CONVERGENCE ACCELERATION IN NUMERICAL METHODS
CONVERGENCE ACCELERATION IN THE NUMERICAL SOLUTION OF FIELD PROBLEMS

by

WITOLD KINSNER, M.Sc.(Eng.)

A Thesis Submitted to the Faculty of Graduate Studies in Partial Fulfilment of the Requirements for the Degree Doctor of Philosophy

McMaster University March 1974
DOCTOR OF PHILOSOPHY (1974)  
(Electrical Engineering)  
McMASTER UNIVERSITY  
Hamilton, Ontario.

TITLE : Convergence Acceleration in the Numerical Solution of Field Problems

AUTHOR : Witold Kinsner  
M.Sc.(Eng.) (University of Wroclaw)

SUPERVISOR : Professor E. Della Torre

NUMBER OF PAGES:

SCOPE AND CONTENTS: xv, 303.

The purpose of this thesis is to present a unified treatment of numerical solutions to the elliptic partial differential equations, particularly the Laplace, Poisson, and Helmholtz equations, in homogeneous and inhomogeneous isotropic media in electrostatic, magnetostatic, and electromagnetic field problems, with particular attention given to iterative solutions. Another objective is to design a family of methods for accelerating the convergence of these solutions. The concept of the convergence acceleration is generalized to deterministic and stochastic vector sequences. Several hybrid methods, combining the finite difference and finite element approaches, are proposed. The methods have been tested either by test examples or by practical solutions to field problems.
To the memory of
my Parents
PREFACE

This thesis is basically centered around the theoretical and practical implementation of a family of methods for accelerating the convergence of vector sequences which arise in various methods of successive approximations and, in particular, in the iterative solutions of elliptic partial differential equations such as the Laplace, Poisson, and Helmholtz equations associated with field problems. This part of the study may yield more economical solutions to many problems. The secondary objective is to introduce some new or improved methods of solving field problems.

The chapters have been organized so that the introductory material is condensed in the opening two chapters which are followed by the main development of the thesis, and closed by numerical examples.

Chapter 1 is devoted to the most important methods of solving field problems. In view of the number of textbooks and articles published in this area in recent years, it would be unjustifiable to give here a detailed description of the methods. The presentation utilizes a direct-analysis point of view, i.e., it begins with Maxwell's equation and specializes to static fields, rather than a synthesis procedure which builds up and generalizes the experimental laws for static fields to time-varying fields. Approximation of partial differential equations by sets of simultaneous equations is briefly discussed. A classification of errors in numerical methods and the concepts of convergence, stability and consistency is presented.
Chapter 2 presents a complete finite difference formulation. It discusses various discretization techniques by meshes, various difference operators, boundary conditions, singularities and the modification of operators in inhomogeneous media. Solutions of the matrix equation, although connected with the finite difference approximation, refer to many other methods; therefore, a detailed consideration is given to various direct and iterative methods. The discussion of the iterative methods includes all the stationary and some nonstationary techniques, both point and block families of the methods. The most important techniques of computing eigenvalues and eigenvectors from the matrix eigenvalue equation are presented. Scanning techniques used in iterative methods are discussed.

Although the material of Chapters 1 and 2 could be incorporated in the remaining chapters, its grouping into separate units was considered more useful for further reference.

Chapter 3 develops and generalizes basic concepts of deterministic, related stochastic sequences with geometric and linear convergence. It shows the conditions under which the sequences, generated by various iterative methods, converge linearly. A new algorithm for accelerating the convergence is presented. An explicit analysis is given for the Aitken $\delta^2$-process and the new algorithm. A sensitivity analysis shows the upper and lower bounds on the generalized perturbation matrix and the extrapolation factor. A family of extrapolation methods is designed to improve the basic algorithm. Finally, higher order nonlinear transformations for vector and scalar sequences are presented.

Chapter 4 deals with an iterative approach to the finite element
methods that are ordinarily solved by various direct techniques. The iterative formulation leads to sequences whose convergence may be improved by the techniques discussed in the previous chapters. Extrapolation methods based upon element subdivision are given. A simple analysis yields explicit formulae for the discretization error in the finite element methods. The residual error and error criteria are also discussed.

Chapter 5 reports numerical solutions to various test and practical field problems. Other problems are also illustrated by numerical examples. The Chapter closes with some essential programming remarks regarding the methods of Chapter 3.

Appendices are included to supplement the text with certain material that had to be left out otherwise.
ACKNOWLEDGEMENTS

First and foremost I would like to express my sincere gratitude to Dr. E. Della Torre for having made this work a pleasurable and bilateral exchange of information. My special thanks go to Dr. J.W. Bandler for his helpful advice, encouragement, and discussions. Appreciation is also expressed to Dr. E.A. Ballik for his interest in the work.

I would like to thank Dr. C.M. Crowe, Dr. N.K. Sinha, Dr. J.J. Emery, and Dr. A. Rosa for their comments concerning some mathematical aspects of this work. Thanks are also due to Dr. P. Silvester and Dr. A. Wexler for their brief comments on the convergence acceleration concept. Discussions with Dr. C. Charalambous are acknowledged.

Appreciation is expressed to the National Research Council of Canada and McMaster University for their financial support.

I am very grateful to my friends for their encouragement and patience.

Finally, I would like to thank Mrs. B. Pare, Mrs. R. Maxwell, and Miss A. Hooker for the help in typing the manuscript.
TABLE OF CONTENTS

PREFACE

CHAPTER 1 - FIELD EQUATIONS AND THEIR SOLUTIONS

1.1 - Fundamental Equations
1.2 - Field-Vector Wave Equations
1.3 - Space-Function Vector Helmholtz Equation
1.4 - Vector and Scalar Helmholtz Equations
1.5 - Poisson's Equation
1.6 - Laplace's Equation
1.7 - Classification of Second-Order Partial Differential Equations
1.8 - Initial and Boundary Conditions. Uniqueness
1.9 - Analytical and Numerical Methods of Solution
  1.9.1 - Separation of Variables
  1.9.2 - Conformal Transformation
  1.9.3 - Variational Methods
  1.9.4 - Perturbational Methods
  1.9.5 - Finite Difference Methods
  1.9.6 - Finite Element Methods
  1.9.7 - Integral Equation Methods
  1.9.8 - Null Field Method
  1.9.9 - Point Matching Methods
  1.9.10 - Other Methods
1.10 - Large Systems of Linear Equations
1.11 - Matrix Eigenvalue Problem
1.12 - Errors in Numerical Methods
1.13 - Convergence, Stability, Consistency
1.14 - Summary
1.15 - Remarks

CHAPTER 2 - FINITE DIFFERENCE FORMULATION

2.1 - Introduction
2.2 - Discretization
  2.2.1 - The Lattice
  2.2.2 - The Mesh
  2.2.3 - Regular and Near-Boundary Interior Points
  2.2.4 - Mesh Cells
  2.2.5 - Difference Operators
  2.2.6 - Treatment of Boundary Conditions
  2.2.7 - Choice of Networks
  2.2.8 - Singularities
  2.2.9 - Inhomogeneous Media

Page

1
3
4
7
8
8
9
9
11
12
12
13
15
16
16
17
18
19
19
20
21
22
24
25
27
29
29
30
30
32
32
33
35
38
41
41
CHAPTER 2 - Continued

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.3 - Solution of Elliptic Difference Equations</td>
<td></td>
</tr>
<tr>
<td>2.3.1 - Matrix Equation and Its Properties</td>
<td>47</td>
</tr>
<tr>
<td>2.4 - Direct Methods</td>
<td></td>
</tr>
<tr>
<td>2.4.1 - Gaussian Elimination (Pivotal Condensation)</td>
<td>52</td>
</tr>
<tr>
<td>2.4.2 - Choleski Triangular Decomposition</td>
<td>53</td>
</tr>
<tr>
<td>2.4.3 - Gauss-Jordan Elimination</td>
<td>55</td>
</tr>
<tr>
<td>2.5 - Iterative Methods</td>
<td></td>
</tr>
<tr>
<td>2.5.1 - Point Iterative Methods</td>
<td>56</td>
</tr>
<tr>
<td>2.5.1.1 - Jacobi Method</td>
<td>56</td>
</tr>
<tr>
<td>2.5.1.2 - Jacobi Overrelaxation Method (JOR)</td>
<td>61</td>
</tr>
<tr>
<td>2.5.1.3 - Gauss-Seidel Method</td>
<td>62</td>
</tr>
<tr>
<td>2.5.1.4 - Successive Overrelaxation Method (SOR)</td>
<td>63</td>
</tr>
<tr>
<td>2.5.1.5 - Other Point Iterative Methods</td>
<td>65</td>
</tr>
<tr>
<td>- Gradient Methods</td>
<td>65</td>
</tr>
<tr>
<td>- Richardson's Method</td>
<td>66</td>
</tr>
<tr>
<td>- Semi-Iterative Methods</td>
<td>68</td>
</tr>
<tr>
<td>- Symmetric SOR (SSOR) Method</td>
<td>69</td>
</tr>
<tr>
<td>2.5.1.6 - Optimum Relaxation Factor</td>
<td>70</td>
</tr>
<tr>
<td>2.5.1.7 - Scanning in SOR</td>
<td>74</td>
</tr>
<tr>
<td>2.5.2 - Group and Block Iterative Methods</td>
<td>74</td>
</tr>
<tr>
<td>2.5.2.1 - Line Iterative Methods</td>
<td>81</td>
</tr>
<tr>
<td>2.5.2.2 - Alternating Direction Implicit (ADI) Method</td>
<td>81</td>
</tr>
<tr>
<td>2.6 - Matrix Eigenvalue Equation</td>
<td></td>
</tr>
<tr>
<td>2.6.1 - Direct Methods</td>
<td></td>
</tr>
<tr>
<td>2.6.1.1 - Jacobi Algorithm</td>
<td>85</td>
</tr>
<tr>
<td>2.6.1.2 - Given's Algorithm</td>
<td>86</td>
</tr>
<tr>
<td>2.6.1.3 - Householder's Algorithm</td>
<td>88</td>
</tr>
<tr>
<td>2.6.1.4 - Sturm Sequence and Bisection</td>
<td>89</td>
</tr>
<tr>
<td>2.6.1.5 - QR Algorithm</td>
<td>89</td>
</tr>
<tr>
<td>2.6.1.6 - Inverse (Wilandt) Iteration</td>
<td>93</td>
</tr>
<tr>
<td>2.6.2 - Iterative Methods</td>
<td>96</td>
</tr>
<tr>
<td>2.7 - Summary</td>
<td>99</td>
</tr>
</tbody>
</table>

CHAPTER 3 - ACCELERATION OF ITERATIVE PROCESSES

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1 - Introduction</td>
<td>100</td>
</tr>
<tr>
<td>3.2 - Deterministic and Related Stochastic Sequences</td>
<td>101</td>
</tr>
<tr>
<td>3.3 - Geometric and Linear Convergent Sequences</td>
<td>104</td>
</tr>
<tr>
<td>3.4 - Physical Problems that Lead to Geometrically Convergent Sequences</td>
<td>105</td>
</tr>
<tr>
<td>3.5 - Successive Extrapolated Relaxation (SER) Method</td>
<td></td>
</tr>
<tr>
<td>3.5.1 - Derivation of Basic Formula</td>
<td>110</td>
</tr>
<tr>
<td>3.5.2 - Linear Convergence of Linear Iterative Methods</td>
<td>111</td>
</tr>
<tr>
<td>3.5.3 - Convergence Analysis of Aitken's $\delta^2$ Process and SER</td>
<td>118</td>
</tr>
</tbody>
</table>
CHAPTER 6 - CONCLUSIONS

