature rule values $\{G_m f, \hat{G}_{m+\ell} f\}$ are likely to bracket $I f$. This is in agreement with computational experience.

Corollary 33. Let the integrand $f$ have the expansion (4.59). Then the average rule (4.57) satisfies

$$
\hat{A}_{m+\ell} f = I f - \sum_{j=2m+2\ell-1}^{\infty} \eta_j (I - \hat{A}_{m+\ell})s_j.
$$

In particular, if $f \in \mathbb{P}^{2m+2\ell-2}$, then $\eta_j = 0$ for $j \geq 2m + 2\ell - 1$, and therefore $\hat{A}_{m+\ell} f = I f$, in agreement with (4.58).

Proof. The result follows by substituting the expansions (4.63) and (4.64) into (4.57). □
4.4 Computed Examples

We illustrate the performance of real-valued generalized anti-Gauss rules, simplified real-valued generalized anti-Gauss rules, and their block versions. We first describe a few applications of these quadrature rules to the approximation of real-valued functionals, and subsequently discuss applications to matrix-valued functions. Examples from network analysis also are presented.

Table 4.1: Example 34: $F(A) = u^T(I + A^2)^{-1}u$, $A$ a symmetric Toeplitz matrix.

<table>
<thead>
<tr>
<th>$m$</th>
<th>$\ell$</th>
<th>$(I - G_m)f$</th>
<th>$(I - G^{(\ell)}_{m+\ell})f$</th>
<th>$(I - A^{(\ell)}_{m+\ell})f$</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1</td>
<td>$4.48 \cdot 10^{-3}$</td>
<td>$-4.82 \cdot 10^{-6}$</td>
<td>$-4.16 \cdot 10^{-6}$</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>$-4.76 \cdot 10^{-5}$</td>
<td>$-1.39 \cdot 10^{-6}$</td>
<td>$2.04 \cdot 10^{-6}$</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>$-4.73 \cdot 10^{-5}$</td>
<td>$-1.29 \cdot 10^{-6}$</td>
<td>$2.07 \cdot 10^{-6}$</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>$-3.64 \cdot 10^{-4}$</td>
<td>$1.09 \cdot 10^{-9}$</td>
<td>$-1.10 \cdot 10^{-8}$</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>$3.64 \cdot 10^{-7}$</td>
<td>$-6.47 \cdot 10^{-11}$</td>
<td>$-1.22 \cdot 10^{-8}$</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>$3.64 \cdot 10^{-7}$</td>
<td>$-6.18 \cdot 10^{-11}$</td>
<td>$-1.22 \cdot 10^{-8}$</td>
</tr>
</tbody>
</table>

Example 34. We would like to compute approximations of the functional

$$\mathcal{I}f = F(A) := u^T(I + A^2)^{-1}u,$$

with a symmetric Toeplitz matrix $A \in \mathbb{R}^{200 \times 200}$ with first row $[1, 1/2, 1/3, \ldots, 1/200]$. Thus, $f(t) = (1 + t^2)^{-1}$. The vector $u$ has normally distributed entries with zero mean and is normalized to be of unit norm. The desired value is $F(A) \approx 1.36 \cdot 10^{-2}$. Table 4.1 shows the pairs $\{G_m f, \tilde{G}^{(\ell)}_{m+\ell} f\}$ as well as the pairs $\{G_m f, \hat{G}^{(\ell)}_{m+\ell} f\}$ to bracket $\mathcal{I}f$ for $\ell \in \{1, 2, 3\}$ and $m \in \{3, 6\}$. The simplified generalized anti-Gauss rules $\tilde{G}^{(\ell)}_{m+\ell} f$ are seen to give slightly more accurate approximations of $\mathcal{I}f$ than the (standard) generalized anti-Gauss rules $\hat{G}^{(\ell)}_{m+\ell} f$. We remark that pairs of Gauss and Gauss–Radau quadrature rules are not guaranteed to furnish upper and lower bounds for $\mathcal{I}f$, because higher derivatives of the integrand $f$ change sign on the convex hull of the spectrum of $A$. $\Box$

Example 35. We determine approximations of the functional

$$\mathcal{I}f = F(A) := u^T(I + A^2)^{-1}v,$$
We would like to approximate $F(A) = u^T (I + A^2)^{-1} v$, $A$ a nonsymmetric Toeplitz matrix. 

where $A$ is a $200 \times 200$ real nonsymmetric Toeplitz matrix with first row and column $[1, 1/2, 1/3, \ldots, 1/200]$ and $[1, 1, \ldots, 1]^T$, respectively. The vectors $u$ and $v$ have normally distributed random entries with zero mean; they are scaled so that $u^T v = 1$. The exact value is $F(A) \approx 5.36 \cdot 10^{-3}$. Since the matrix $A$ is nonsymmetric, the technique described by Golub and Meurant [46] of evaluating pairs of Gauss and Gauss–Radau quadrature rules is not guaranteed to furnish upper and lower bounds for $I f$. Table 4.2 shows the errors in approximations determined by Gauss, generalized anti-Gauss, simplified generalized anti-Gauss, and average quadrature rules. Pairs of Gauss rules $G_m f$ and generalized anti-Gauss rules $\tilde{G}_{m+\ell} f$ and pairs of Gauss rules and simplified generalized anti-Gauss rules $\tilde{G}_{m+\ell} f$ bracket $F(A)$ for $\ell \in \{1, 2, 3\}$ and $m = 4$, as well as for $\ell \in \{1, 3\}$ and $m = 5$, but not for $\ell = 2$ and $m = 5$. This illustrates that it may be beneficial to compute quadrature rules $\tilde{G}_{m+\ell} f$ or $\tilde{G}_{m+\ell} f$ for several values of $m$ and $\ell$. \[ \square \]

Table 4.2: Example 35: $F(A) = u^T (I + A^2)^{-1} v$, $A$ a nonsymmetric Toeplitz matrix.

