The series includes handbooks and software guides as well as monographs on practical implementation of computational methods, environments, and tools. The focus is on making recent developments available in a practical format to researchers and other users of these methods and tools.

**Editor-in-Chief**
Jack J. Dongarra
University of Tennessee and Oak Ridge National Laboratory

**Editorial Board**
James W. Demmel, University of California, Berkeley
Dennis Gannon, Indiana University
Eric Grosse, AT&T Bell Laboratories
Ken Kennedy, Rice University
Jorge J. Moré, Argonne National Laboratory

**Software, Environments, and Tools**

Louis Komzsik, *The Lanczos Method: Evolution and Application*


Stefan Goedecker and Adolfy Hoisie, *Performance Optimization of Numerically Intensive Codes*


Lloyd N. Trefethen, *Spectral Methods in MATLAB*


Michael W. Berry and Murray Browne, *Understanding Search Engines: Mathematical Modeling and Text Retrieval*

Jack J. Dongarra, Iain S. Duff, Danny C. Sorensen, and Henk A. van der Vorst, *Numerical Linear Algebra for High-Performance Computers*

R. B. Lehoucq, D. C. Sorensen, and C. Yang, *ARPACK Users’ Guide: Solution of Large-Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods*

Randolph E. Bank, *PLTMG: A Software Package for Solving Elliptic Partial Differential Equations, Users’ Guide 8.0*


Greg Astfalk, editor, *Applications on Advanced Architecture Computers*

Françoise Chaitin-Chatelin and Valérie Fraysse, *Lectures on Finite Precision Computations*

Roger W. Hockney, *The Science of Computer Benchmarking*

Richard Barrett, Michael Berry, Tony F. Chan, James Demmel, June Donato, Jack Dongarra, Victor Eijkhout, Roldan Pozo, Charles Romine, and Henk van der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*


Jack J. Dongarra, Iain S. Duff, Danny C. Sorensen, and Henk van der Vorst, *Solving Linear Systems on Vector and Shared Memory Computers*

Louis Komzsik
Schaeffer Automated Simulation, LLC
Altadena, California

THE LANCZOS METHOD
Evolution and Application

Society for Industrial and Applied Mathematics
Philadelphia
To Stella and Victor
This page intentionally left blank
Contents

Preface xi

I  EVOLUTION 1

1  The classical Lanczos method 3
   1.1 The eigenvalue problem ........................................ 3
   1.2 The method of minimized iterations .......................... 4
   1.3 Calculation of eigenvalues and eigenvectors ................. 6
   1.4 Geometric interpretation ..................................... 8

2  The Lanczos method in exact arithmetic 9
   2.1 Computational formulation .................................. 9
   2.2 Solving the tridiagonal problem .............................. 11
   2.3 Exact arithmetic algorithm ................................. 12
   2.4 Computational example .................................... 13

3  The Lanczos method in finite precision 17
   3.1 Breakdown of the Lanczos process ......................... 17
   3.2 Measuring and maintaining orthogonality .................. 18
   3.3 Monitoring convergence and estimating accuracy .......... 19
   3.4 Finite precision algorithm ............................... 20

4  Block real symmetric Lanczos method 23
   4.1 The block Lanczos method ................................ 23
   4.2 Measuring and maintaining block orthogonality .......... 25
   4.3 Reduction of block tridiagonal form ...................... 26
   4.4 Block real symmetric algorithm .......................... 26

5  Block unsymmetric Lanczos method 29
   5.1 Block biorthogonal Lanczos method ....................... 29
   5.2 Solution of the block tridiagonal problem ................. 30
   5.3 Error analysis ........................................... 30
   5.4 Block unsymmetric algorithm ............................ 31
   5.5 Adapting the block size ................................. 32
5.6 Preventing breakdown ........................................ 33
5.7 Maintaining biorthonormality .......................... 34