BIBLIOGRAPHY

APPENDICES

1 - Some Second Order PDE with Two Independent Variables 266
2 - Finite Difference Approximations on Various Meshes 267
3 - Molecules for the Operators $\nabla^2$, $\nabla^4$, and $(\nabla^2 + k^2)$ 271
4 - Comparison of the Number of Arithmetic Operations in Various Methods 279
5 - Comparison of Convergence Rates of Iterative Methods 280
6 - Storage Requirement for Solving $Ax = b$ with $n \times n$ Matrix 282
7 - The Carré Algorithm 283
8 - Spiral and Contour Scannings 286
9 - The Thomas Algorithm for a Tridiagonal Matrix 287
10 - Threshold Jacobi Diagonalization Algorithm for Eigenvalues and Eigenvectors 288
11 - The Aitken and Higher Order Transforms 290
12 - Various Forms of Aitken's Formula 297
13 - $T_L$ Transform (Lyusternik's Approach) 301
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>(a) Continuous domain ( \Omega = R \cup G ); (b) Discrete representation ( \tilde{R} = \tilde{R} \cup \tilde{F} ) of ( \Omega ): ( x ) - for regular interior nodes ( \tilde{R} ), ( \bigcirc ) - for near-boundary nodes ( \tilde{R}_1 ), ( \bigtriangledown ) - for boundary points ( \Gamma^+ ), ( \bigcirc ) - for ( \Gamma_1^+ ), ( \square ) - for ( \Gamma_2^- ), ( \bigtriangledown ) - for ( \Gamma_2^+ ).</td>
<td>31</td>
</tr>
<tr>
<td>2.2</td>
<td>Mesh cells: (a) for a regular interior node, and (b) for a near-boundary node.</td>
<td>33</td>
</tr>
<tr>
<td>2.3</td>
<td>Five-point operator.</td>
<td>35</td>
</tr>
<tr>
<td>2.4</td>
<td>Polygonal interpolation of Dirichlet boundary conditions: (a) internal, (b) external, and (c) averaging.</td>
<td>37</td>
</tr>
<tr>
<td>2.5</td>
<td>Normal derivative on rectilinear boundary.</td>
<td>38</td>
</tr>
<tr>
<td>2.6</td>
<td>Graded mesh.</td>
<td>40</td>
</tr>
<tr>
<td>2.7</td>
<td>Re-entrant corner with ( \alpha &gt; \pi ).</td>
<td>41</td>
</tr>
<tr>
<td>2.8</td>
<td>Regular interior node ( 0 ) with six adjacent nodes.</td>
<td>43</td>
</tr>
<tr>
<td>2.9</td>
<td>General three-dimensional finite difference Laplacian inside inhomogeneous media (Eq. 2.24).</td>
<td>45</td>
</tr>
<tr>
<td>2.10</td>
<td>Three-dimensional Laplacian on discrete rectilinear interfaces.</td>
<td>46</td>
</tr>
<tr>
<td>2.11</td>
<td>Finite difference operators with Property A can be constructed on the two classes of points, ( 0 ) and ( x ); (a) 5- and 13-point operators with Property A, (b) 9- and 13-point operators without Property A.</td>
<td>51</td>
</tr>
<tr>
<td>2.12</td>
<td>Spectral radius ( \rho(L_\omega) ) as a function of ( \omega ) for various ( \rho(J) ).</td>
<td>72</td>
</tr>
<tr>
<td>2.13</td>
<td>Convergence of a potential value at the point ( (8, 13) ) of the problem shown in Fig. 2.14, for different scanning methods.</td>
<td>76</td>
</tr>
<tr>
<td>2.14</td>
<td>Model problem associated with Fig. 2.13.</td>
<td>76</td>
</tr>
<tr>
<td>2.15</td>
<td>Raster (natural) scan.</td>
<td>77</td>
</tr>
<tr>
<td>2.16</td>
<td>Spiral scan.</td>
<td>78</td>
</tr>
<tr>
<td>2.17</td>
<td>Double spiral scan.</td>
<td>79</td>
</tr>
</tbody>
</table>

xii
Figure

2.18 Optimal contour scan. 80

2.19 Arbitrary Sturm sequence $P_i(\lambda)$ for $i=0,1,2,3$. 90

3.1 Aitken's extrapolation of (a) convergent, (b) divergent, and (c) oscillating sequence. 106

3.2 Typical potential sequence for given boundary conditions. 107

3.3 Potential sequence due to a single boundary point. 109

3.4 Convergent sequence of iterates. 112

3.5 Hard and soft limiting of extrapolation. 115

3.6 Forced extrapolation. 117

3.7 Convergence improvement using Aitken's transform to linearly convergent sequences when $\alpha = 0.5$. 126.

3.8 Convergence improvement using Aitken's transform for general $\alpha$. 127

3.9 Convergence improvement using Aitken's transform for $a_i^{(n-2)} = 0$. 128

3.10 Convergence improvement using SER ($\kappa = 7/6$) for linearly convergent sequences. 130

3.11 Convergence improvement using optimized SER on linearly convergent sequences. 131

3.12 Error spectrum for weighted SER. 143

3.13 Sequences as mathematical transients. 147

3.14 Third-order nonlinear transformation based on $2k+1=7$ elements of $\{x^{(n)}\}$. 147

4.1 A general nonconvex two-dimensional multiply connected region with two types of boundary, placed in global coordinates $X,Y$. Triangulation of the region with $M$ elements and $N$ global nodes. Local coordinates $x,y$ are connected with an element $E_e$. 161

xiii
4.2 One of m triangles adjacent to a node $P_0$. The origin of local coordinates lies on $P_1P_2$.

4.3 Five-point Laplacian operators resulting from various triangulations of the region: (a) two types of nodes with 4 adjacent triangles, and 8 adjacent triangles; (b) one type of nodes with 6 adjacent triangles.

4.4 One of m triangles adjacent to a node $P_0$, placed in general local coordinates $x,y$.

4.5 Seven-point finite element operator for the dominant mode solution of Helmholtz's equation.

4.6 A 17-point finite difference operator. It is positive definite, has Property A and improved diagonal dominance.

4.7 Element subdivision by bisection of its sides.

4.8 Triangular element in general coordinates (a), and related hexagon (b).

5.1 Dirichlet boundary value problem.

5.2 Dirichlet and Neumann boundary value problem.

5.3 Boundary conditions for the dominant TM mode in a ridge waveguide. Only one half of the symmetrical guide is shown.

5.4 The dominant TM mode in a symmetrical ridge waveguide shown in Fig. 5.5.

5.5 Placement of boundary data for a circular quadrant of radius 3.0. Characteristic dimension $a = 3.0$. The TE$_{31}$ mode with $k_a = 4.1774$ generated by the boundary conditions is indicated with the dashed lines.

5.6 T-bar two-dimensional bubble propagating Permalloy circuit.

5.7 Y-bar structure.

5.8 Chevron circuit.

5.9 Sequence of computer plots showing normalized z-averaged z-component of magnetic field intensity in bubble crystals.
Figure 5.10: Same as Fig. 5.9 but for Y-bar propagating structure. 227
Figure 5.11: Same as Fig. 5.9 but for Chevron propagating structure. 228
Figure 5.12: Sample of boundary conditions and spiral scan (Problem 1). 230
Figure 5.13: Sample of boundary conditions and contour scan (Problem 2). 231
Figure 5.14: Nearly-geometric convergence of $\gamma$. 234
Figure 5.15: Nearly-geometric convergence of errors $|e_\gamma|$ and $|e_{\gamma\gamma}|$. 235

Tables
Table 4.1: A flow diagram of an iterative scheme. 179
Table 5.1: Comparison between various finite difference methods. 211
Table 5.2: Number of iterations required to reduce the maximum error to 0.1% for various scans. 232
CHAPTER 1
FIELD EQUATIONS AND THEIR SOLUTIONS

1.1 Fundamental Equations

Electromagnetic fields can be described by the four fundamental Maxwell equations

\[ \nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \]  \tag{1.1}

\[ \nabla \times \mathbf{H} = \mathbf{J} + \frac{\partial \mathbf{D}}{\partial t} \] \tag{1.2}

\[ \nabla \cdot \mathbf{D} = \rho \] \tag{1.3}

\[ \nabla \cdot \mathbf{B} = 0 \] \tag{1.4}

This set of equations contains four field vectors: the electric field strength \( \mathbf{E} \), the magnetic field strength \( \mathbf{H} \), the electric displacement \( \mathbf{D} \), the magnetic induction \( \mathbf{B} \), and two source variables: the conduction current density \( \mathbf{J} \), and the electric charge density \( \rho \). \( \partial \mathbf{D}/\partial t \) and \( \partial \mathbf{B}/\partial t \) are called the displacement current and the magnetic decay, respectively. These equations are valid for media in rest. The four equations are not entirely independent of one another; they become, however, a complete system in which the temporal course of an electromagnetic phenomenon can be uniquely calculated if three other constitutive equations are introduced

\[ \mathbf{D} = \varepsilon \mathbf{E} \] \tag{1.5}

\[ \mathbf{B} = \mu \mathbf{H} \] \tag{1.6}

\[ \mathbf{J} = \sigma (\mathbf{E} + \mathbf{E}^\prime) \] \tag{1.7}

where \( \varepsilon = \varepsilon_r \varepsilon_0 \) is the electric permittivity, \( \mu = \mu_r \mu_0 \) is the
magnetic permeability, \( \sigma \) is the electric conductivity, and \( \mathbf{E}^I \) is the "impressed" field strength [Becker and Sauter (1964) p. 149]. The three equations encounter the special properties displayed by isotropic matter in the presence of a field; they are not, however, general and must be modified for nonlinear media and also for high frequencies of the field or after-effect phenomena in polarization and magnetization.

Although the equations (1.5) and (1.6) involving \( \varepsilon \) and \( \mu \) are usually of greater practical value, the following expanded forms are often useful

\[
\begin{align*}
\mathbf{P} & \equiv \varepsilon_0 \mathbf{E} + \mathbf{P} \\
\mathbf{B} & \equiv \mu_0 (\mathbf{H} + \mathbf{M})
\end{align*}
\]  

(1.8)  

(1.9)

where \( \mathbf{P} \) is the dielectric polarization vector, \( \mathbf{M} \) is the magnetization (or magnetic polarization) vector, \( \varepsilon_0 \) is the permittivity of vacuum, and \( \mu_0 \) is the permeability of vacuum.

Equations (1.1) to (1.9) constitute the basis of all electromagnetic theory (nonrelativistic and unquantized) in point form. They are concise and simple in appearance but the solution of a practical problem is generally difficult to obtain. Thus the above equations are often supplemented by additional definitions and transformed from the differential into other forms to fit certain types of problems. For example, sometimes it is easier to use the integral form or the scalar magnetic potential.


\[
\oint \mathbf{E} \cdot \mathbf{d} \mathbf{r} = -\frac{\partial \mathbf{B}}{\partial t}
\]  

(1.10)
\[ \oint_{\Gamma} \mathbf{H} \cdot d\mathbf{r} = i + \frac{\partial \mathbf{F}}{\partial t} \quad (1.11) \]
\[ \oint_{S} \mathbf{D} \cdot d\mathbf{S} = Q \quad (1.12) \]
\[ \oint_{S} \mathbf{B} \cdot d\mathbf{S} = 0 \quad (1.13) \]

where \( \Phi \) is the magnetic flux, \( \mathbf{F} \) is the electric flux, \( i \) is the electric current, \( Q \) is the total charge within a volume bounded by a simply connected surface \( S \), and \( \Gamma \) is the boundary of \( S \). Since the integral expressions specify field distributions in entire regions of space, their physical interpretation often becomes easier compared to the differential form, which refers to the immediate neighborhood of a point.

1.2 Field-Vector Wave Equations

From a physical point of view, Maxwell's equations give a stimulating insight into the behaviour of matter and can lead to quantitative relations within it, viz., to a solution of any electromagnetic problem. Mathematically, four partial differential equations and their boundary conditions must be solved as a simultaneous system. A general method of solving this mathematical problem is to eliminate variables until a single equation in only one field vector is obtained hoping that a direct solution of this equation is possible.

For linear media, this elimination procedure yields the general homogeneous wave equation [Silvester (1968) p. 213]

\[ \nabla^2 \mathbf{H} = \mu \sigma \frac{\partial \mathbf{H}}{\partial t} + \mu \varepsilon \frac{\partial^2 \mathbf{H}}{\partial t^2} \quad (1.14) \]

which also holds for \( \mathbf{E} \), after dividing (1.14) by \( \mu \), or it gives the general inhomogeneous wave equation [Silvester (1968) p. 214]
\[ \nabla^2 \vec{E} = \mu \sigma \frac{\partial \vec{E}}{\partial t} + \mu c \frac{\partial^2 \vec{E}}{\partial t^2} + \varepsilon^{-1} \nabla \cdot \rho, \] 

(1.15)

also satisfied by \( \rho \) when (1.15) is multiplied by \( \varepsilon \). Both equations (1.14) and (1.15) are similar if \( \rho = \text{const} \) for which \( \nabla \cdot \rho = 0 \).

In homogeneous, isotropic, source-free media (1.14) and (1.15) reduce to the lossless wave equations

\[ \nabla^2 \vec{H} = \nu^{-2} \frac{\partial^2 \vec{H}}{\partial t^2} \] 

(1.16)

\[ \nabla^2 \vec{E} = \nu^{-2} \frac{\partial^2 \vec{E}}{\partial t^2} \] 

(1.17)

where \( \nu = (\mu \varepsilon)^{-1/2} \) is the velocity of propagation. For vacuum, equations (1.16) and (1.17) have \( \nu = c = (\mu_0 \varepsilon_0)^{-1/2} \) where \( c \) is the velocity of light in vacuum.