<table>
<thead>
<tr>
<th>$m$</th>
<th>$\ell$</th>
<th>$(I - G_m) f$</th>
<th>$(I - \tilde{G}_{m+\ell} f)$</th>
<th>$(I - \tilde{A}_{m+\ell}^{(1)}) f$</th>
<th>$(I - \tilde{A}_{m+\ell}^{(2)}) f$</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>1</td>
<td>$-3.71 \cdot 10^{-3}$</td>
<td>$2.64 \cdot 10^{-3}$</td>
<td>$-5.41 \cdot 10^{-4}$</td>
<td>$4.41 \cdot 10^{-3}$</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>$2.43 \cdot 10^{-3}$</td>
<td>$-6.42 \cdot 10^{-4}$</td>
<td>$8.46 \cdot 10^{-3}$</td>
<td>$2.37 \cdot 10^{-3}$</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>$2.64 \cdot 10^{-3}$</td>
<td>$-5.41 \cdot 10^{-4}$</td>
<td>$4.41 \cdot 10^{-3}$</td>
<td>$3.47 \cdot 10^{-4}$</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>$1.46 \cdot 10^{-3}$</td>
<td>$-4.07 \cdot 10^{-3}$</td>
<td>$-1.31 \cdot 10^{-3}$</td>
<td>$-6.09 \cdot 10^{-3}$</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>$1.54 \cdot 10^{-3}$</td>
<td>$1.50 \cdot 10^{-3}$</td>
<td>$1.44 \cdot 10^{-3}$</td>
<td>$1.45 \cdot 10^{-3}$</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>$-4.07 \cdot 10^{-3}$</td>
<td>$-1.31 \cdot 10^{-3}$</td>
<td>$-2.68 \cdot 10^{-3}$</td>
<td>$-6.09 \cdot 10^{-4}$</td>
</tr>
</tbody>
</table>

Example 36. We would like to approximate

$I f = F(A) := u^T \exp(A) v,$

Table 4.3: Example 36: $F(A) = u^T \exp(A) v$, $A$ a nonsymmetric Toeplitz matrix.

<table>
<thead>
<tr>
<th>$m$</th>
<th>$\ell$</th>
<th>$(I - G_m) f$</th>
<th>$(I - \tilde{G}_{m+\ell} f)$</th>
<th>$(I - \tilde{A}_{m+\ell}^{(1)}) f$</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td>1</td>
<td>$-6.51 \cdot 10^1$</td>
<td>$-8.42 \cdot 10^1$</td>
<td>$-7.47 \cdot 10^1$</td>
</tr>
<tr>
<td>14</td>
<td>2</td>
<td>$-9.13 \cdot 10^1$</td>
<td>$-7.82 \cdot 10^1$</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>3</td>
<td>$1.57 \cdot 10^3$</td>
<td>$7.53 \cdot 10^2$</td>
<td></td>
</tr>
</tbody>
</table>
Table 4.4: Example 37: $F(A) = U^T \exp(A)U$, $A$ symmetric Toeplitz matrix.

<table>
<thead>
<tr>
<th>$m$</th>
<th>$\ell$</th>
<th>$(I - G_m)f$</th>
<th>$(I - G_{m+\ell}^{(f)})f$</th>
<th>$(I - A_{m+\ell}^{(f)})f$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>1</td>
<td>$[\begin{array}{cc} -8.98 &amp; -1.00 \ -1.00 &amp; -1.49 \end{array}] \cdot 10^{-4}$</td>
<td>$[\begin{array}{cc} 8.96 &amp; 1.00 \ 1.00 &amp; 1.49 \end{array}] \cdot 10^{-4}$</td>
<td>$[\begin{array}{cc} -9.84 &amp; 1.95 \ 1.95 &amp; -0.26 \end{array}] \cdot 10^{-7}$</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>$[\begin{array}{cc} 9.19 &amp; 1.03 \ 1.03 &amp; 1.54 \end{array}] \cdot 10^{-4}$</td>
<td>$[\begin{array}{cc} 1.04 &amp; 0.16 \ 0.16 &amp; 0.25 \end{array}] \cdot 10^{-5}$</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>$[\begin{array}{cc} 9.19 &amp; 1.03 \ 1.03 &amp; 1.54 \end{array}] \cdot 10^{-4}$</td>
<td>$[\begin{array}{cc} 1.06 &amp; 1.60 \ 1.60 &amp; 2.4 \end{array}] \cdot 10^{-5}$</td>
<td></td>
</tr>
</tbody>
</table>

Table 4.5: Example 37: $F(A) = U^T \exp(A)U$, $A$ symmetric Toeplitz matrix.

<table>
<thead>
<tr>
<th>$m$</th>
<th>$\ell$</th>
<th>$(I - G_m)f$</th>
<th>$(I - G_{m+\ell}^{(f)})f$</th>
<th>$(I - A_{m+\ell}^{(f)})f$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>1</td>
<td>$[\begin{array}{cc} -8.98 &amp; -1.00 \ -1.00 &amp; -1.49 \end{array}] \cdot 10^{-4}$</td>
<td>$[\begin{array}{cc} 8.96 &amp; 1.00 \ 1.00 &amp; 1.49 \end{array}] \cdot 10^{-4}$</td>
<td>$[\begin{array}{cc} -9.84 &amp; 1.95 \ 1.95 &amp; -0.26 \end{array}] \cdot 10^{-7}$</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>$[\begin{array}{cc} 9.23 &amp; 1.04 \ 1.04 &amp; 1.55 \end{array}] \cdot 10^{-4}$</td>
<td>$[\begin{array}{cc} 1.23 &amp; 0.20 \ 0.20 &amp; 0.30 \end{array}] \cdot 10^{-5}$</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>$[\begin{array}{cc} 9.25 &amp; 1.04 \ 1.04 &amp; 1.56 \end{array}] \cdot 10^{-4}$</td>
<td>$[\begin{array}{cc} 1.34 &amp; 0.20 \ 0.20 &amp; 0.32 \end{array}] \cdot 10^{-5}$</td>
<td></td>
</tr>
</tbody>
</table>

where $A$ is the nonsymmetric Toeplitz matrix of Example 15. The vectors $u$ and $v$ also are determined as in Example 15. Table 4.3 shows the pairs $\{G_{14} f, \hat{G}_{14+\ell}^{(f)}\}$ to bracket $F(A)$ for $\ell = 3$, but not for $\ell \in \{1, 2\}$. While the absolute quadrature errors are large, the relative errors are quite small since $\mathcal{I} f \approx 4.04 \cdot 10^{11}$. □

**Example 37.** This example discusses an application of generalized block anti-Gauss rules and simplified generalized block anti-Gauss rules to the evaluation of matrix functions of the form

$$\mathcal{I} f = F(A) := U^T \exp(A)U,$$

where $A \in \mathbb{R}^{200 \times 200}$ is the symmetric Toeplitz matrix of Example 14 and $U \in \mathbb{R}^{200 \times 2}$ is a
block vector with random orthonormal columns. Table 4.4 displays the block Gauss rule $G_5 f$ and the generalized block anti-Gauss rules $\tilde{G}_{5+\ell} f$ for $\ell = 1, 2, 3$ to give quadrature errors of component-wise opposite sign and of about the same magnitude. The desired value is

$$F(A) \approx \begin{bmatrix} 6.2667 & -0.2506 \\ -0.2506 & 0.0208 \end{bmatrix} \cdot 10^3.$$ 

This value is computed with the MATLAB function `expm`. Thus, the relative errors achieved with the quadrature rules of Tables 4.4 and 4.5 are quite small.