II APPLICATIONS ......................................................... 37

6 Industrial implementation of the Lanczos method .......................... 39
6.1 Spectral transformation .................................. 39
6.2 Frequency domain decomposition ....................... 41
6.3 Geometric domain decomposition ...................... 42
   6.3.1 Matrix partitioning ..................................... 43
   6.3.2 Partitioned matrix decomposition ................. 43
   6.3.3 Partitioned substitution .......................... 44
   6.3.4 Partitioned Lanczos step ......................... 45
6.4 Parallel computational strategy ....................... 45

7 Free undamped vibrations ........................................ 47
7.1 Analysis of mechanical systems ....................... 47
7.2 Generalized linear eigenvalue problem .................. 49
7.3 Normal modes analysis application .................... 50

8 Free damped vibrations ............................................ 53
8.1 Generalized quadratic eigenvalue problem ............. 53
8.2 Recovery of physical solution ......................... 54
8.3 Orthogonality diagnostics ............................. 56
8.4 Implicit operator multiplication ..................... 57
8.5 Implicit operator algorithm ............................ 59
8.6 Complex eigenvalue analysis application ............. 60

9 Forced vibration analysis ........................................ 63
9.1 The interior acoustic problem ......................... 63
9.2 Fluid-structure interaction ............................ 65
9.3 Coupled forced vibration problem ..................... 66
9.4 Padé approximation via the Lanczos method ......... 67
   9.4.1 Transfer function calculation ................. 67
   9.4.2 Approximate transfer function .......... 68
9.5 Acoustic optimization application ..................... 69

10 Linear systems and the Lanczos method .................... 71
10.1 Exact solution .................................................. 71
10.2 Approximate solution ...................................... 72
10.3 Recursive approximate solution ....................... 73
10.4 Lanczos linear solver algorithm ..................... 74
10.5 Linear static analysis application ..................... 75
Contents

Closing Remarks 77
A Brief Biography of Cornelius Lanczos 79
Bibliography 81
Index 85
Preface

The subject of this book, the method of Lanczos, is probably one of the most influential methods of computational mathematics in the second half of the 20th century. Since Lanczos's seminal paper [2] in 1950, despite some early setbacks about the applicability of the method in computers with finite precision arithmetic, the method found its way into many aspects of science and engineering. The applications are so widespread that it is practically impossible to describe them in a single book. This book follows the evolution of the method as it became more and more established and understood, and began to solve a wide variety of engineering analysis problems.

My personal involvement with and admiration of the method started in the early 1970s in Budapest as a graduate student at the successor of Lanczos's alma mater. While at that time both Lanczos and his method had somewhat tarnished reputations, for political and numerical reasons, respectively, I was taken by the beauty of the three-member recurrence. The second half of the 1970s saw the restoration of the numerical reputation of the method worldwide, and by the end of the decade Lanczos was also put on his well-deserved pedestal, even in Hungary.

The material in this book comes from seminars and lectures I had given on the topic during the past two decades. The seminars, held by leading corporations of the automobile and aerospace industries in the United States, Europe, and Asia, were attended by engineers and computer scientists and focused on applications of the method in commercial finite element analysis, specifically in structural analysis. The lectures I had given recently as a SIAM Visiting Lecturer at various academic institutions were attended by both faculty and students and centered on practical implementation and computational performance issues. The interest of the audience in both camps and the lack of a text encompassing the evolution of the method contributed to the decision to write this book. I hope that the readers share this interest, enjoy a brief travel of time through the history of the method, and find the book useful in their applications.

The book has two distinct parts. The first part, Chapters 1 through 5, demonstrates the evolution of the method from the review of Lanczos's original method to the state-of-the-art adaptive methods. The second part, Chapters 6 through 10, addresses the practical implementation and industrial application of the method. Specifically, in Chapters 7, 8, and 9 the well-established industrial applications of normal modes and complex eigenvalue analyses, as well as the frequency response analysis, are discussed. The book concludes with the application of the Lanczos method for the solution of linear systems.

While heavy on mathematical content, in order to achieve readability, rigorous statement of theorems and proofs are omitted. Similarly, topics in the linear algebraic foundation
(QR and singular value decomposition, Givens rotations, etc.) are not discussed in detail to keep the focus sharp. Several chapters contain a computational algorithm enabling the reader to implement some of the methods either in a MATLAB environment or in a high-level programming language.