In conductive media where \( \lambda \gg \partial \rho / \partial t \), instead of (1.14) and (1.15), the simplified form can be derived [Silvester (1968) p. 215]

\[ \nabla^2 \vec{H} = \mu \sigma \frac{\partial \vec{H}}{\partial t} \] 

(1.18)

\[ \nabla^2 \vec{E} = \mu \sigma \frac{\partial \vec{E}}{\partial t} \] 

(1.19)

which have the form of the diffusion equation. The flux densities \( \vec{H} \) and \( \rho \) as well as \( \lambda \) obey a similar diffusion equation.

1.3 Space-Function Vector Helmholtz Equation

A general form of this important equation arises when one attempts to solve the wave equations (1.14) or (1.15) by the process of separating variables [Silvester (1968) p. 215, Moon-Spencer (1969) p. 249].
This process when applied to (1.14) leads to the space-function vector Helmholtz equation

\[(\nabla^2 + k^2)\hat{S} = 0\]  \hspace{1cm} (1.20)

where \(\hat{S}\) is a time-invariant vector, and \(-k^2\) is a constant independent of time; and it also leads to an equation reflecting the time variations inherent in (1.14)

\[\mu cT''(t) + \omega \sigma T'(t) + k^2 T(t) = 0\]  \hspace{1cm} (1.21)

where \(T(t)\) is a space-invariant time function. These equations are mathematically independent but physically related through the same constant \(k\).

Equation (1.21), a second-order ordinary differential equation, can be solved readily, yielding the constant \(k\). The other equation (1.20), a vector partial differential equation, requires a special treatment, for instance, the application of the separation process once again. But even though this process may be successful, the individual solutions to the resulting equations may not be possible. The reason for the difficulty is that it may not be possible to find a suitable set of boundary values which would correspond exactly to the boundary conditions of the original equation (1.14). In general, the boundary conditions can be applied to the separated equations individually if the original problem boundaries coincide with coordinate surfaces. This unfortunate, imperative demand limits the variety of physical problems to be solved analytically, for the number of suitable coordinate systems is limited.

The field time variations have not been so far restricted. In practice, many problems involve steady-state sinusoidal time-varying
fields. It is convenient to represent each field vector as a complex phasor [Plonsey-Collin (1961) p. 310, Boast (1964) p. 416]. For the sinusoidal time variation, it is equivalent to setting the time function

$$T(t) = e^{j\omega t}$$

where $\omega$ is the angular radian frequency of the phasor.

Substituting $T(t)$ into (1.21) yields

$$k = [\omega^2 \mu \varepsilon - j \omega \mu \sigma]^{1/2}$$  \hspace{1cm} (1.22)

and hence (1.20) becomes

$$(\nabla^2 + \omega^2 \mu \varepsilon - j \omega \mu \sigma) \mathbf{E} = 0$$  \hspace{1cm} (1.23)

The constant $k$ in (1.22) is called the wave number.

The homogeneous vector Helmholtz equation, for harmonic time dependence of $\mathbf{E}$, can be obtained directly from the homogeneous wave equation and reads

$$(\nabla^2 + k^2) \mathbf{E} = 0$$  \hspace{1cm} (1.24)

where $\mathbf{E}$ is now a complex phasor space vector independent of time and $k$ is given by

$$k = \omega / \sqrt{\mu \varepsilon} = \frac{\omega}{\nu} = 2\pi \frac{f}{\lambda} = \frac{2\pi}{\lambda}$$  \hspace{1cm} (1.25)

where $f$ is the frequency, and $\lambda$ is the wavelength. The vector $\mathbf{H}$ satisfies the same Helmholtz equation (1.24).

The inhomogeneous wave equation (1.15) for sinusoidal time variations yields

$$(\nabla^2 + k^2) \mathbf{E} = j \omega \mu \mathbf{J} - \frac{1}{j \omega \varepsilon} \nabla \times \mathbf{H} \cdot \mathbf{J}$$  \hspace{1cm} (1.26a)

where $\mathbf{E}$ and $\mathbf{J}$ are phasor quantities, and the current $\mathbf{J}$ is linked with the charge density $\rho$ through the continuity equation $\nabla \cdot \mathbf{J} = -j \omega p$. The $\mathbf{H}$ vector satisfies the following relation [Plonsey-Collin (1961) p. 321]

$$(\nabla^2 + k^2) \mathbf{H} = -\nabla \times \mathbf{J}$$  \hspace{1cm} (1.26b)

Equations (1.26a) and (1.26b) are referred to as inhomogeneous Helmholtz
1.4 Vector and Scalar Helmholtz Equations

Direct methods of finding the fields from the above equations are feasible in source-free space or for static fields, but they lead to difficulties when solutions are sought for time-varying problems, especially with given sources. Such solutions are greatly facilitated by introducing auxiliary vector and scalar potentials, depending on the type of the given source.

The inhomogeneous vector Helmholtz equation is given [Silvester (1968) p. 234] by

\[ \nabla^2 A - \frac{\partial^2 A}{\partial (ct)^2} = - \omega^2 \vec{A} \]  
(1.27)

for a general case, and for sinusoidal variations

\[ \left( \nabla^2 + k^2 \right) A = - \omega^2 \vec{A} \]  
(1.28)

where \( \vec{A} \) is the magnetic vector potential defined by \( \vec{B} = \nabla \times \vec{A} \).

The inhomogeneous scalar Helmholtz equation is given by

\[ \nabla^2 \phi - \frac{\partial^2 \phi}{\partial (ct)^2} = - \frac{\rho}{\varepsilon} \]  
(1.29)

or for the harmonic variations

\[ \left( \nabla^2 + k^2 \right) \phi = - \frac{\rho}{\varepsilon} \]  
(1.30)

where \( \phi \) is the scalar electric potential. The derivation of both forms of Helmholtz's equation employs the Lorentz condition [Morse-Feshbach (1953) p. 208]

\[ \nabla \cdot \vec{A} = - \mu c \frac{\partial \phi}{\partial t} \]  
(1.31)

which establishes the relation between the two potentials and generalizes
the Coulomb gauge, $\nabla \cdot \mathbf{A} = 0$ [Della Torre-Longo (1969) p. 477].

When the sources $j$ and $\rho$ are identically zero, equations (1.28) and (1.30) reduce to the homogeneous Helmholtz equations

$$\nabla^2 \mathbf{A} = 0,$$  \hspace{1cm} (1.32)\\
and

$$\nabla^2 \phi = 0.$$  \hspace{1cm} (1.33)

1.5 Poisson Equation

If the electric and magnetic fields are static, then $k = 0$ and equations (1.28) and (1.30) reduce to the Poisson equations

$$\nabla^2 \mathbf{A} = -\mu j,$$  \hspace{1cm} (1.34)\\
$$\nabla^2 \phi = -\frac{\rho}{\varepsilon},$$  \hspace{1cm} (1.35)\\
or

$$\nabla^2 \psi = \rho_m$$  \hspace{1cm} (1.36)

where $\psi$ is the magnetic scalar potential, and $\rho_m$ is the magnetic pole density defined by

$$\rho_m = -\text{div} \mathbf{M}$$  \hspace{1cm} (1.37a)

for a volume distribution of poles, plus for the surface distribution

$$\rho_{ms} = \mathbf{I}_s \cdot \mathbf{M}$$  \hspace{1cm} (1.37b)

where $\mathbf{I}_s$ is a unit vector normal to the surface.

1.6 Laplace's Equation

If the sources $j$, $\rho$, and $\rho_m$ are zero, Poisson's equation further reduces to the Laplace equation

$$\nabla^2 \phi = 0.$$  \hspace{1cm} (1.38)

This equation is probably the most important equation of classical mathematical physics [Morse-Feshbach (1953)]. It applies in electro-
and magnetostatics, steady-state thermal conduction, electric conduction, irrotational fluid flow, and gravitation.

1.7 Classification of Second-Order Partial Differential Equations

Most of the above equations are linear second-order partial differential equations. The most general two-dimensional form of this type of equation is given by

\[ A \frac{\partial^2 u}{\partial x^2} + 2B \frac{\partial^2 u}{\partial x \partial y} + C \frac{\partial^2 u}{\partial y^2} + D \frac{\partial u}{\partial x} + E \frac{\partial u}{\partial y} + F u = G \quad (1.39) \]

where \( A, B, \ldots, G \) are assumed here to be functions of the variables \( x \) and \( y \) only, and \( u \) is the sought function. It is convenient to classify second order equations in terms of the coefficients \( A, B, \) and \( C \) in (1.39). It is possible to do so because the discriminant \( (B^2 - AC) \) represents an inherent property of the differential equation (1.39) [Garabedian (1964) p. 57]. The classification is as follows

\[ B^2 - AC > 0, \text{ the hyperbolic type; } \quad (1.40a) \]

\[ B^2 - AC = 0, \text{ the parabolic type; } \quad (1.40b) \]

\[ B^2 - AC < 0, \text{ the elliptic type. } \quad (1.40c) \]

Behaviour of (1.39), methods of solution, including establishing the kind of initial boundary conditions that lead to a unique solution in a material way, depend largely on the type of (1.39). A collection of various equations of different types is given in Appendix I.

1.8 Initial and Boundary Conditions. Uniqueness

A single partial differential equation for one unknown function might possess many solutions, thus the conditions that serve to disting-
uish them from one another in a unique way become essential. Examining the partial differential equations of mathematical physics leads to the concept of the initial conditions or boundary conditions naturally associated with them through the context in which they arise. Furthermore, it is convenient to distinguish between initial value problems and boundary value problems; the former indicates a problem of determining the solution of a partial differential equation with a number of data related to the order of the equation prescribed at an initial point, along an initial curve or line, the latter requires auxiliary data on the boundary of a domain where the equation is defined.

The boundary value problems can be classified according to the following boundary conditions

\[ u(\Gamma) = g(\Gamma), \text{ Dirichlet b.c.} \]  \hspace{1cm} (1.41)

\[ \frac{\partial u}{\partial n} \big|_{\Gamma} = g(\Gamma), \text{ Neumann b.c.} \]  \hspace{1cm} (1.42)

\[ \frac{\partial u}{\partial n} \big|_{\Gamma} + \alpha(\Gamma) u(\Gamma) = g(\Gamma), \text{ third (or Cauchy) b.c.} \]  \hspace{1cm} (1.43)

The first condition states that the solution \( u \) takes on prescribed values \( g \) at the boundary \( \Gamma \), the second condition requires the normal derivatives of the solution \( u \) to take on prescribed values, and the third condition is a linear combination of (1.41) and (1.42).

A wide class of eigenvalue problems ought to be distinguished among the problems associated with the elliptic equations. A general analysis of those problems is given by Courant-Hilbert (1953), Chap. 6.

There exist various proofs of the uniqueness theorem for the three boundary value problems [Moon-Spencer (1969), Garabedian (1964),]
etc.). One of the convenient proofs is based on the maximum principle which states that the solutions of certain partial differential equations of the elliptic type never achieve a strong relative maximum or minimum in the interior of their domain of definition. [Della Torre-Longo (1969) p. 172, Forsythe-Nasow (1960) p. 174, etc.]

Heretofore the problems have been of the initial value, boundary value, and eigenvalue type. The following discussion will concentrate on the boundary value and eigenvalue problems of the elliptic partial differential equations.

1.9 Analytical and Numerical Methods of Solution

Many practical field problems can be solved using direct analytical methods such as the separation of variables used in previous sections. However, even more problems cannot be solved by these methods. For example, Laplace's equation is separable in 11 different coordinate systems in three dimensions, and virtually an infinite number of different systems in two dimensions [Flønsey-Collin (1961) p. 119, Morse-Feshbach (1953) Ch. 5.1] The analytic solutions of the scalar Helmholtz equation are even more restricted than those of Laplace's equation. Thus, when the exact analytical solution is not obtainable, one may resort to a numerical approximation or to a graphical one [Boast (1964) Ch. 23, Moon-Spencer (1961) p. 15] or to an experimental solution [Moon-Spencer (1961) p. 21]. The major methods of the first two types will be reviewed below.

A general introduction to numerical methods in electromagnetic field problems is given by Wexler (1969). A review of some numerical
methods for the waveguide problem is presented by Davies (1972).

1.9.1 Separation of Variables

The method of separation of variables involves a process of breaking the partial differential equation into ordinary differential equations, each of which is a function of one independent variable. These resulting equations are then solved, and any product of their solutions is a particular solution of the original equation. The method, however, is not general. The separability conditions are discussed in general by Moon-Spencer (1961) Chap. 11.