Table 4.5 shows the component-wise difference between the exact value and approximations determined by simplified generalized block anti-Gauss and associated average rules. They are defined by setting $\tilde{\Omega}_{m+\ell} = \Omega_{m+\ell-1}$. The simplified generalized block anti-Gauss and the block Gauss rule also bracket $F(A)$ component-wise. The average rules can be seen to give the best approximations of $F(A)$ in both Tables 4.4 and 4.5. □

Our last examples illustrate the application of the quadrature rules of this paper to quantities of interest in network analysis.

Table 4.6: Example 38: $I f = F(A) = e^T \exp(A) e$, $A$ the symmetric adjacency matrix for the Yeast network, $e = [1, 1, \ldots, 1]^T$.

<table>
<thead>
<tr>
<th>$m$</th>
<th>$\ell$</th>
<th>$(I - G_m) f$</th>
<th>$(I - G_{m+\ell}) f$</th>
<th>$(I - A^{(e)}_{m+\ell}) f$</th>
<th>$(I - \tilde{G}_{m+\ell}) f$</th>
<th>$(I - A^{(e)}_{m+\ell}) f$</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>1</td>
<td>$2.03 \cdot 10^{-4}$</td>
<td>$-2.03 \cdot 10^{-4}$</td>
<td>$-2.79 \cdot 10^{-7}$</td>
<td>$-2.03 \cdot 10^{-4}$</td>
<td>$-2.79 \cdot 10^{-7}$</td>
</tr>
<tr>
<td>8</td>
<td>2</td>
<td>$1.95 \cdot 10^{-7}$</td>
<td>$4.02 \cdot 10^{-6}$</td>
<td>$-2.03 \cdot 10^{-4}$</td>
<td>$-1.95 \cdot 10^{-7}$</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>$1.97 \cdot 10^{-7}$</td>
<td>$-1.97 \cdot 10^{-7}$</td>
<td>$8.77 \cdot 10^{-10}$</td>
<td>$-1.95 \cdot 10^{-7}$</td>
<td>$8.77 \cdot 10^{-10}$</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>$1.97 \cdot 10^{-7}$</td>
<td>$-2.72 \cdot 10^{-11}$</td>
<td>$-1.97 \cdot 10^{-7}$</td>
<td>$-1.80 \cdot 10^{-10}$</td>
<td></td>
</tr>
</tbody>
</table>

Example 38. We consider the network Yeast, which is represented by an undirected graph with 2114 vertices and 4480 edges. It describes the protein interaction of yeast. We would like to determine the total communicability $F(A) = e^T \exp(A) e$, where $e = [1, 1, \ldots, 1]^T$, of the network; see [11, 62]. We use quadrature rules of Section 3.1 to provide estimates. The exact value is approximately 221. Table 4.6 shows the errors in computed approximations.
of the total communicability determined by Gauss, generalized anti-Gauss, simplified
generalized anti-Gauss, and average quadrature rules. Quadrature rules with only a few nodes
are seen to yield approximations with higher accuracy than what typically is required in
applications. Pairs of Gauss and generalized anti-Gauss, as well as pairs of Gauss and
simplified generalized anti-Gauss rules are seen to bracket the exact value.

Table 4.7: Example 38: \( I f = F(A) = e^T \exp(A)e \), \( A \) a nonsymmetric adjacency matrix for
the modified Yeast network, \( e = [1, 1, \ldots, 1]^T \).

<table>
<thead>
<tr>
<th>( m )</th>
<th>( \ell )</th>
<th>( (I - \mathcal{G}_m)f )</th>
<th>( (I - \mathcal{G}_m^{(\ell)} f) )</th>
<th>( (I - \mathcal{A}_m^{(\ell)} f) )</th>
<th>( (I - \mathcal{A}_m^{(\ell)} f) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>1</td>
<td>(-4.12 \cdot 10^{-3})</td>
<td>(4.18 \cdot 10^{-3})</td>
<td>(2.58 \cdot 10^{-3})</td>
<td>(3.75 \cdot 10^{-3})</td>
</tr>
<tr>
<td>7</td>
<td>2</td>
<td>(4.28 \cdot 10^{-3})</td>
<td>(7.67 \cdot 10^{-5})</td>
<td>(3.87 \cdot 10^{-3})</td>
<td>(-1.26 \cdot 10^{-4})</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>(-2.02 \cdot 10^{-4})</td>
<td>(2.03 \cdot 10^{-4})</td>
<td>(2.98 \cdot 10^{-1})</td>
<td>(2.07 \cdot 10^{-4})</td>
</tr>
<tr>
<td>8</td>
<td>2</td>
<td>(2.03 \cdot 10^{-4})</td>
<td>(4.08 \cdot 10^{-7})</td>
<td>(2.07 \cdot 10^{-4})</td>
<td>(2.27 \cdot 10^{-6})</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>(-1.96 \cdot 10^{-7})</td>
<td>(1.95 \cdot 10^{-7})</td>
<td>(-8.43 \cdot 10^{-10})</td>
<td>(1.98 \cdot 10^{-7})</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>(1.99 \cdot 10^{-7})</td>
<td>(1.37 \cdot 10^{-9})</td>
<td>(2.04 \cdot 10^{-7})</td>
<td>(3.69 \cdot 10^{-9})</td>
</tr>
</tbody>
</table>

It is easy to modify the Yeast network to obtain a directed network with a nonsymmetric
adjacency matrix. Then the technique by Golub and Meurant is not guaranteed to determine
bounds for the total communicability. We replace the superdiagonal entries in rows 2 to
4 of the adjacency matrix by zero. This implies that we replace “two-way streets” to the
nodes 2 to 4, by “one-way streets”. Table 4.7 shows that pairs of Gauss and generalized
anti-Gauss or simplified generalized anti-Gauss rules provide upper and lower bounds for
the total communicability for this directed network. \( \square \)