During the past quarter century I have cooperated with many people in various aspects of the Lanczos method. I specifically thank Prof. Beresford Parlett of UC Berkeley and Prof. Gene Golub of Stanford University for their most valuable theoretical influence, which I enjoyed from their books as well as personal contacts. I am also very much indebted to Dr. Horst Simon of Berkeley National Laboratory, Dr. John Lewis of Boeing, and Prof. Zhaojun Bai of UC Davis for their very important cooperation in the practical implementation aspects of the Lanczos method. Finally, special thanks are due to my colleague, Dr. Tom Kowalski, who, besides participating in the implementation of some of the methods mentioned in this book into NASTRAN, has also dutifully proofread the manuscript and provided valuable corrections and suggestions.

Louis Komzsik
2002
Part I

EVOLUTION
This page intentionally left blank
At the time of Lanczos’s work on the eigenvalue problem during the Second World War, most methods focused on finding the characteristic polynomial of matrices in order to find their eigenvalues. In fact, Lanczos’s original paper [2] was also mostly concerned with this problem; however, he was trying to reduce the round-off errors in such calculations. He called his method the method of minimized iterations, which we will now review to lay the foundation.

### 1.1 The eigenvalue problem

For a real, square matrix $A$ of order $n$, the product

$$x^T A x = \sum_{k=1}^{n} \sum_{l=1}^{n} a_{kl} x_k x_l$$

defines a quadratic form. This is a continuous function of $x = (x_1, x_2, \ldots, x_n)$. When $A$ is symmetric positive definite, the equation

$$x^T A x = 1$$

defines an $n$-dimensional ellipsoid in $\mathbb{R}^n$ which is in all likelihood rotated. The eigenvalue problem is to find the $n$ principal axes of this ellipsoid which are the eigenvectors $x_i, i = 1, \ldots, n$. The square roots of the lengths of the principal axes are the eigenvalues $\lambda_i$. They satisfy the equation

$$A x_i = \lambda_i x_i.$$  

This is easy to verify considering the fact that the directions of the principal axes of the ellipsoid are where the surface normal $\vec{n}$ is colinear with the principal axis location vector pointing to the surface $x_i$,

$$x_i = c \vec{n},$$

where $c$ is a scalar constant. Since the normal points into the direction of the gradient,

$$\vec{n} = \nabla (x^T A x - 1) = A x_i,$$
it follows that $c$ is $1/\lambda_i$.

### 1.2 The method of minimized iterations

Lanczos first considered symmetric matrices ($A^T = A$) and set out to find the characteristic polynomial of

$$G(\mu) = \det(A - \mu I) = 0$$

(1.6)

in order to solve

$$Au = \mu u,$$

(1.7)

where $u$ is an eigenvector and $\mu$ is the corresponding eigenvalue. In deference to Lanczos, in this section we adhere to his original notation as much as possible. Specifically, the inner products commonly written as $b_0^T b_0$ in today's literature is noted below as $b_0^2$.

Lanczos sought the characteristic polynomial by generating a sequence of trial vectors, resulting in a successive set of polynomials. Starting from a randomly selected vector $b_0$, the new vector $b_1$ is chosen as a certain linear combination of $b_0$ and $Ab_0$,

$$b_1 = Ab_0 - \alpha_0 b_0.$$  

(1.8)

Here the parameter $\alpha_0$ is found from the condition of $b_1$ having as small magnitude as possible:

$$b_1^2 = (Ab_0 - \alpha_0 b_0)^2 = \text{min}.$$  

(1.9)

Differentiation and algebra yields

$$\alpha_0 = \frac{(Ab_0)b_0}{b_0^2}. $$  

(1.10)

It is important to notice that the new $b_1$ vector is orthogonal to the original $b_0$ vector, i.e.,

$$b_1 b_0 = 0.$$  

(1.11)