Laplace's and Helmholtz's equations in the usual (Cartesian, cylindrical, and spherical) coordinate systems are solved by this method in most textbooks [e.g., Moon-Spencer (1961), Morse-Feshbach (1953) Ch. 5, Artley (1965)]. Rectangular and circular waveguides were analysed by Schelkunoff (1943) Ch. 8, Jordan-Balmain (1968), the elliptical waveguide by Chu (1938), Rayevskiy-Smorgonskiy (1970), Kretzschmar (1970, 1971), and the parabolic waveguide by Zagrodzinski (1968), Horiuchi et al. (1968), Larsen (1972). Other waveguide shapes have been also attempted by this method [see Ng (1972)].

1.9.2 Conformal Transformation

The conformal transformation (or mapping) preserves both the magnitudes and the sense of angles between curves when mapped from one plane onto another. Many practical problems in electrostatics, heat flow, and fluid can be solved by the conformal mapping. The basic ideas involved are as follows: (i) Find a mapping function that
transforms a given problem into a corresponding one for the unit circle or half plane whose solution is easier obtainable. (ii) Solve the transformed problem. (iii) The inverse mapping function provides the solution in the initial region. A class of problems with polygonal boundaries is amenable to the Schwarz-Christoffel transformation [Silvester (1968) p. 106, Spiegel (1964) p. 204]. Further mathematical and physical discussion of this transformation is given by Walker (1964), Gibbs (1958) and Koppenfels-Stallman (1959). Davies (1972) presents a bibliography concerned with the conformal transformation method as applied to the Helmholtz equation.

1.9.3 Variational Methods

Many field problems described by the elliptic or parabolic partial differential equations may be solved by minimizing (or maximizing) some functional describing their behaviour. The mathematical formulation of this principle leads the minimization of the following integral

[Forsythe-Wasow (1960) p. 163]

$$I(u) = \int_{R} F(x,y,u,u_{x},u_{y}) \, dx \, dy + \int_{\Gamma} G(\xi,u,u_{\xi}) \, d\xi$$  \hspace{1cm} (1.44)

where $R$ is the region of definition of $u$, $\Gamma$ is the boundary of $R$ and it is piecewise analytic [Forsythe-Wasow (1960) p. 146], $\xi$ represents arc length on $\Gamma$, and $u_{\xi} = \partial u / \partial \xi$ along the positive direction of $\Gamma$. Variational calculus applied to (1.44) yields special requirements on the functional form of $u$ that minimizes $I(u)$ [Forsythe-Wasow (1960) Sec. 19.1; Morse-Feshbach (1953) Ch. 3; Courant-Hilbert (1953); Kantorovich-Krylov (1958)]. These requirements are expressed by the
Euler equation. Eq. (1.44) has the following Euler equation
\[ F_{u_{x}u_{x}} u_{x} u_{xx} + 2 F_{u_{x}u_{y}} u_{x} u_{xy} + F_{u_{y}u_{y}} u_{y} u_{yy} + F_{u_{x}u_{y}} u_{x} u_{y} + F_{u_{y}u_{y}} u_{y} u_{y} + F_{u_{x}u_{x}} u_{x} \]
\[ + F_{u_{y}u_{y}} u_{y} + F_{u_{x}u_{x}} + F_{u_{y}u_{y}} - F_u = 0 \text{ in } \mathbb{R} \] (1.45)

The Laplace, Helmholtz, Schrödinger, and other equations of physics are the Euler equations of some integrals of the type of (1.44). Thus, a particular problem may be solved by a direct solution of its Euler equation or by the minimization of the corresponding integral. The latter process of obtaining the solution belongs to the variational methods.

An important consequence of the minimization of \( I(u) \) without any restriction on \( u \) at the boundary is the concept of the natural (or free) boundary condition [Courant-Hilbert (1953) p. 208; Forsythe-Wasow (1960) p. 162; Hazel-Wexler (1972); Chester (1971) p. 344; Mikhlin-Smolitskiy (1967) p. 161]. The natural boundary condition for the general variational problem (1.44) is given by
\[-G + G_{u_{x}} u_{x} u_{x} + G_{u_{y}} u_{y} u_{y} + G_{u_{x}} u_{x} + G_{u_{y}} u_{y} - F_{u_{x}} \frac{dy}{dz} - F_{u_{y}} \frac{dx}{dz} = 0 \text{ on } \Gamma \] (1.46)

This boundary condition is called natural because it is satisfied by the function \( u^* \) minimizing \( I(u) \). The variational method eliminates, therefore, certain boundary conditions such as the homogeneous Neumann boundary conditions for the Helmholtz equation, from special treatment by showing them to be natural.

Two often-used variational approximate methods of finding \( u^* \) are the Rayleigh-Ritz method [Kantorovich-Krylov (1958) p. 258; Garabedian (1964) Sec. 8.1 and 11.1; Berezin-Zhidkov (1965) p. 608; Collatz (1960),


1.9.4 Perturbational Methods

Perturbational methods [Harrington (1961) Ch. 7] can yield some quantitative changes in the desired quantities, such as resonant frequencies, impedances, etc., when the geometry of the problem or the properties of the involved mediums are perturbed. It is in contrast to the variational methods that lead to the desired quantities themselves.

Convenient boundary perturbation techniques are given by Schelkunoff (1943) Sec. 10.9 and Davies-Kretzschmar (1972). Several references concerned with the geometric approximation technique related to the perturbational methods are given by Ng (1972).
1.9.5 Finite Difference Methods

The finite difference methods have been successfully used in boundary value problems for many years. The general theory of these methods is given in numerous books [e.g., Forsythe-Wasow (1960); Collatz (1960); Greenspan (1965); Smith (1965); Ames (1969), Rosenberg (1969), Berezin-Zhidkov (1965)]. The basic ideas of the finite difference approach to field problems are also discussed by Wexler (1969), Green (1965, 1967), Sinnott et al. (1969), Gupta (1969) Davies (1972). These papers cite many references concerned with the solution of Laplace's and Poisson's equations by the finite difference methods. Davies-Muilwyk (1966), Steele (1968); Beaubien-Wexler (1968, 1970, 1971a, 1971b), and Pontoppidan (1969) dealt with the Helmholtz equation. Many finite difference techniques have been given by Samarskii [e.g., Samarskii-Fryazinov (1971)].

The finite difference schemes have been used for solutions of nonlinear magnetic field problems [Ahamed (1965); Fuchs-Erdelyi (1973); Hong-Stefanokos (1973); Simons et al. (1970); Kozakoff-Simons (1970)].

Some unbounded (or exterior) field problems have been solved by the boundary relaxation technique with the help of the finite difference methods [Cermak-Silvester (1968, 1970); Silvester-Cermak (1969); Sandy-Sage (1971); Richter (1970); Greenspan-Werner (1966)].

1.9.6 Finite Element Methods

These methods belong essentially to the variational techniques because they attempt to minimize the (energy) functionals associated with specific problems. The apparent differences between the finite
difference and the finite element methods [Silvester (1972a); Davies (1972)] caused that the latter methods attracted great attention from various areas of physics and engineering. The methods are applied in structural mechanics, electrostatics, magnetostatics, electromagnetics, microwave acoustics, etc.


1.9.7 Integral Equation Methods

Many boundary-value problems arising in electromagnetics can be expressed in a dual form: by the differential or integral equations (see Sec. 1.1). The important advantages of the integral operator formulations are: (i) the inclusion of all boundary conditions in the process of obtaining the mathematical model, (ii) the flexibility in handling irregular and unbounded regions; and (iii) the possibility of reducing by one the number of space variables.

The vector and scalar potentials describing a field problem are
expressible as integrals in terms of the field sources. The sources may be approximated by various models; the integral equation takes therefore different forms.

Analytical solutions of integral equations are usually very difficult to achieve [Collatz (1960) Ch. 4; Courant-Hilbert (1953) Ch. 3; Morse-Feshbach (1953) Ch. 8]. Numerical solutions, however, of these equations are available [Mikhlin-Smolitskiy (1967) Ch. 4; Berezin-Zhidhov (1965) Sec. 10.10; Kantorovich-Krylov (1958) Ch. 2]; in particular, the projective method [Silvester-Hsieh (1971b)], the matrix methods [Harrington (1967)], and the method of moments [Kantorovich-Krylov (1958) pp. 150-4; Harrington (1968); Vorobyev (1965)].

Microstrip problems were solved by a projective method [Benedek-Silvester (1972); Silvester-Benedek (1972, 1973)] and by method of moments [Farrar-Adams (1971, 1972)]. Waveguide problems were attempted by the method of moments [Spielman-Harrington (1972), Wu-Chow (1972), Chow-Wu (1973). Magnetic field problems were attached by direct or iterative matrix methods [Zaky-Robertson (1973); Karmaker-Robertson (1973); Halacy (1971)]. Unbounded field problems were treated by a combined integral-finite element approach [McDonald-Wexler (1972a, 1972b)].

The two following subsections will list techniques related to the integral equation methods.

1.9.8 Null Field Method

This method uses integral equations with nonsingular integrands.
The null field equations have been derived by Bates (1969) using an extended boundary condition [Waterman (1965)]. The method was applied to two-dimensional scatterers [Bates (1968, 1970); Hunter-Bates (1970, Hunter-Bates (1972)], and for various waveguide shapes with sharp re-entrant corners [Ng-Bates (1972)]. The accuracy of the method as compared to Silvester's high order finite element method [Silvester (1969c)] was discussed Ng-Bates (1972).

1.9.9 Point Matching Methods

The formulations of these methods employ functions that exactly satisfy appropriate elliptic partial differential equations, in particular, the Helmholtz equation. The resulting integral equations can be solved by various numerical methods.

These methods are classified as the straightforward point matching method [Yee-Audeh (1965, 1966a, 1966b); Bates (1967, 1969); Fuller-Audeh (1969)], the complete point matching and the alternative point matching methods [Bates (1969)], and the extended point matching method [Bates-Ng (1973)].

These methods were used in solving the propagation problems in waveguides, the scattering from metallic boundaries, the waveguide junction circulators [El-Shandwily (1973)] and other problems [Adams-Muertz (1969)]. A comparison and an evaluation of the methods is presented by Davies (1972) and Ng (1972).

1.9.10 Other Methods

The methods listed in this subsection can be applied to the
Helmholtz and (or) Laplace equations. Some of them have not been much exploited yet; others have limited application due to their disadvantages.

Harrington's method reduces Helmholtz's equation to a system of coupled first order differential equations [Harrington (1968) pp. 162-166]. This approach appears to be very promising [Davies (1972)].

Network-analogue methods [Kron (1944), Lennartsson (1972)] might be very powerful if one would find a fast method of solving large electrical networks.

Hybrid methods reviewed by Wexler (1969) require further investigation.

The Monte Carlo method [Hammersley-Handscomb (1964) Silvester (1968) Sec. 2.8] was often used [Royer (1971), Nasow (1951)] although it is useful for field computations at a few points only.

The method of partial regions [Collins-Daly (1964)] and the transverse resonance method [Pyle (1966)] have both limited applicability. Ng (1972) cites other authors who used these methods for waveguide problems.

1.10 Large Systems of Linear Equations

Many of the methods listed in the previous section utilize certain approximations to the original equation describing a problem. As a consequence of these approximations, large sets of linear, simultaneous equations are generated. A solution of the equations yields an approximate answer to the original problem.

A set of linear, simultaneous equations may be conveniently represented in the matrix notation
(1.47)

where \( A \) is an \( n \times n \) matrix, \( \chi \) and \( \zeta \) are \( n \)-element vectors (column matrices) from which \( \zeta \) is known. This system can be solved by various direct and/or iterative techniques [Faddeev-Faddeeva (1963); Bodewig (1959); Householder (1964); Walsh (1967); Lanczos (1956); Traub (1964); Varga (1962); Young (1971); Wachspress (1966); Forsythe-Moler (1967); Acton (1970); Westlake (1968)]. Some of these methods will be analysed in Chapter 2.