Table 4.8: Example 39: Subgraph centrality for the vertices 100, 224, and 1000 of the
modified Yeast network.

| \( t \) | \(|f(A)|_{ii}\) | \(|f(A)|_{ii} - \mathcal{G}_m f\) |
|---|---|---|
| 100 | \(3.919 \cdot 10^7\) | \(-1.54 \cdot 10^{-6}\) |
| 224 | \(9.124 \cdot 10^2\) | \(-6.71 \cdot 10^{-6}\) |
| 1000 | \(1.031 \cdot 10^1\) | \(-1.58 \cdot 10^{-5}\) |

Example 39. Let \( f(t) = \exp(t) \). The subgraph centrality of the node \( i \) is defined as the \( i \)th
diagonal entry, \(|f(A)|_{ii}\), of \( f(A) \). A (relatively) large value indicates that node \( i \) is important;
see [31] for a discussion. We compute approximations of the subgraph centralities \(|f(A)|_{ii}\)
for $i \in \{100, 224, 1000\}$ for the modified Yeast network described in above example. Thus, the adjacency matrix is nonsymmetric. Table 4.8 displays the exact values and the errors obtained by using the Gauss rule $G_8 f$. The application of this rule to approximate $[f(A)]_{ii}$ requires 8 steps of the nonsymmetric Lanczos process applied to $A$ with the axis vector $e_i = [0, \ldots, 0, 1, 0, \ldots, 0]^T$ as initial vectors. Table 4.9 shows the errors in approximations determined by corresponding generalized anti-Gauss and simplified generalized anti-Gauss rules. The table also shows the errors for the associated average rules. Table 4.8 shows node 224 to be more important than the nodes 100 and 1000. □

Table 4.10: Example 40: $I f = F(A) = e^T \exp(A)e$, $A$ a nonsymmetric adjacency matrix for the Celegans network, $e = [1, 1, \ldots, 1]^T$.

<table>
<thead>
<tr>
<th>$m$</th>
<th>$\ell$</th>
<th>$(I - G_m)f$</th>
<th>$(I - A_{m+1})f$</th>
<th>$(I - G_{m+1}^{(t)})f$</th>
<th>$(I - A_{m+1}^{(t)})f$</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1</td>
<td>8.49 · 10^2</td>
<td>-9.83 · 10^2</td>
<td>-6.66 · 10^1</td>
<td>-2.39 · 10^3</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>-4.35 · 10^3</td>
<td>-1.75 · 10^3</td>
<td>-7.65 · 10^3</td>
<td>-3.40 · 10^3</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>3.59 · 10^3</td>
<td>-4.89 · 10^3</td>
<td>-6.49 · 10^-1</td>
<td>-4.24 · 10^0</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>-1.15 · 10^2</td>
<td>-5.60 · 10^1</td>
<td>-8.85 · 10^1</td>
<td>-4.24 · 10^1</td>
</tr>
</tbody>
</table>

**Example 40.** Consider the metabolic network Celegans with 306 nodes and 2345 edges of the nematode (roundworm) caenorhabditis elegans [27]. The data set is available at [3]. The network is directed, i.e., the adjacency matrix is nonsymmetric. Table 4.10 shows the errors of computed approximations of the total communicability determined by Gauss, generalized anti-Gauss, simplified generalized anti-Gauss, and average quadrature rules. The exact value is about $I f \approx 1.09 \cdot 10^4$. This example illustrates that accuracy that typically is sufficient in applications can be achieved already with quadrature rules with
very few nodes. The quadrature rules are seen to bracket the exact value.
In this chapter, we are concerned with the evaluation of expressions of the form

\[(5.1) \quad F(A) := W^T f(A)V,\]

where \(A \in \mathbb{R}^{n \times n}\) is a large symmetric matrix, \(f\) is a function that is analytic on the convex hull of the spectrum of \(A\), \(W \in \mathbb{R}^{n \times r}\) and \(V \in \mathbb{R}^{n \times r'}\) with \(1 \leq r, r' \ll n\) are block vectors. When \(A\) is small, the expression (5.1) can be evaluated by first explicitly evaluating \(f(A)\) and multiplying this matrix by \(W^T\) and \(V\). Many algorithms for evaluating matrix functions \(f(A)\) for small to moderately sized matrices \(A\) are described and analyzed by Higham [52]. We are interested in the situation when the matrix \(A\) is so large that the explicit computation of \(f(A)\) requires too much computational effort or computer memory to be practical or attractive.

The present chapter is concerned with the situation when \(r \neq r'\). Let for the moment \(1 = r' < r\). We will use the notation \(v = V\) and \(W = [w_1, w_2, \ldots, w_r]\). Introduce the spectral factorization

\[(5.2) \quad A = U \Lambda U^T,\]

where the nontrivial entries of \(\Lambda = \text{diag}[\lambda_1, \lambda_2, \ldots, \lambda_n] \in \mathbb{R}^{n \times n}\) are the eigenvalues of \(A\) and the matrix \(U \in \mathbb{R}^{n \times n}\) is orthogonal. Substituting the factorization (5.2) into (5.1) yields
the sums

\[
F(A) = [w_1^T f(A)v, w_2^T f(A)v, \ldots, w_r^T f(A)v]^T = [\tilde{w}_1^T f(\Lambda)\tilde{v}, \tilde{w}_2^T f(\Lambda)\tilde{v}, \ldots, \tilde{w}_r^T f(\Lambda)\tilde{v}]^T
\]

where \(\tilde{w}_j = [\tilde{w}_{j,1}, \tilde{w}_{j,2}, \ldots, \tilde{w}_{j,n}]^T = U^T w_j, 1 \leq j \leq r\), and \(\tilde{v} = [\tilde{v}_1, \tilde{v}_2, \ldots, \tilde{v}_n]^T = U^T v\). We formally write these sums as Stieltjes integrals associated with measures \(d\omega_j\) with piecewise constant distribution functions \(\omega_j, 1 \leq j \leq r\). Thus,

\[
F(A) = \left[ \int f(t)d\omega_1(t), \int f(t)d\omega_2(t), \ldots, \int f(t)d\omega_r(t) \right]^T.
\]

This shows that our computational task is to determine approximations of integrals of the function \(f\) with respect to \(r\) measures.