Continuing the process, one can find a $b_2$ vector by choosing the linear combination

$$b_2 = Ab_1 - \alpha_1 b_1 - \beta_0 b_0,$$  

(1.12)

where once again the constants are defined by the fact that $b_2^2$ shall be minimal. Some algebraic work yields

$$\alpha_1 = \frac{(Ab_1)b_1}{b_1^2}, \quad \beta_0 = \frac{(Ab_1)b_0}{b_0^2}. $$  

(1.13)

Since

$$(Ab_1)b_0 = b_1 (Ab_0) = b_1^2,$$  

(1.14)

the new $b_2$ vector is orthogonal to both $b_1$ and $b_0$. Once more continuing the process, we need

$$b_3 = Ab_2 - \alpha_2 b_2 - \beta_1 b_1 - \gamma_0 b_0.$$  

(1.15)
However, in view of the orthogonality of $b_2$ to both previous vectors, we get

$$v_0 = \frac{(Ab_2)b_0}{b_2^*b_0} = \frac{b_2(\alpha_b)}{b_2^*b_0} = 0. \quad (1.16)$$

The brilliant observation of Lanczos is that in every step of the iteration we will need only two correction terms: the famous three-member recurrence. The process established by Lanczos is now

$$b_0 = \text{random},$$

$$b_1 = (A - \alpha_0)b_0,$$

$$b_2 = (A - \alpha_1)b_1 - \beta_0b_0,$$

$$b_3 = (A - \alpha_2)b_2 - \beta_1b_1,$$

$$\cdots$$

$$b_m = (A - \alpha_{m-1})b_{m-1} - \beta_{m-2}b_{m-2} = 0.$$  

The equality to zero on the $m$th recurrence equation means the end of the process. In Lanczos’s view we reached the order of the minimum polynomial, where

$$m \leq n,$$

and $n$ is the order of the matrix $A$. Unfortunately, in finite precision arithmetic, the process may reach a state where $\beta_k$ is very small for $k < m$ before the full order of the minimum polynomial is obtained. This phenomenon, at the time not fully understood, contributed to the method’s bad numerical reputation in the 1960s.

Lanczos then applied the method to unsymmetric matrices by the simultaneous execution of the process to $A$ and its transpose, $A^T$. The connection of the two sets of vectors is maintained via inner products when calculating the denominators of the constants. This so-called biorthogonal process starts from random vectors $b_0$ and $b_0^*$. Please note that $b_0^* \neq b_0^T$; it is just another starting vector. The first step produces

$$b_1 = Ab_0 - \alpha_0b_0, \quad (1.17)$$

$$b_1^* = A^Tb_0^* - \alpha_0b_0^*, \quad (1.18)$$

where the value of $\alpha_0$ satisfying the biorthogonality conditions is

$$\alpha_0 = \frac{(Ab_0)b_0^*}{b_0b_0^*} = \frac{(A^Tb_0^*)b_0}{b_0^*b_0}. \quad (1.19)$$

The second step brings

$$b_2 = Ab_1 - \alpha_1b_1 - \beta_0b_0, \quad (1.20)$$

$$b_2^* = A^Tb_1^* - \alpha_1b_1^* - \beta_0b_0^*, \quad (1.21)$$

where

$$\alpha_1 = \frac{(Ab_1)b_1^*}{b_1b_1^*} = \frac{(A^Tb_1^*)b_1}{b_1^*b_1}. \quad (1.22)$$
and
\[ \beta_0 = \frac{(Ab_1)b_0^*}{b_1b_0^*} = \frac{(A^Tb_1^*)b_0}{b_0^*b_0} = \frac{b_1^*b_1}{b_0^*b_0}. \]

The process now may be continued, leading to the following polynomials:

\[
\begin{align*}
p_0 &= 1, \\
p_1(\mu) &= \mu - \alpha_0, \\
p_2(\mu) &= (\mu - \alpha_1)p_1(\mu) - \beta_0p_0(\mu), \\
p_n(\mu) &= (\mu - \alpha_{n-1})p_{n-1}(\mu) - \beta_{n-2}p_{n-2}.
\end{align*}
\]

For simplicity, let us assume now that all the polynomials until the \( n \)th may be obtained by this process (none of the \( \beta_i \) vanish). Then \( p_n \) is the characteristic polynomial of \( A \) with roots \( \mu_i, i = 1, 2, \ldots, n \).