1.11 Matrix Eigenvalue Problem

An approximation of Helmholtz's equation leads to the general matrix eigenvalue problem

\[
A \chi = \lambda B \chi
\]  

(1.48)

where \( A \) and \( B \) are \( n \times n \) matrices, \( \lambda \) represents the eigenvalue(s) (characteristic, proper values), \( \lambda_i \), of the mathematical model described by (1.48), and \( \chi \) is the \( n \)-element eigenvector(s) associated with the particular \( \lambda_i \) (\( i = 1, \ldots, n \)). The problem is to find eigenvalues \( \lambda \) and associated eigenvectors \( \chi \) such that (1.48) holds. A variety of direct and iterative methods for solving (1.48) have been presented [e.g., Berezin-Zhidkov (1965) Ch. 8; Krzyzanski (1971) Ch. 14]. Some of these methods yield only partial eigensolutions, i.e., the dominant and several closest eigenvectors, others lead to the complete eigensolution. Certain aspects associated with this problem shall be discussed in Chapter 2.
1.12 Errors in Numerical Methods

Modelling (or approximating) a physical problem, described by partial differential equation, and the use of computers with finite word length are the main sources of errors. Assume a domain Ω with its boundary Γ and sufficiently differentiable functions u on Ω = R U Γ that may be required to satisfy certain boundary conditions on Γ. A specific problem may be represented by an operator equation, say \( L[u] = f \), where \( L \) is a self-adjoint\(^1\) linear differential operator. The exact solution function of that problem will be denoted by \( u^* \). Further assume that \( \Gamma \) approximates \( Γ \), and \( \Gamma \) is the boundary set of \( R^2 \); each dependent function \( u \) is replaced by a function \( U \) defined on \( R \). The problem involving \( L[u] \) and determining \( u^* \) is thus replaced by an associated problem having \( L_h[U] \) and determining \( U^* \). This process is called discretization. It leads to the following two errors:

The truncation error reflects the discrepancy between \( L \) and \( L_h \) on \( R \). For example, the truncation error of the finite difference equation approximating Laplace's equation in three dimensions may be

\[ \int_\Gamma [wL(v) - vL(w)] dxdy \]

is a function of the values of \( v, w \) and their derivatives on \( Γ \) above. \[ (\text{Forsythe-Wasow (1960) p. 167}). \] Self-adjointness is analogous to the symmetry of a matrix operator.

1 Self-adjointness of \( L \) means that, for any two sufficiently smooth functions \( v, w \),

2 \( R \) is a finite set, as in finite difference/element methods.

The members of \( R \) may be points, arcs, areas, or other entities.
\[ \sum_{i=1}^{n} \theta_{i}^2 \] where \( \theta_{i} \) is the interval size in each dimension (see footnote 1 for the 0-notation). This error indicates the error of the method and not of the solution.

The discretization error indicates the error in the solution due to an approximation of \( L[u] \) by \( L_{h}[U] \) on \( \Omega = \overline{\Omega} \cup \Gamma \). It includes a boundary truncation error resulting from approximating the non-Dirichlet boundary conditions on \( \Gamma \), thus giving a measure of the error between the true solution \( U^* \) on \( \Omega \) and the approximate solution \( U^* \) on \( \Omega \).

The round-off error results from the finite computer word length. It means that an approximate problem involving \( L_{h}[U] \) cannot be solved exactly and \( U^* \) is not practically obtainable. To ascertain the contribution of the round-off error to the best possible estimate to \( U^* \), probability methods must be employed because of the random nature of this error. [Forsythe-Wasow (1960) Sec. 23, Ames (1969) p. 25]. This error affects both direct and iterative methods [Forsythe-Moler (1967); Wilkinson (1971)].

The residual error is an extension of the round-off error concept. It indicates the error between \( U^* \) and the actual estimate to \( U^* \). It is particularly useful in the iterative processes in which the required estimate to \( U^* \) may be obtained when some error criterion is satisfied; thus, the residual error may not be equal to the round-off error.

---

1 An approximation \( U(x) \) to \( u(x) \) is of order \( n \) with respect to some quantity \( \Delta x \) if \( n \) is the largest possible positive real number such that \( |U - u| = O((\Delta x)^n) \) as \( \Delta x \to 0 \). It is Landau's asymptotic 0 (big 'oh') notation.
1.13 Convergence, Stability, Consistency

The concept of convergence is related to the discretization and residual errors. The first type of convergence describes how $U^* + u^*$ when certain parameters, usually the interval sizes, tend to zero; the rate of convergence is usually measured in Landau's O-notation. It is frequently encountered in the literature [Forsythe-Wasow (1960)] that this concept is somewhat simplified by assuming equivalence between $\mathbb{R} U \bar{\Gamma}$ and $R U \Gamma$, thus showing the convergence of $U^*$ on $R U \Gamma$ to $u^*$ on the same region when the interval sizes approach zero.

The convergence associated with the residual error shows how successive estimates $U^{(n)}$ approach $U^*$ when the number of iterations, $n$, increases; the rate of convergence is usually measured in terms of the eigenvalues of an iteration matrix. There are various methods of estimating the eigenvalues of an iteration matrix, leading to different values of the measure of convergence [Young (1971) p. 84].

The round-off error and any other computational error may lead to numerical instability of a problem even though the problem is physically stable. Various stability criteria are required for different equations [Forsythe-Wasow (1960) Sec. 5, 12, 23, and 29].

The concept of consistency (compatibility) is related to the discretization error and residual error. The first type of error may cause that successive refinement of the discretizing intervals generates a solution which is stable, but converges to the solution of an equation of different type [Ames (1969) p. 62]. In this case, an approximating equation is not consistent with the original equation.
The consistency of an iterative method (residual error) requires that, ideally, for any starting \( U^{(0)} \) it converges to \( U^* \) [Young (1971) Sec. 3.2].

1.14 Summary

Many problems of engineering and physics can be classified within the following three physical categories: propagation problems, eigenvalue problems, and equilibrium problems.

Propagation problems have an unsteady state or transient nature. A solution to such a problem describes the behaviour of a system subject to some initial state. Mathematically, these problems are known as initial value problems, and are described by parabolic or hyperbolic equations of the type

\[ L[x] = f \] (1.49a)

with the initial state

\[ E_i[x] = e_i \] (1.49b)

and prescribed conditions

\[ B_i[x] = b_i \] (1.49c)

on the open boundaries. \( L, E \) and \( B \) are linear operators acting upon a function \( x \) which is sought within an open region \( R \).

Eigenvalue problems describe steady-state systems with one or more characteristic states. The solution(s) to such problems must yield not only the steady-state configuration \( x \) but also the associated eigenvalue(s) \( \lambda \), from the following operator equation
\[ L[x] = \lambda M[x] \]  \hspace{1cm} (1.50a)

within a domain \( R \), and

\[ B_i[x] = \lambda C_i[x] \]  \hspace{1cm} (1.50b)

on the boundary \( \Gamma_i \) of the region \( R \) which is often closed and bounded.

The operators \( L \) and \( M \) are of elliptic type.

Equilibrium problems also include steady-state systems but with only one equilibrium state. Thus the elliptic equation (1.50) simplifies to the following form

\[ L[x] = f \]  \hspace{1cm} (1.51a)

and

\[ B_i[x] = b_i \]  \hspace{1cm} (1.51b)

These problems are known as boundary-value problems.

Solutions to the above problems are obtained by analytical or numerical methods; the important methods have been reviewed in this chapter. The finite difference methods and the finite element methods, particularly amenable to high-speed computing machines, will be discussed in the following chapters in order to expedite subsequent convergence acceleration analyses of these and related methods.
1.15 Remarks

The basic concepts concerning fields follow those of Artley (1965), Silvester (1968), Della Torre-Longo (1969), Fano-Chu-Adler (1960), Harrington (1961), etc.

The wave equations, when material polarization $\mathbf{P}$ and $\mathbf{M}$ are explicitly taken into account, are given in Collin (1966) p. 61.

The method of separation of variables is used in most of the books concerned with electromagnetics. All the basic equations are solved by this method in Moon-Spencer (1961).

It should be noted that there is a difference between the scalar Laplacian (the Laplacian acting on a scalar) and the vector Laplacian (the Laplacian acting on a vector). Moon-Spencer (1969) p. 285 suggest different symbols for them, i.e., the ordinary scalar Laplacian is defined as $\nabla^2 = \text{div} \ \text{grad}$ and the vector Laplacian is $\bigtriangledown^2 = \text{div} \ \text{grad} - \text{curl} \ \text{curl}$. The first symbol above is used in this thesis following a common notation in most of the texts. It is also justified by the fact that the vector Laplacian in rectangular coordinates is the scalar Laplacian for each component of the vector potential.

If $J = \rho = 0$ and the dipole polarization $\mathbf{P}$ or the magnetic dipole $\mathbf{M}$ are considered, the vector Helmholtz equation (1.28) requires minor modifications only [Collin (1966) pp. 62-63].

It is interesting to note that the most attractive, finite element methods have been suggested in Courant's (1943) paper which showed that the variational approach, using piecewise linear trial functions, could lead to the standard 5-point Laplacian. This idea, however,
remained dormant for many years. Sinnott (1969) and Sinnott et al. (1969) used Synge's pyramid functions [Synge (1957)] as the trial functions and obtained results related to the simple finite element method. Recent surveys on the finite element method have been compiled by Zienkiewicz (1970) and Singhal (1969).

CHAPTER 2

FINITE DIFFERENCE FORMULATION

2.1 Introduction

The finite difference methods (Sec. 1.9.5) (or the mesh, or grid methods) are widely used in solving boundary value problems governed by differential equations of the elliptic type. The basic idea of discretization, developed in Sec. 1.12, is applied in the finite difference approach to partial differential equations, approximating them by finite difference equations. This process of substituting difference equations for differential equations can be accomplished in various ways and different formulae can result from those derivations. The discretization error does depend on the chosen formula.

The discretization gives rise to a system of linear simultaneous equations which can be solved by direct or iterative methods. The coefficient matrix of the system possesses certain properties which should be considered in choosing a proper solution method of the system.

These problems will be discussed in this chapter.

2.2 Discretization

To illustrate the discretization, the Dirichlet problem for Laplace's equation will be approximated by a finite difference equation. The problem is

\[
\frac{\partial^2 u(x)}{\partial x_1^2} + \frac{\partial^2 u(x)}{\partial x^2} = 0 \quad \text{for } x \in \mathbb{R} \quad (2.1)
\]
\[ u = g(x) \quad \text{for } x \in \Gamma \]  

(2.2)

where it is presupposed that \( u(x) \) is continuous and sufficiently differentiable within the region of its definition \( \Omega \) and takes definite values of \( g \) on the boundary \( \Gamma \), and \( x = (x_1, x_2) \) is a point of two-dimensional space \( V^2 \). A solution \( u(x) \in C^4 (\Omega = \mathbb{R} \cup \Gamma) \) of (2.1) and (2.2) is assumed to exist.

2.2.1 The Lattice

Consider two families of parallel straight lines (Fig. 2.1)

\[ x_1 = x_{10} + ih_1, \quad i = 0, \pm 1, \ldots \]  

(2.3a)

\[ x_2 = x_{20} + jh_2, \quad j = 0, \pm 1, \ldots \]  

(2.3b)

where \( h_1 \) is a positive variable. The points of intersection of these lines are denoted by \( x \) and they form a lattice \( V_h^2 \) in the initial space \( V^2 \).

2.2.2 The Mesh

The nodes of the lattice \( V_h^2 \), belonging to \( \mathbb{R} \) are called interior nodes and are denoted by

\[ \bar{\Omega} = \{ x \in \Omega \cap V_h^2 \}. \]  

(2.4)

The intersection of all the straight lines with the boundary \( \Gamma \) forms the boundary points (not necessarily nodes) denoted by \( \Gamma^{-1} \). It is convenient to distinguish subsets of boundary points with respect to the direction of the coordinate axes \( x_\alpha (\alpha = 1, 2) \), i.e., \( \Gamma_\alpha = \Gamma^{+}_\alpha \cup \Gamma^{-}_\alpha \), where \( \Gamma^{+}_\alpha \) and \( \Gamma^{-}_\alpha \) are the sets of right- and left-hand boundary points with

\[ \text{Note that } \Gamma \text{ may not belong to the lattice } V_h^2. \]
Fig. 2.1. (a) Continuous domain $\Omega = \mathbb{R} \cup \Gamma$; (b) Discrete representation $\hat{\Omega} = \hat{\mathbb{R}} \cup \hat{\Gamma}$ of $\Omega$: $\times$ - for regular interior nodes $\hat{\mathbb{R}}_0$, $\Theta$ - for near-boundary nodes $\hat{\mathbb{R}}_1$, $\bigcirc$ - for boundary points $\hat{\Gamma}_1$, $\bigcirc$ - for $\hat{\Gamma}_1^+$, $\Box$ - for $\hat{\Gamma}_2^-$, $\blacksquare$ - for $\hat{\Gamma}_2^+$. 
respect to \( x_a \). Thus \( \Gamma = \bigcup_{a=1}^{2} \Gamma_a \). All the interior and boundary base-points form the mesh \( \bar{\Omega} = \bar{\Omega} \cup \Gamma \).

2.2.3 Regular and Near-Boundary Interior Nodes

Two base-points are adjacent (or neighbours) along \( x_a \) if the distance between them is not greater than \( h_a \), where \( h_a \) denotes the mesh step (or size, or interval) along \( x_a \). An interior node which has at least one neighbour belonging to \( \Gamma \) is called a near-boundary node. A set of these points is denoted by \( \bar{\Omega}_I \). The complement up to \( \bar{\Omega} \) is denoted \( \bar{\Omega}_O \), so that \( \bar{\Omega} = \bar{\Omega}_I \cup \bar{\Omega}_O \). \( \bar{\Omega}_O \) contains the regular interior nodes, all the neighbours of which are interior nodes only (see Fig. 2.1).