One approach to approximate \(F(A)\) is to compute approximations of each one of the \(r\) integrals in (5.3) independently. This can be achieved by approximating each one of the expressions \(w_j^T f(A)v\) by applying the nonsymmetric Lanczos algorithm to \(A\) with initial vectors \(w_j\) and \(v\) independently for \(1 \leq j \leq r\); see [23, 35]. However, computed examples reported in Section 3.3 show that it may be possible to determine an approximation of \(F(A)\) of the same quality with fewer matrix-vector product evaluations by computing an approximation using Gauss-type quadrature rules associated with multiple orthogonal polynomials. These polynomials satisfy certain orthogonality conditions with respect to several inner products or bilinear forms determined by the different measures. Since for large matrices \(A\), the main computational effort for determining approximations of \(F(A)\) is the evaluation of matrix-vector products with \(A\), reducing the number of these products is beneficial.

Multiple orthogonal polynomials arise in quadrature problems when the same function has to be integrated with respect to several measures and one would like to use the same nodes for the quadrature rules for each measure. They also emerge in simultaneous Padé
approximations when several functions are to be approximated by rational functions with the same denominator; see, e.g., [26, 36, 37, 38, 58, 59]. An application to illumination models is described by Borges [17]. We are not aware of applications of multiple orthogonal polynomials and associated Gauss-type quadrature rules to the evaluation of matrix functions of the form (5.1). It is the purpose of this chapter to discuss this application.

The evaluation of our approximation of $F(A)$ requires the use of a generalized Lanczos algorithm. Such an algorithm can be derived from the recursion relations of multiple orthogonal polynomials; see, e.g., [26, 58, 59] and Section 3.1. One also can derive a generalized Lanczos algorithm by using linear algebra techniques only, without reference to multiple orthogonal polynomials. Derivations of the latter kind are described by Aliaga et al. [1] and Freund [40]. These authors focus on the handling of breakdowns of the recursion relations. It is well known that the standard nonsymmetric Lanczos algorithm may suffer from breakdowns or near breakdowns; see, e.g., [5, 65, 71] for discussions. This also holds for the generalized Lanczos algorithm. This algorithm also can be applied to model reduction; see Freund [41] for a nice survey.

This chapter is organized as follows. Section 5.1 reviews results about multiple orthogonal polynomials and associated Gauss-type quadrature rules. In particular, we discuss the recurrence relation for multiple orthogonal polynomials; the recurrence formula can be expressed with a lower Hessenberg matrix, whose lower bandwidth depends on the number of measures. A few computed examples are presented in Section 5.2.

5.1 Multiple Orthogonal Polynomials

This section provides an overview of multiple orthogonal polynomials associated with $r$ distinct measures on the real line. For more details on multiple orthogonal polynomials, we refer to [26, 37, 38, 58, 59].

Let $r \geq 1$ be an integer and let $d\omega_1, d\omega_2, \ldots, d\omega_r$ denote nonnegative measures with
support on (part of) the real axis. Introduce the multi-index \( \vec{m} = [m_1, m_2, \ldots, m_r] \), whose entries are nonnegative integers. The length of \( \vec{m} \) is defined as \( |\vec{m}| = m_1 + m_2 + \ldots + m_r \). We are concerned with type II multiple orthogonal polynomials. They are commonly labelled with a multi-index. Thus, the type II multiple orthogonal polynomial \( p_{\vec{m}} \) is a monic orthogonal polynomial of degree \( |\vec{m}| \) that satisfies the orthogonality conditions

\[
\int p_{\vec{m}}(t) t^k d\omega_1(t) = 0, \quad k = 0, 1, \ldots, m_1 - 1,
\]
\[
\int p_{\vec{m}}(t) t^k d\omega_2(t) = 0, \quad k = 0, 1, \ldots, m_2 - 1,
\]
\[
\vdots
\]
\[
\int p_{\vec{m}}(t) t^k d\omega_r(t) = 0, \quad k = 0, 1, \ldots, m_r - 1.
\]

We assume the measures to be such that we can define \( r \) inner products or bilinear forms for functions \( f \) and \( g \) that are sufficiently smooth on the union of the supports of the measures,

\[
(f, g)_k = I_k(f g), \quad 1 \leq k \leq r.
\]

For \( r = 1 \), the monic multiple orthogonal polynomials simplify to monic polynomials that are orthogonal with respect to one inner product. The above orthogonality conditions yield a linear system of \( |\vec{m}| \) equations. There are \( |\vec{m}| \) unknown coefficients \( \beta_{k,\vec{m}}, k = 0, 1, \ldots, |\vec{m}| \) of the polynomial

\[
p_{\vec{m}}(t) = t^{|\vec{m}|} + \sum_{k=0}^{|\vec{m}|-1} \beta_{k,\vec{m}} t^k.
\]

We will assume the index vectors to be of the form

\[
\vec{m} = [\underbrace{\ell + 1, \ell + 1, \ldots, \ell + 1}_j, \underbrace{\ell, \ell, \ldots, \ell}_{r-j}],
\]

where \( |\vec{m}| = \ell r + j \) for some integers \( \ell \geq 0 \) and \( 0 \leq j < r \). Then the polynomial \( p_{\vec{m}} \) can be identified with the polynomial \( p_m \) with the simple index \( m = \ell r + j \). We will write \( p_m = p_{\vec{m}} \).

The monic multiple orthogonal polynomials satisfy a recurrence relation of the form

\[
(t p_m)(t) = p_{m+1}(t) + \sum_{i=0}^{\min\{r,m\}} \alpha_{m-r-i} p_{m-i}(t), \quad m \geq 0.
\]
We define \( p_0(t) \equiv 1 \) and \( p_j(t) \equiv 0 \) for \( j = -1, -2, \ldots, -r \). The recurrence relations for the first \( m + 1 \) monic multiple orthogonal polynomials can be expressed as

\[
\begin{pmatrix}
p_0(t) \\
p_1(t) \\
\vdots \\
p_{m-1}(t)
\end{pmatrix} = H_m \begin{pmatrix}
p_0(t) \\
p_1(t) \\
\vdots \\
p_{m-1}(t)
\end{pmatrix} + p_m(t) \begin{pmatrix} 0 \\
0 \\
\vdots \\
1 \end{pmatrix},
\]

where the matrix \( H_m \in \mathbb{R}^{m \times m} \) is lower Hessenberg and banded,

\[
H_m := \begin{bmatrix}
\alpha_{0,r} & 1 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\
\alpha_{1,r} & \alpha_{1,r-1} & 1 & \ddots & & & & \\
\vdots & \vdots & \ddots & \ddots & \ddots & & & \\
\alpha_{r,0} & \cdots & \alpha_{r,r-1} & \alpha_{r,r} & 1 & & & \\
\vdots & \vdots & \ddots & \ddots & \ddots & \ddots & & \\
\vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \\
\vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \alpha_{m-2,0} \\
0 & \cdots & \cdots & \cdots & \cdots & \cdots & \alpha_{m-1,0} & \alpha_{m-1,r-1} & \alpha_{m-1,r}
\end{bmatrix}.
\]

The entries of the above matrix can be determined by evaluating certain inner products.