### 1.3 Calculation of eigenvalues and eigenvectors

Lanczos used the characteristic polynomial developed above and the biorthogonality of the \( b_i, b_i^* \) sequence to find an explicit solution for the eigenvectors in terms of the \( b_i, b_i^* \) vectors. Assuming that \( A \) is of full rank, the \( b_i \) vectors may be expressed as linear combinations of the eigenvectors

\[ b_i = p_i(\mu_1)u_1 + p_i(\mu_2)u_2 + \cdots + p_i(\mu_n)u_n. \]

Taking an inner product with \( u_k^* \) which is orthogonal to the \( u_i \) vectors, we get

\[ b_iu_k^* = p_i(\mu_k)u_ku_k^*, \]

since all other inner products vanish. Please note again that \( u_k^* \neq u_k^H \); they are just the \( k \)th members of the two simultaneous sequences. The reverse process of expressing eigenvectors in terms of the \( b_i \) vectors yields

\[ u_i = \alpha_{i,0}b_0 + \alpha_{i,1}b_1 + \cdots + \alpha_{i,n-1}b_{n-1}. \]

Taking inner products again yields

\[ u_i b_k^* = \alpha_{i,k}b_k b_k^*, \]

or

\[ \alpha_{i,k} = \frac{u_i b_k^*}{b_k b_k^*}. \]

Using this, the expansion for the eigenvectors becomes

\[ u_i = \frac{b_0}{b_0 b_0^*} + \frac{p_1(\mu_i)}{b_1 b_1^*} + \cdots + \frac{p_{n-1}(\mu_i)}{b_{n-1} b_{n-1}^*}. \]
The adjoint, or left-handed, eigenvectors are calculated similarly:

\[
u_i^* = \frac{b_0^*}{b_0 b_0^*} + p_1(\mu_i) \frac{b_1^*}{b_1 b_1^*} + \cdots + p_{n-1}(\mu_i) \frac{b_{n-1}^*}{b_{n-1} b_{n-1}^*}.
\]  

(1.30)

In the case of rank deficiency, the expansion is still valid for \( m \leq n \). The shortcoming of this method is the repeated calculation and evaluation of the characteristic polynomial. Note that explicit formulation of the polynomial is not necessary. It is easy to see that premultiplying \( A \) by \( B^*T \) and postmultiplying the product by \( B \) yields

\[T = B^*T AB,
\]

(1.31)

where

\[
B = \begin{bmatrix}
b_0 & b_1 & \cdots & b_{n-1}
\end{bmatrix},
\]

(1.32)

\[
B^* = \begin{bmatrix}
b_0^* & b_1^* & \cdots & b_{n-1}^*
\end{bmatrix},
\]

(1.33)

and

\[
T = \begin{bmatrix}
\alpha_0 & \beta_0 & & & \\
\beta_0 & \alpha_1 & \beta_1 & & \\
& \ddots & \ddots & \ddots & \\
& & \beta_{n-2} & \alpha_{n-1} & \beta_{n-1} \\
& & & \beta_{n-1} & \alpha_n
\end{bmatrix}.
\]

(1.34)

This observation allows a more efficient eigenvector calculation scheme. With appropriate pre- and postmultiplications of the problem (1.7),

\[B^*T AB B^*T u = \mu B^*T u,
\]

(1.35)

and using the biorthogonality property of the \( b \) vectors,

\[BB^*T = I,
\]

(1.36)

we get

\[Tv = \mu v,
\]

(1.37)

where

\[v = B^*T u, \quad u = Bv.
\]

(1.38)

The latter two equations propose to calculate the \( v \) eigenvectors of the \( T \) tridiagonal matrix and calculate the eigenvectors of the original matrix with the multiplication by the Lanczos vectors, a process still in principal use.