2.2.4 Mesh Cells

In order to introduce the concept of a closed region \( V_\varepsilon(x) \) (the cell area), a mean mesh step \( \bar{h}_a(x_k) \), at an interval node \( x_k \) in \( x_a \) direction, is needed. This mean mesh step is defined as

\[
\bar{h}_a = 0.5(h_a^+ + h_a^-)
\]

where \( h_a^+ = x_{a,k+1} - x_{a,k} \) and \( h_a^- = x_{a,k-1} - x_{a,k-1} \). It is assumed that all the nodes involved here belong to \( V_{h}^2 \). The distance between a near-boundary node and the adjacent boundary base-point, which is not a point of the lattice \( V_{h}^2 \), will be denoted as \( h_{a*}^+ \) or \( h_{a*}^- \). Consequently,

\[
\bar{h}_a = 0.5(h_{a*}^+ + h_{a*}^-) \quad \text{or} \quad \bar{h}_a = 0.5(h_{a*}^+ + h_{a*}^-).
\]

Henceforth, a mesh cell associated with an internal node can be
expressed by

\[ V_c(x) \in \mathbb{R} = \prod_{\alpha=1}^{2} h_\alpha. \]  

Cells for a regular interior mesh point and for a near-boundary point are shown in Fig. 2.2(a) and (b), respectively.

Fig. 2.2. Mesh cells: (a) for a regular interior node, and (b) for a near-boundary node.

The concept of the mean mesh step and the mesh cell is often used in determining the truncation or discretization errors (see Sec. 4.5).

2.2.5 Difference Operators

To solve the problem (2.1) and (2.2) on mesh \( \mathbb{R} \), two types of operators are required: (i) for the regular interior nodes in \( \mathbb{R}_o \), and (ii) for the near-boundary nodes in \( \mathbb{R}_f \). The second type will be considered in the next section.
Assume that \( h_1 = h_2 = h \). For any \( x \in \mathbb{R} \) the derivatives in 
\( (2.1) \) can be substituted by a finite difference (forward, backward, or
central [Spiegel (1971) Ch. 1]) approximation. An equivalent approach
is the application of the two-dimensional form of Taylor's series to
the function \( u \) at any point. The Taylor series is formally given by
\[
    u(x) = e^{x_1 \frac{3}{\partial x_1} + x_2 \frac{3}{\partial x_2}} u(0).
\]
\[ (2.8) \]
According to (2.3), using the double subscript notation for \( u \) at each
point yields
\[
    \sum u = \left[ e^{x_1 \frac{3}{\partial x_1}} + e^{x_1 \frac{3}{\partial x_1}} + e^{x_2 \frac{3}{\partial x_2}} + e^{x_2 \frac{3}{\partial x_2}} \right] u_{i,j}
\]
\[
    = 2[1 + \frac{h^2}{2} \frac{\partial^2}{\partial x^2} + \frac{h^4}{4!} \frac{\partial^4}{\partial x^4} + \ldots
\]
\[
    + 1 + \frac{h^2}{2} \frac{\partial^2}{\partial x^2} + \frac{h^4}{4!} \frac{\partial^4}{\partial x^4} + \ldots] u_{i,j}
\]
\[
    = 4 u_{i,j} + h^2 v^2 u_{i,j} + O[h^4] \quad (2.9)
\]
where \( \sum u \) represents the sum of the function values at the adjacent
nodes. Eq. (2.9), upon division by \( h^2 \), results in the relation
\[
    v^2 u_{i,j} = h^{-2}[u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{i,j}] + O[h^2]
\]
\[ (2.10) \]
The quantity \( O[h^2] \) represents Landau's asymptotic notation of the
truncation error of this approximation. Thus the 5-point operator can
be written as
\[
    h^{-2}[u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{i,j}] = 0, \quad (2.11)
\]
or symbolically as shown in Fig. 2.3.

\[ h^2 \]

\[ 1 \]

\[ -4 \]

\[ 1 \]

\[ 1 \]

Fig. 2.3. Five-point operator.

Various networks [Collatz (1966) p. 389] can be utilized in the derivation of the finite difference operators (see Sec. 2.2.7). Appendix 2 lists such operators for Poisson's equation on uniform networks. A variety of finite difference operators for the differential operators \( \nabla^2 \) and \( \nabla^4 \) is given in Appendix 3. Silvester (1970c) introduced a numerical automatic method for the finite difference operators formulation, as an alternative to the Taylor's series method.

The finite difference solution of eigenvalue problems will be discussed at the end of this chapter, after the introduction of the necessary definitions.

2.2.6 Treatment of Boundary Conditions

An examination of Fig. 2.1b and the definition of the boundary and near-boundary base-points reveal several problems associated with
the approximation of boundaries. For arbitrarily shaped boundaries, some boundary points may not belong to the lattice \( V_h^n \). Therefore, the equal-arm operators (on a uniform network) cannot be used. There are two possible treatments of this problem: (i) the boundary is changed to coincide with the lattice nodes, or (ii) unequal-arm operators are derived to include the boundary points not belonging to the lattice.

In the first approach, one has three options of approximating the boundary by deforming them to: (a) collocate with the near-boundary nodes \( \bar{R} \), (b) collocate with the "near-boundary" nodes outside the mesh \( \bar{\Omega} \), or (c) collocate with some closest (internal) near-boundary nodes and some closest (external) "near-boundary" nodes (see Fig. 2.4). The author's experience (see also Davies-Muilwyk (1966); Beauhien-Wexler (1968)) indicates that the third approximation is the best, for the random perturbations from the actual boundary tend to cancel out their effects on the field computation. In the waveguide problems the cutoff frequency is smaller than for the true boundary when method (a) is used, and higher for method (b).
Fig. 2.4. Polygonal interpolation of Dirichlet boundary conditions: (a) internal, (b) external, and (c) averaging.

The interpolation of degree zero, shown in Fig. 2.4, is particularly suitable for the 5-point operator. The local truncation error of such interpolation is \(0[h]\). Higher-degree (one or two) interpolation have been also used [Forsythe-Wasow (1960) p. 200]. They have smaller truncation errors. They also relate to the unequal-arm operators [Fox (1962) p. 262; Ames (1969) p. 30]. Referring to Fig. 2.4(a), the 5-point unequal-arm operators must be used at all the near-boundary nodes if the polygonal interpolation is not used.

The procedures for treating the Neumann boundary conditions is more complicated. In many applications, these boundaries result from symmetry consideration and, therefore, can be taken parallel to the network (Fig. 2.5).
Fig. 2.5. Normal derivatives on rectilinear boundary.

The normal derivatives can be readily approximated by finite differences and the fictitious points eliminated from the finite difference operator which has to be applied at each boundary point. Wexler (1969) analyses the 5-point operator. Curved boundaries with conditions of Neumann type are discussed by Forsythe-Wasow (1960) p. 202, Ames (1969) p. 32, and Beaubien-Wexler (1970) p. 1135.

2.2.7 Choice of Networks

The discretization of $R$ can be performed by means of various sets, i.e., rectangular, triangular, hexagonal, or irregular. From the digital computing point of view, the regular nets are preferable because they simplify programming details of the resulting difference equations, decreasing the execution time of a program. Other factors, however, such as the need for exact treatment of curved boundaries or very non-uniform variation of the gradient of $u$ within $R$, may determine the choice of unequal-arm operators, graded regular meshes, or even
nonuniform meshes.

The unequal-arm operators [Fox (1962) p. 262] have been successfully used by Beaubien-Wexler (1970, 1971b) in the solution of higher waveguide modes. Execution time of the computer program [Beaubien-Wexler (1971b)] is several minutes per mode on a CDC 6400, after the complete conversion of the program from double to single precision.

The graded mesh [Allen (1954) p. 69] may be useful in many practical field computations. For example, computing a magnetic field distribution in a thin film Permalloy and in an outside region, much larger than the thickness of the film, is facilitated by the use of a graded mesh. Such meshes require several formulae associated with the mesh size. For instance, a solution of Poisson's equation on a mesh shown in Fig. 2.6 utilizes the following 5-point operators

\[
\sum_{i=1}^{4} u_i - 4u_0 = h^2 f_0, \tag{2.12}
\]

for the nodes marked □ on the coarse mesh,

\[
\sum_{i=1}^{4} u_i - 4u_0 = \frac{h}{\sqrt{2}} f_0, \tag{2.13}
\]

at the nodes ○ on the transition mesh, and

\[
\sum_{i=1}^{4} u_i - 4u_0 = \frac{h}{2} f_0 \tag{2.14}
\]
for the fine mesh with nodes $\bullet$.

![Graded mesh diagram]

Fig. 2.6. Graded mesh.

The nonuniform n-dimensional lattices with unequal-arm operators at curved boundaries have been considered by Samarskii-Fryazinov (1971). The derived finite difference schemes for solving the Dirichlet problem for an elliptic partial differential equation within an arbitrary region are shown to have the discretization error $O[h^2 \ln(V_o/V_e^*)]$ where $V_o$ is the volume of the region $R$ in the n-dimensional space $V^n$, $h$ is the maximum step in the spatial lattice $V^n_h$

$$h = \max_{x^\alpha \in \Omega} \max_{1 \leq \alpha \leq n} \overline{h}_\alpha(x^\alpha), \quad (2.15)$$

$\overline{h}_\alpha(x^\alpha)$ is the mean step at the node $x^\alpha$ of the mesh $\overline{R}$ in the direction of the $x^\alpha$ axis ($\alpha = 1, 2, \ldots, n$), and $V_e^*$ is the minimum volume of a cell,

$$V_e^* = \min_{x^\alpha \in \Omega} V_e(x^\alpha), \quad V_e(x^\alpha) = \prod_{\alpha=1}^{n} \overline{h}_\alpha(x^\alpha) \quad (2.16)$$
A similar notation shall be used for showing the truncation error for the simple finite element method (see Sec. 4.5).

2.2.8 **Singularities**

A *singular point* of a function $f(x)$ is a value of $x$ at which $f(x)$ fails to be analytic (single-valued and continuous with its derivatives). A partial differential equation has a singularity if one, or more than one, of its coefficients becomes singular at any point (or points). Thus, a finite difference equation will ordinarily also have a singularity at this point. Physically, *re-entrant corners* (Fig. 2.7) belong to singular points.

![Fig. 2.7. Re-entrant corner with $\alpha > \pi$.](image)

A finite difference scheme must be modified to adopt singularities. Such techniques as mesh refinement in the vicinity of a singular point, the method of Motz and Woods, subtracting off the singularity, and removal of singularity are discussed by Ames (1969) Sec. 5.1.

2.2.9 **Inhomogeneous Media**

The equations derived so far are valid for homogeneous media filling the region of interest. Many problems do involve media whose physical, chemical, and/or thermo-dynamic properties are functions of
position, therefore, the equations have to be modified. Some of these problems involve piecewise inhomogeneous media with distinct boundaries, called \textit{interfaces} \cite{Della Torre-Longo (1969) p. 148; Silvester (1968) p. 179}.

For inhomogeneous magnetic media, the Poisson equation is given by \cite{Kinsner-Della Torre (1973a)}

\begin{equation}
\nabla^2 \mathbf{A} + \mu^{-1} (\nabla \mathbf{u}) \times (\nabla \times \mathbf{A}) = -J
\end{equation}

where $\mathbf{A}$ is the magnetic vector potential ($B = \text{curl } \mathbf{A}$), $J$ is the current density, and $\mu$ is the permeability, or by

\begin{equation}
\nabla^2 \psi + \mu^{-1} (\nabla \mathbf{u}) \cdot (\nabla \psi) = \rho_m
\end{equation}

where $\psi$ is the magnetic scalar potential ($H = -\text{grad } \psi$), and $\rho_m = -\text{div } M$ is the magnetic charge density. Eq. (2.18) is identical to the electrostatic potential, $\phi$, obeying Poisson's equation in inhomogeneous media \cite{Della Torre-Longo (1965)}

\begin{equation}
\nabla^2 \phi + \epsilon^{-1} (\nabla \epsilon) \cdot (\nabla \phi) = -\epsilon^{-1} \rho
\end{equation}

where $\epsilon$ is the permittivity, and $\rho$ is the charge density.

It is seen that Poisson's equation is modified by the terms $\mu^{-1} (\text{grad } \mathbf{u}) \cdot (\text{curl } \mathbf{A})$ in regions with currents or by $\mu^{-1} (\text{grad } \mathbf{u}) \cdot (\text{grad } \psi)$ in current-free regions. The same term appears in Laplace's equation for source-free regions.

Thus, a finite difference analogue to Poisson's or Laplace's equations also involves a finite difference representation of this term. Let $S$ be a finite difference operator which approximates the gradient operator in the limit, and $T$ the corresponding operator for the Laplacian.
Fig. 2.8. Regular interior node $0$ with six adjacent nodes.