We have

\[
p_j(t) = (t - \alpha_{j,r}) p_j(t) - \sum_{i=1}^{\min\{r,j\}} \alpha_{j,r-i} p_{j-i}(t), \quad j = 0, 1, \ldots.
\]

Let \( j = \ell r + v, \ell = \lfloor j/r \rfloor \), and \( v \in \{0, 1, 2, \ldots, r - 1\} \). Then

\[
\alpha_{j,0} = \frac{(tp_j, P_{\lfloor (j-r)/r \rfloor})_{v+1}}{(p_{j-r}, P_{\lfloor (j-r)/r \rfloor})_{v+1}}, \quad j = r, r + 1, \ldots,
\]

and

\[
\alpha_{j,k} = \frac{(tp_j - \sum_{i=0}^{k-1} \alpha_{j,i} p_{j-r+i} P_{\lfloor (j-r+k)/r \rfloor})_{1+((v+k) \mod r)}}{(p_{j-r+k}, P_{\lfloor (j-r+k)/r \rfloor})_{1+((v+k) \mod r)}},
\]

for \( k = \max\{r-j, 1\}, \max\{r-j, 1\} + 1, \ldots, r \) and \( j = 0, 1, \ldots \). Here \( \lfloor t \rfloor \) denotes the integer part of \( t \geq 0 \); when \( t \) is negative \( \lfloor t \rfloor \) is the smallest integer bounded below by \( t \). Having
computed the entry $\alpha_{0,r}$ of $H_m$, the first row of equation (5.6) yields the polynomial $p_1$, and we can evaluate the entries $\alpha_{1,r}$ and $\alpha_{1,r-1}$ in the second row of $H_m$. The latter entries determine the polynomial $p_2$. Using (5.6), we can proceed in the same manner to determine the entries of $H_m$ row by row until all entries are known. This is the Stieltjes procedure for computing multiple orthogonal polynomials.

Algorithm 5: Generation of multiple orthogonal polynomials.

1: **Input:** Symmetric matrix $A \in \mathbb{R}^{n \times n}$, initial block vectors $V = v_0 \in \mathbb{R}^n$,
2: $W = [w_{0,1}, w_{0,2}, \ldots, w_{0,r}] \in \mathbb{R}^{n \times r}$,
3: $v_j := w_{0,j} := [0, 0, \ldots, 0]^T \in \mathbb{R}^n$ for $j = -1, -2, \ldots, -r$,
4: $\alpha_{r-1,0} := 0$, $\alpha_{0,r} := v_0^T A w_{0,1} / v_0^T w_{0,1}$,
5: number of steps $m$ (a multiple of $r$).
6: $v_1 := A v_0 - \alpha_{0,r} v_0$
7: for $\ell = 1$ to $r$
8: $w_{1,\ell} := A w_{0,\ell} - \alpha_{0,r} w_{0,\ell}$
9: end for
10: for $j = 1$ to $m$
11: $s_j := \lfloor (j - r)/r \rfloor$
12: $\beta_{j,0} := v_j^T \beta_{j,0}
13: if $\beta_{j,0} \neq 0$ then $\alpha_{j,0} := (A v_j, w_{s_j,1+(j mod r)}) / \beta_{j,0}$ else $\alpha_{j,0} := 0$
14: for $k = 1$ to $r$
15: $s_{j+k} := \lfloor (j - r + k)/r \rfloor$
16: $\beta_{j,k} := v_j^T \beta_{j,k}
17: if $\beta_{j,k} \neq 0$ then $\alpha_{j,k} := (A v_j - \sum_{i=0}^{k-1} \alpha_{j,i} v_j - i, w_{s_{j+k},1+(j+k mod r)}) / \beta_{j,k}$
18: else $\alpha_{j,k} := 0$
19: end
20: end
21: end
22: end for
23: for $\ell = 1$ to $r$
24: $w_{j+1,\ell} := A w_{j,\ell} - \alpha_{j,r} w_{j,\ell} - \sum_{i=1}^{\min\{r,j\}} \alpha_{j,r-i} w_{j-i,\ell}$
25: end for
26: $v_{j+1} := A v_j - \alpha_{j,r} v_j - \sum_{i=1}^{\min\{r,j\}} \alpha_{j,r-i} v_{j-i}$
27: end for
28: **Output:** Entries $\alpha_{0,r}, \alpha_{1,r-1}, \alpha_{1,r}, \ldots, \alpha_{r,0}, \ldots, \alpha_{r,r-1}, \alpha_{r,r}, \ldots,$ and $\alpha_{m-1,0}, \ldots, \alpha_{m-1,r-1}, \alpha_{m-1,r}$ of the lower Hessenberg matrix $H_m$.

The computations are summarized by Algorithm 5. Each execution of the $k$-loop yields
one new coefficient of the matrix $H_m$. The matrix-vector products $Aw_{s_j,1+(j \mod r)}$ on lines 13 and $Aw_{s_{j+k},1+(j+k) \mod r}$ on line 18 already have been computed previously and do not have to be recomputed if they have been stored. Thus, execution of the algorithm requires $(r + 1)m$ matrix-vector product evaluations. For notational simplicity, we assume that $m$ is a multiple of $r$ in the algorithm. This restriction easily can be removed.

We approximate the integrals in (5.3) by Gauss-type quadrature rules associated with the multiple orthogonal polynomials $p_0, p_1, \ldots$ and each one of the measures $d\omega_1, d\omega_2, \ldots, d\omega_r$. For each $1 \leq j \leq r$, the $m$-point Gauss-type rule associated with the measure $d\omega_j$ can be expressed as

$$G_{j,m}f = \sum_{\ell=1}^{m} w_{j,\ell,m} f(t_{\ell})$$

and satisfies

$$G_{j,m}f = I_j f \quad \forall f \in P_{m+m_j-1},$$

where $P_j$ denotes the set of polynomials of degree at most $j$, and $m_j$ is the $j$th component of the multi-index $\vec{m}$ associated with $m$, i.e., $m = |\vec{m}|$ with $\vec{m} = [m_1, m_2, \ldots, m_r]$. The quadrature nodes $t_1, t_2, \ldots, t_m$ are independent of $j$; they are the eigenvalues of the matrix $H_m$. We will assume that the nodes are pairwise distinct. A sufficient condition for this to hold is that the measures $d\omega_1, d\omega_2, \ldots, d\omega_r$ form an AT system; see [17, 26].

Following Milovanović and Stanić [58], we compute the weights by solving $r$ linear systems of equations with the same matrix and different right-hand sides. Let $t_1, t_2, \ldots, t_m$ be the eigenvalues of $H_m$. The columns of the Vandermonde-like matrix $P$ are

$$P = \begin{bmatrix}
p_0(t_1) & p_0(t_2) & \cdots & p_0(t_m) 
p_1(t_1) & p_1(t_2) & \cdots & p_1(t_m) 
\vdots & \vdots & \ddots & \vdots 
p_{m-1}(t_1) & p_{m-1}(t_2) & \cdots & p_{m-1}(t_m)
\end{bmatrix}$$
are eigenvectors of $H_m$. Since the polynomials $p_j$ are monic, all entries of the first row equal one. The weights for the quadrature rule associated with the measure $d\omega_k$ satisfy

$$P \begin{bmatrix} w_{j,1,m} \\ w_{j,2,m} \\ \vdots \\ w_{j,m,m} \end{bmatrix} = \begin{bmatrix} (p_0,p_0)_j \\ (p_1,p_0)_j \\ \vdots \\ (p_{m-1},p_0)_j \end{bmatrix}, \quad 1 \leq j \leq r.$$

These systems can be solved by fast solution methods. For instance, the solution method by Higham [51] first computes an UL factorization of $P$ in $O(m^2)$ arithmetic floating point operations (flops) and then solves the system of equations by backward and forward substitution also in $O(m^2)$ flops. A different approach to computing the weights is described in [26].

We have so far focused on the situation when $r' = 1$ and $r \geq 1$ in (5.1). It is straightforward to extend the discussion to the situation when $r' > 1$. Let $V = [v_1,v_2,\ldots,v_{r'}] \in \mathbb{R}^{n \times r'}$ and $W = [w_1,w_2,\ldots,w_r] \in \mathbb{R}^{n \times r}$. Then (5.1) can be expressed as

$$F(A) = \begin{bmatrix} w_1^T f(A)v_1 & w_1^T f(A)v_2 & \cdots & w_1^T f(A)v_{r'} \\ w_2^T f(A)v_1 & w_2^T f(A)v_2 & \cdots & w_2^T f(A)v_{r'} \\ \vdots & \vdots & \ddots & \vdots \\ w_r^T f(A)v_1 & w_r^T f(A)v_2 & \cdots & w_r^T f(A)v_{r'} \end{bmatrix}.$$

Substituting the spectral factorization (5.2) of $A$ into $f(A)$ shows that each term can be expressed as a Stieltjes integral, analogously to (5.3). We can estimate these integrals using multiple orthogonal polynomials for $rr'$ weight functions.

5.2 Computed Examples

We illustrate the performance of Gauss quadrature rules which we apply for multiple orthogonal polynomials for the two measures as presented in this chapter. We first describe a few applications of these quadrature rules to the approximation of real-valued functionals.
Table 5.1: Example 41: \( \mathcal{I}f = W^T(I + A)^{-1}V, \) \( A \) a symmetric Toeplitz matrix.

<table>
<thead>
<tr>
<th>( m )</th>
<th>( \mathcal{I}f )</th>
<th>( W^T(I + A)^{-1}V_1 )</th>
<th>( W^T(I + A)^{-1}V_2 )</th>
<th>( W^T(I + A)^{-1}V_k )</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>4.8263 4.7883 ( \cdot 10^{-3} )</td>
<td>5.78 ( \cdot 10^{-4} )</td>
<td>2.28 ( \cdot 10^{-3} )</td>
<td>[-1.20 -5.54] ( \cdot 10^{-5} )</td>
</tr>
<tr>
<td>5</td>
<td>2.16 ( \cdot 10^{-5} )</td>
<td>1.00 ( \cdot 10^{-4} )</td>
<td>[-1.26 -5.66] ( \cdot 10^{-5} )</td>
<td></td>
</tr>
</tbody>
</table>

These examples use different types of functions as well as different matrices \( A \). The examples illustrate that Gauss quadrature rules provide approximation for the expressions (5.1) in a variety of situations. We compare the computed approximations obtained with quadrature rules to values determined by explicitly evaluating the functions (5.1).

**Example 41.** This example shows that the use of multiple orthogonal polynomials and associated quadrature rules to approximate (5.1) with \( W \in \mathbb{R}^{200} \) and \( V = [V_1, V_2] \in \mathbb{R}^{200 \times 2} \) may require fewer matrix-vector product evaluations than application of the nonsymmetric Lanczos process to \( A \) with initial vector pairs \( \{W, V_1\} \) and \( \{W, V_2\} \), independently. Table 5.1 shows the error of computed approximations of the functional

\[
F(A) := W^T(I + A)^{-1}V,
\]

where \( A \in \mathbb{R}^{200 \times 200} \) is a symmetric pentadiagonal Toeplitz matrix with first row . The vector \( W \) has normally distributed entries with zero mean and is normalized to be of unit norm, and the block vector \( V \) is given by \( V = [e_1 + 3/2e_2 + 2e_3, e_1 + 2e_2 + 3e_3] + E \), where \( e_j = [0, \ldots, 0, 1, 0, \ldots, 0]^T \in \mathbb{R}^{200} \) denotes the \( j \)th axis vector and \( E = [e, e] \) with \( e = [0, 0, 0, 1, 1, \ldots, 1]^T \in \mathbb{R}^{200} \). The exact value of \( F(A) \) is provided in the first column of Table 5.1. The second column of the table shows the error in the computed approximation of the first component of \( F(A) \) using the nonsymmetric Lanczos process. Specifically, \( m \) steps of this process is applied to \( A \) with initial vectors \( W \) and \( V_1 \), and the connection with Gauss quadrature is exploited to determined an approximation of the first component of
Table 5.2: Example 42: $\mathcal{I}f = W^T(I + A)^{-1}V$, $A$ a symmetric Toeplitz matrix.