The adjacent nodes to a regular interior node are defined as shown in Fig. 2.8 for a uniform lattice with spacing $h$. Then the homogeneous equation for (2.18) is approximated by

$$ T(\psi) + \mu^{-1} \xi(\mu) \cdot \xi(\psi) = 0 $$

(2.20)

where

$$ \mu^{-1} \xi(\mu) \cdot \xi(\psi) = \frac{1}{\mu(0)h^2} \left[ \left( \frac{\mu(1) - \mu(2)}{2} \right) \lambda_x + \left( \frac{\mu(3) - \mu(4)}{2} \right) \lambda_y + \left( \frac{\mu(5) - \mu(6)}{2} \right) \lambda_z \right]. $$

$$ = \frac{1}{h^2} \left\{ \alpha \left[ \psi(1) - \psi(2) \right] + \beta \left[ \psi(3) - \psi(4) \right] + \gamma \left[ \psi(5) - \psi(6) \right] \right\} $$

(2.21)

with

$$ \alpha = \frac{\mu(1) - \mu(2)}{4\mu(0)} , $$

(2.22a)

$$ \beta = \frac{\mu(3) - \mu(4)}{4\mu(0)} . $$

(2.22b)
\[ \gamma = \frac{\mu(5)-\mu(6)}{4\mu(0)} \]

and

\[ T(\psi) = \frac{1}{2} \left[ \sum_{i=1}^{6} \psi(i) - 6\psi(0) \right]. \tag{2.23} \]

Combining these yields

\[(1+\alpha)\psi(1)+(1-\alpha)\psi(2)+(1+\beta)\psi(3)+(1-\beta)\psi(4)+(1+\gamma)\psi(5)+(1-\gamma)\psi(6)-6\psi(0) = 0 \tag{2.24} \]

This operator is illustrated in Fig. 2.9 and is valid in regions where the permeability is continuous or piecewise continuous, i.e., \( \psi \) exists and is finite. The operator may be used in iterative solutions of problems when \( \mu \) is not only a function of position but also a function of the magnitude of the magnetic field intensity, \( \mu = \mu(H) \).

A case of special interest is the discrete inhomogeneity. For example, when the point 0 lies on an interface between two uniform media whose boundary is the y-z plane, as shown in Fig. 2.10, then \( \beta \) and \( \gamma \) are zero. However, it is incorrect to simply replace \( \mu(1) \) by \( \mu(0^+) \) and \( \mu(2) \) by \( \mu(0^-) \) in (2.21) in order to approximate the gradient of \( \mu \). The gradient of \( \mu \) is singular at the boundary and the correct operator can be obtained by equating the normal components of \( \vec{B} \). That is

\[ \mu_1[\psi(1) - \psi(0)] = \mu_2[\psi(0) - \psi(2)] \tag{2.25} \]

which leads to the operator illustrated in Fig. 2.10. A similar approach to the discrete inhomogeneities has been demonstrated by Dorny (1969) and Monson (1972).

Analogous operators can be obtained for other interface geometries.
Fig. 2.9. General three-dimensional finite difference Laplacian inside inhomogeneous media (Eq. 2.24).
Fig. 2.10. Three-dimensional Laplacian on discrete rectilinear interfaces.
They may also be extended to Poisson’s equation in both magnetic and
dielectric media. For applications of such operators see Sec. 5.4.

2.3 Solution of Elliptic Difference Equations

The discretization of the elliptic-linear partial differential
continuous operator $L$ within $\Omega = RU \Gamma$ has led to specific finite
difference formulae valid at each base point $x_b$ of $\Omega = RU \Gamma$. Thus
the corresponding global finite difference operator $L_h$ can be expresed in the form

$$L_h[u] \triangleq \bigcup_{x_b \in \Omega} S[u(x_b)]$$  \hspace{1cm} (2.26)

where $S$ is a specific operator acting upon $u$ at $x_b$. Assuming that there
are $N$ base points with $N$ values of $u$, constituting an ordered set,
implies that $L_h$ is represented as an $N \times N$ (square) matrix, say $A$.

Eq. (2.26) can then be written as

$$L_h[u] = Au$$  \hspace{1cm} (2.27)

It is seen that the initial goal of the simplification of a
partial differential equation by the discretization process has been,
at least formally, achieved, for it should be easier to solve the
resulting matrix equation.

2.3.1 Matrix Equation and Its Properties

According to (2.27) a boundary-value problem $L[u] = g$ has its
analogue expressed as a matrix equation

$$Au = g$$  \hspace{1cm} (2.28)
where $\mathbf{\chi}$ is a column vector whose components are determined by $g$ and the boundary conditions. The matrix $[A] = (a_{ij})$ arising in such problems usually has the following properties:

a) $[A]$ is of large order $N$, where $N$ is 5,000 to 100,000 ($N = 108,000$ [Varga (1962) p. 1]). In order to obtain a higher accuracy of a finite difference solution $N$ should be increased; however, the numerical instability caused by round-off errors and/or the limited computer storage determine the upper bound on $N$;

b) $[A]$ is sparse, i.e., the ratio of nonzero elements to the $N \times N$ elements of $A$ (the density $d$ of $A$) is small. For example, the application of the 5-point operator yields at most five nonzero elements in a row of $A$, independently of the order $N$, thus for Laplace's equation with Dirichlet boundary conditions and 50 internal nodes, $d < 0.1$;

c) $[A]$ may be easily generated, and not stored, whenever it is needed.

The matrix $A$ may also possess additional important properties:

d) $[A]$ may be real symmetric (or almost symmetric), that is, $A = A^T$ where $A^T$ is the transpose of $A$. Neumann boundary conditions or unequal-arm operators at the near-boundary nodes or graded mesh make $A$ slightly nonsymmetric;

e) $a_{ii} > 0$;

f) $a_{ij} < 0$ for $i \neq j$;

g) $[A]$ may be positive definite, i.e., $A$ is a matrix whose eigenvalues are all positive or, equivalently, the inner product $(\chi, Ay) > 0, \chi \neq 0$. If all the eigenvalues of $A$ are non-negative, $A$ is said to be positive semidefinite;
h) [A] may be *diagonally dominant* [Varga (1962) p. 23; Fox (1962) p. 288], that is,

\[ |a_{i,i}| \geq \sum_{j=1, j\neq i}^{N} |a_{i,j}|, \quad \forall i \in N \tag{2.29a} \]

and

\[ |a_{i,i}| > \sum_{j=1}^{N} |a_{i,j}|, \text{ for at least one } i \in N; \tag{2.29b} \]

i) [A] may be *irreducible* [Varga (1962) p. 18], that is, there does not exist an \( N \times N \) permutation matrix\(^1\) \( \rho \) such that

\[
PAP^T = \begin{vmatrix} F & G \\ O & H \end{vmatrix}
\]

where \( F \) is an \( n \times n \) submatrix and \( H \) is an \((N-n) \times (N-n)\) submatrix, where \( 1 \leq n < N \), and \( N \geq 2 \). It implies that the equation (2.28) cannot be *partitioned* and, effectively, reduced to lower-order matrix equations. The partitioning can occur if \( \Omega \) consists of at least two sets unconnected within \( \Omega \); but this apparently means that there are at least two isolated problems. Note that the properties (d), (e), (h) and (i) imply (g);

j) [A] may possess Property A. This important property has been introduced by Young [Young (1971), p. 42, Young (1954)]. An \( N \times N \) matrix \( A \) has Property A if there exist two disjoint subsets

---

\(^1\) A permutation matrix is a square matrix with exactly one 1 in each row and column and 0's elsewhere. \( P^{-1} = P^T \) [Young (1971) p. 10] since \( P \) is orthogonal (\( PP^T = I \)).
\( S_1 \) and \( S_2 \) of \( W = \{1, 2, \ldots, N\} \) such that \( S_1 \cup S_2 = W \) and such that if either \( a_{ij} \neq 0 \) or \( a_{ji} \neq 0 \), then \( i \in S_1 \) and \( j \in S_2 \) or else \( i \in S_2 \) and \( j \in S_1 \). This definition is equivalent to stating that, by appropriate permutations of the rows and corresponding columns, the matrix \( A \) can be written in the form

\[
A = \begin{bmatrix}
D_1 & F \\
G & D_2
\end{bmatrix},
\]

where the \( D_i \) are square diagonal matrices, and \( F, G \) are arbitrary rectangular matrices. From a practical point of view, a matrix has Property A if it is generated by a finite difference operator which links a central node from one class to nodes of the opposite class where the classes refer to, for instance, "white" and "black" points interchanged in "chessboard" fashion and shown in Fig. 2.11a [Engel et al. (1959 p. 86)];

\[\text{(a)}\]

1 A matrix whose the only nonzero elements are located along the main diagonal is the diagonal matrix.
Fig. 2.11. Finite difference operators with Property A can be constructed on the two classes of points, o and x; (a) 5- and 13-point operators with Property A, (b) 9- and 13-point operators without Property A.

k) [A] may be a consistently ordered matrix. An N x N matrix A is a consistently ordered matrix (a CO-matrix) if there exist disjoint subsets $S_1, S_2, \ldots, S_t$ of $W = \{1, 2, \ldots, N\}$ such that $\bigcup_{k=1}^{t} S_k = W$ and such that if either $a_{ij} \neq 0$ or $a_{ji} \neq 0$ then $j \in S_{k+1}$ if $j > i$ or $j \in S_{k-1}$ if $j < i$, where $S_k$ is the subset containing $i$. Such a matrix results from the application of an appropriate scanning of the base-points in a finite difference scheme. Such a scanning must possess a consistent ordering [Young (1954) p. 108, Young (1971) Sec. 5.6] The raster¹ "(natural), diagonal², and "white black"³ scans have the consistent ordering.

1 The raster scan orders all the points in a page-wise fashion (Latin, Hebrew, Chinese, or from-bottom-to-top).

2 The diagonal scan sweeps all the diagonals of the mesh $\overline{N}$.

3 The "white-black" scan orders all the points $(x \o + ph, y \o + gh)$ with $p+g$ even ("white" points) before those with $p+g$ odd ("black" points).
If a matrix $A$ is a CO-matrix, it is possible to establish its Property A; if $A$ has Property A, then by permutation of its rows and columns one can obtain a CO-matrix. The consistently ordered matrices possessing Property A simplify and expedite an analysis of iterative schemes.

The properties (a) to (k) not only facilitate the analysis of the solution methods of (2.28) but also indicate the group of methods best suited for the solution. The literature on these methods is so enormous that no survey could hope to be exhaustive. The major representatives of two general classes of the numerical procedures, the direct methods and the indirect (iterative) methods, will be listed in the following sections.

2.4 Direct Methods

Direct methods yield an exact solution in a finite number of operations assuming that there is no round-off error in the computation process. The number of operations can be specified in advance. The residual error can also be specified from the finite length of a computer word and the type of operations involved in the computation.

The Cramer rule applied to large matrices is hopeless because the number of operations involved is phenomenal ($\approx 2(N+1)!$ multiplications for an $N \times N$ matrix). The methods based on triangularization (e.g., Gaussian elimination, Choleski decomposition), and diagonalization (e.g., Gauss-Jordan elimination) are much more economical.
2.4.1 Gaussian Elimination (Pivotal Condensation)

Let $A$ be nonsingular$^1$. Then $A$ may be expressed (or rearranged so that it may be expressed) as $A = LDU$, where $L$ is lower triangular$^2$, $D$ diagonal, and $U$ upper triangular$^3$ [Forsythe-Moler (1967) p. 27] (This is referred as the LDU or LU theorem). The basic idea of the elimination schemes is the factorization of $A$ as the product of $LU$ [Wilkinson (1965) p. 200]. The equation

$$AU = b$$

(2.32)

can be written

$$LUU = b$$

(2.33)

or

$$LU = b$$

(2.34)

where

$$U = b$$

(2.35)

The calculation of $L$ and $U$ and the solution of (2.34) is called forward elimination, and the solution of (2.35) is the back substitution. The computation of $L$ and $U$ alone is also referred to as triangular decomposition. The various elimination methods differ in the order in which the operations are performed in the forward elimination.

The Gauss elimination procedure annihilates successive elements of $A$ in such a way as to reduce $A$ to $U$. An appropriate multiple of

---

1 A matrix $A$ is nonsingular if $\det A \neq 0$; hence $A^{-1}$ exists.

2 A lower triangular matrix is a square matrix $L = (l_{ij})$ such that $l_{ij} = 0$ for $i < j$. (If $l_{ij} = 0 \forall i \leq j$, $L$ is strictly lower triangular).