<table>
<thead>
<tr>
<th>$m$</th>
<th>$\mathcal{I}f$</th>
<th>$\mathcal{E}_m f = \mathcal{I}f - \mathcal{G}_m f$</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>$[4.0380 \ 4.0380] \cdot 10^{-1}$</td>
<td>$[\ 9.85 \ 9.87] \cdot 10^{-5}$</td>
</tr>
<tr>
<td>4</td>
<td>$[-2.14 \ -1.49] \cdot 10^{-6}$</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>$[-0.21 \ -9.71] \cdot 10^{-7}$</td>
<td></td>
</tr>
</tbody>
</table>

$F(A)$. A detailed description of the nonsymmetric Lanczos process and its application to the evaluation of matrix functions can be found in [35]. The evaluation of the approximation so determined requires $2m$ matrix-vector product evaluations with $A$. An approximation of the second component of $F(A)$ is computed similarly, i.e., by applying $m$ steps of the nonsymmetric Lanczos process to $A$ with initial vector pair $\{W, V_2\}$. The error of the approximations so determined are found in the second column of Table 5.1. The evaluation of this approximation also requires the evaluation of $2m$ matrix-vector products with $A$. Thus, the computation of the first two columns of the table demands $4m$ matrix-vector products. The last column of Table 5.1 displays the error in approximations of the components of $F(A)$ determined by using $m$ steps with Algorithm 5. Their evaluation requires $3m + 3$ matrix-vector product evaluation with $A$. Thus, Algorithm 5 gives higher accuracy and requires less number of matrix-vector product evaluations than two applications of the nonsymmetric Lanczos process. □

**Example 42.** We would like to compute approximations of the functional

$$\mathcal{I}f = F(A) := W^T(I + A)^{-1}V,$$

with a symmetric Toeplitz matrix $A \in \mathbb{R}^{100 \times 100}$ with first row $[1/2, 1/2^2, 1/2^3, \ldots, 1/2^{100}]$. Thus, $f(t) = (1 + t^2)^{-1}$. The vector $W \in \mathbb{R}^{100 \times 1}$ has normally distributed entries with zero mean and is normalized to be of unit norm, and $V \in \mathbb{R}^{100 \times 2}$ also has normally distributed
Table 5.3: Example 43: \( I f = W^T (\sqrt{A}) V, \ A \) a symmetric Toeplitz matrix.

<table>
<thead>
<tr>
<th>( m )</th>
<th>( \overline{I f} )</th>
<th>( \mathcal{E}<em>{mf} = I f - \mathcal{G}</em>{mf} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>[ 3.74825 \ 3.74825 ]</td>
<td>[ 4.44 \ 8.88 ] ( \cdot 10^{-16} )</td>
</tr>
<tr>
<td>4</td>
<td>[ 2.22 \ 2.66 ] ( \cdot 10^{-15} )</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>[ -3.10 \ -2.22 ] ( \cdot 10^{-15} )</td>
<td></td>
</tr>
</tbody>
</table>

Entries with zero mean and is normalized to be of unit norm. Table 5.2 shows the desired value \( I f \) and the error of the Gauss quadrature rules for different values of \( m \). It is clear that our error size is dependent on the size of our initial vectors from both \( W \) and \( V \).

**Example 43.** We would like to compute approximations of the functional

\[
F(A) := W^T (\sqrt{A}) V,
\]

with a symmetric Toeplitz matrix \( A \in \mathbb{R}^{100 \times 100} \) with first row \([1, 1/2, \ldots, 1/100] \). The vector \( W \in \mathbb{R}^{100 \times 1} \) has normally distributed entries with zero mean and is normalized to be of unit norm, and this also holds for the columns of \( V \in \mathbb{R}^{100 \times 2} \). Table 5.3 shows component-wise errors in computed approximations of \( F(A) \) determined by Gauss quadrature rules for \( m \in \{3, 5, 6\} \). The desired value \( I f \) is shown Table 5.3. It is clearly in this example how the error is small.
CHAPTER 6

Conclusion

Matrix Functionals (3.1) where \( v, w \) are given vectors or block vectors, \( A \) is a symmetric or nonsymmetric matrix, and \( f \) a smooth function are commonly approximated by the use of Gauss quadrature rules. Laurie [55] introduced a class of quadrature rules, that he referred to as anti-Gauss rules, for the estimation of the error in associated Gaussian quadrature rules. We derived new simplified anti-Gauss quadrature rules and discussed their evaluation. The advantage of these rules, when compared with the anti-Gauss rules discussed in [35], is that they are cheaper to evaluate. Their use is attractive when the computation of the recursion coefficients for orthogonal polynomials is expensive. Computed examples in chapter 3 illustrate that pairs of Gauss and simplified anti-Gauss quadrature rules yield upper and lower bounds for many matrix functions. When the matrix functions are matrix-valued, the computed upper and lower bounds are element-wise.

We extend the anti-Gauss quadrature rule to the generalized anti-Gauss quadrature rule. In chapter 4 we derived new generalized anti-Gauss quadrature rules, and discussed their properties and computation. Also a family of generalized block anti-Gauss quadrature rules are introduced. The simplified rules also derived in that chapter are easier to evaluate and may exist also when generalized anti-Gauss quadrature rules do not. Computed examples in chapter 4 , which include applications to network analysis, illustrate the performance of the new quadrature rules.

At last, inspired by the intimate relation between the Lanczos method for initial block vectors with different number of columns and multiple orthogonal polynomials, we presented a new technique making use of multiple orthogonal polynomials and the Stieltjes
procedure. We gave a computed example in Section 5.2 to demonstrate the applicability of
the procedures. This example shows that it may be possible to determine an approximation
of $F(A)$ of the same quality with fewer matrix-vector product evaluations by computing
an approximation using Gauss-type quadrature rules associated with multiple orthogonal
polynomials than by applying the standard Lanczos method.
BIBLIOGRAPHY