3 An upper triangular matrix is a square matrix $U = (t_{ij}) \ni t_{ij} = 0 \forall i \geq j$. (If $t_{ij} = 0 \forall i \geq j$, $U$ is strictly upper triangular).
the first equation of the system (2.32) is added to all the other equations so that the \((N - 1)\) equations have zero coefficients for the \(u_1\) component. (If \(a_{11} = 0\), some interchange of equations is needed to obtain \(a_{11} \neq 0\)). Then an appropriate multiple of the next equation is added to all the remaining equations so that the terms with \(u_2\) are eliminated from these. (Note that an interchange may be needed). The successive equations used for partial zeroing the columns are called the pivotal equations and each coefficient responsible for the zeroing operation - the pivotal element. The process is repeated until the \(N\)'th equation has only one term with \(u_N\).

The back substitution (the solution of 2.35) is trivial. (i.e., \(u_N\) is found from the last equation, then substituted into the \((N - 1)\)th equation from which \(u_{N-1}\) is found, and so on). A compact notation for this procedure is given by Westlake (1968) p. 7 and Forsythe-Moller (1967) Sec. 9. The procedure requires \(\sim N^3/3\) multiplications to solve a system of order \(N\).

The above process suffers from round-off errors if the matrix \(A\) is ill conditioned\(^2\). Even if \(A\) is well conditioned, the

---

1 In practice, rows are interchanged so that \(|a_{s5}| \geq |a_{rs}|\) for \(r \neq s\).

2 A system of linear equations is ill-conditioned if the hyperplanes (i.e., the geometrical representation of the system) intersect at small angles, thus the round-off errors cause large changes in the solution (the intersection of the hyperplanes). The coefficient matrix associated with the system is then ill-conditioned. If the hyperplanes intersect at nearly 90°, the system and the matrix are well-conditioned. A matrix can also be ill-conditioned in other cases [see Wilkinson (1965) Sec. 4.4 and 4.11-12].
process can fail when a leading principal minor\(^1\) of A is singular or the process is unstable whenever a leading minor is ill-conditioned.

In order to avoid these inconveniences, other techniques have been incorporated in the process [Westlake (1968)]. Other techniques similar to Gaussian elimination are also available [Tinney-Meyer (1973); Bracha-Saylor (1973); Martin-Wilkinson (1967)].

2.4.2 Choleski Triangular Decomposition

This method [Wilkinson (1965) p. 229; Froberg (1965) p. 76 and 95] can be used if A is symmetric (or Hermitian) and positive definite.

The method is twice as fast as Gaussian elimination and is very stable.

Since A is symmetric, then \( U = L^T \) and from (2.33) \( A = LL^T \) or \( A = U^TU \). The elements of U are given by

\[
\begin{align*}
\text{for } j = 2, \ldots, N \\
\text{for } i = 2, \ldots, N \\
\end{align*}
\]

\[
t_{11} = \sqrt{a_{11}} \quad (2.36a)
\]

\[
t_{1j} = a_{1j} / t_{11} \quad \text{for } j = 2, \ldots, N \quad (2.36b)
\]

\[
t_{ii} = (a_{ii} - \sum_{k=1}^{i-1} t_{ki}^2)^{1/2} \quad \text{for } i = 2, \ldots, N \quad (2.36c)
\]

\[
t_{ij} = \frac{1}{t_{ii}} (a_{ij} - \sum_{k=1}^{i-1} t_{ki} t_{kj}) \quad \text{for } j = i+1, \ldots, N \quad \text{and } i = 2, \ldots, N \quad (2.36d)
\]

---

\(^1\) Principal minor of a matrix is a minor (the matrix obtained by deleting the i\(^{\text{th}}\) row and j\(^{\text{th}}\) column from the original matrix) whose principal-diagonal elements are on the principal (main) diagonal of the matrix.
The solution of \( A\psi = U^T\psi = \psi \) can be sought as \( \psi = U^{-1}U^{-T}\psi \) where \( U^{-T} \) is the transpose of \( U^{-1} \). The solution can be economically obtained [Della Torre-Kinsner (1973a)] by first finding \( U^{-1} = (t_{ij}^*) \) from the following formulae

\[
\begin{align*}
    t_{ii}^* &= \frac{1}{t_{ii}} \quad \text{for } i=1, \ldots, N \quad (2.37a) \\
    t_{ij}^* &= \frac{1}{t_{ii}} \left[ -\sum_{k=i+1}^{j} t_{ik}^* \right] \quad \text{for } i=(N-1), \ldots, 1 \quad \text{and } j=N, \ldots, i+1 \quad (2.37b)
\end{align*}
\]

Similar formulae can be found for \( L^{-1} \) [Wexler (1969) p. 418].

The procedures are described for symmetric positive definite matrices [Martin et al. (1965, 1966, 1968)] and for the same type of matrices but banded [Martin-Wilkinson (1965)].

2.4.3 Gauss-Jordan Elimination

The method is similar to Gaussian elimination with the extension that the elements above the main diagonal are also eliminated, thus the back substitution is not necessary. The method is based on similarity transformations which reduce \( A \) to diagonal form [Wilkinson (1965) p. 6]. The method is described by Westlake (1968) Sec. 2.4.1.

There are other methods which reduce a matrix to tridiagonal form (instead of the diagonal form). They will be discussed in connection with eigensolutions.

2.5 Iterative Methods

Iterative methods repeatedly apply the same algorithm with the purpose of improving an approximation to the solution of a problem.
Neglecting round-off errors, these methods yield the exact solution
(to the approximate problem) only as a limit of a sequence. The direct
methods may yield the same solution much faster. Thus the main, perhaps,
reason for using the iterative methods is the large order of $A$ in
(2.32) which, for accurate solutions, is so large that it may be im-
practical to store the matrix in the fast-access memory and use the
matrix in direct computation. There are many direct methods for
large sparse systems based on block elimination with an external slow
back-up storage [George (1971, 1972), Rose-Williams (1972), Moler
(1972), Irons (1970), Cantin (1971), Key (1973), Bates (1973),
Jennings-Tuff (1970), Tuff-Jennings (1973), Reid (1971)]. These
techniques can solve systems up to several thousand equations; however,
the procedures are time consuming (e.g., up to 4,000 sec. for a
system of 2,500 equations on an IBM 360 with IBM 2314 disk packs
[Cantin (1971)]. A general discussion on these and related methods
for sparse matrices is given by Walsh (1970), Gustavson (1971), and

An iterative method has to converge and the convergence should
be as rapid as possible, to be useful and effective. An iterative
method for solving (2.32) can be defined by

$$
\mathbf{u}^{(0)} = F_0 (A, \mathbf{b})
$$

(2.38a)

$$
\mathbf{u}^{(n)} = F_n (\mathbf{u}^{(0)}, \mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(n-1)}; A, \mathbf{b}).
$$

(2.38b)

The iteration is stationary, if $F_n$ is independent of $n$, and linear if,
for each $n$, $F_n$ is a linear function of $\mathbf{u}^{(0)}, \mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(n-1)}$. The
degree of a stationary iteration expresses the number of vectors $\mathbf{u}$. 
from the sequence \(\{y^{(i)}\}\), involved in constructing \(y^{(n)}\); e.g., the iteration

\[
y^{(n)} = F(y^{(n-1)}; A, \xi)
\]

is of degree one.

A linear stationary iterative method of first degree (2.39) has the form

\[
y^{(n)} = G y^{(n-1)} + \xi, \quad n > 1
\]

for some iteration matrix \(G\) and some vector \(\xi\). A nonstationary method has \(G\) and \(\xi\) dependent on \(n\). For a consistent method, \(y^{(n)} - y^{(n-1)} = 0\) and, therefore, (2.40) can be written as

\[
(I-G)y = \xi
\]

which is called the related linear system. The iterative method (2.40) is consistent [Young (1971) p. 64] with the system \(Ay = \xi\) (2.32) when \(S(A, \xi) \subseteq S(I-G, \xi)\), where \(S(A, b)\) is a set of solutions of (2.32), and \(S(I-G, \xi)\) is a set of solutions of (2.41). This consistency condition guarantees that if, at any iteration \(n\), a solution of (2.32) \(y = A^{-1} \xi\) is obtained, then all the subsequent iterants \(y^{(n+1)}\), \(y^{(n+2)}\), ... remain the same. It is, however, possible that the sequence \(\{y^{(i)}\}\) defined by (2.40) converges to a limit \(\overline{y} \in S(I-G, \xi)\) which is not equal to \(y\). Thus, it is important that whenever the sequence \(\{y^{(i)}\}\) converges, it converges to a solution of (2.32), i.e., \(S(I-G, \xi) \subseteq S(A, \xi)\). The latter implies that the method is reciprocally consistent. Of special interest are completely consistent methods for which \(S(I-G, \xi) = S(A, \xi)\). It is seen from (2.41) that a method is
completely consistent if and only if there exists a nonsingular matrix $M$ such that

$$M = (I-G)A^{-1} \text{ and } \xi = M\xi.$$  \hfill (2.42)

Thus Eq. (2.40) takes the form

$$\mu^{(n)} = G\mu^{(n-1)} + M\xi.$$  \hfill (2.43)

The fundamental question of convergence of (2.43) can readily be answered by examining the error vector $e^{(n)}$ defined by

$$e^{(n)} = \mu^{(n)} - \bar{\mu}$$  \hfill (2.44)

where $\bar{\mu} = A^{-1}\mu$ is the solution vector. Using the complete consistency condition $\bar{\mu} = G\bar{\mu} + M\xi$ yields

$$e^{(n)} = G\mu^{(n-1)} + M\xi - G\bar{\mu} - M\bar{\mu}$$

$$= G_{\bar{\mu}}^{(n-1)}$$

$$= G^ne^{(o)}.$$  \hfill (2.45)

or for nonstationary methods

$$e^{(n)} = \left[ \bigcap_{p=1}^n G_p \right] e^{(o)}.$$  \hfill (2.46)

Thus, $e^{(n)} \to 0$ as $n \to \infty$ for all $e^{(o)}$ if and only if $G^n \to 0$ or $G_n \to 0$ as $n \to \infty$. Furthermore, $G^n \to 0$ as $n \to \infty$ if and only if

$$\rho(G) < 1$$  \hfill (2.47)

where $\rho(G)$ is the spectral radius$^1$ of the matrix $G$, or $G_n \to 0$ as

---

$^1$ The spectral radius of a matrix $G$ is defined as [Young (1971) p. 15]

$$\rho(G) = \max_{\lambda \in S(G)} |\lambda|$$

where $S(G)$ is the set of all eigenvalues of $G$ (the spectrum of $G$).
\[ n \to \infty \text{ if and only if } \quad \rho_n(G_n) < 1 \]  
(2.48)

where \( \rho_n(G_n) \) is the *average spectral radius*\(^1\). If weak inequality is allowed in (2.47) and (2.48) then the methods are said to be *weakly convergent*. In this case, however, \( \chi^{(n)} \) must converge but its limit need not be zero.

A measure of the rate of convergence can be established by further analysis of (2.45). Taking a norm of both sides of (2.45) yields

\[
\frac{||\chi^{(n)}||}{||\chi^{(0)}||} \leq ||G^n|| = \sup_{\chi^{(0)} \neq 0} \frac{||\chi^{(n)}||}{||\chi^{(0)}||} \quad (2.49)
\]

which indicates that \( ||G^n|| \) can be used as a measure of the initial error reduction after \( n \) iterations. Let \( \alpha (\alpha < 1) \) be a desired reduction of \( ||\chi^{(0)}|| \). Then one should iterate until

\[
||G^n|| \leq \alpha \quad (2.50)
\]

which is achievable since \( ||G^n|| \to 0 \) as \( n \to \infty \) as if \( \rho(G) < 1 \). Eq (2.50) can be rewritten into form

\[
n > \frac{-\log \alpha}{n^{-1} \log ||G^n||} \quad (2.51)
\]

where \( n^{-1} \log ||G^n|| \) can be defined as the *average rate of convergence*

---

\(^1\) The average spectral radius is defined as [Young (1971) p. 299]

\[
\rho_n(G_n) = [\rho(G_n)]^{1/n}
\]

where \( \rho(G_n) \) is the spectral radius of \( G_n \). Other concepts of the spectral radius of \( G_n \) are also given by Young (1971) p. 299.
\[ r_n(G) \overset{\Delta}{=} -n^{-1} \log ||G^n|| \]  
(2.52)

Since \( \rho(G) = \lim_{n \to \infty} (||G^n||)^{1/n} \) [Young (1971) p. 87], another quantity, the asymptotic average rate of convergence (shortly: rate of convergence) can be defined as

\[ r(G) \overset{\Delta}{=} \lim_{n \to \infty} r_n(G) = -\log \rho(G) \]  
(2.53)

Using \( r(G) \) in (2.51) gives an underestimated number of iterations needed for convergence, however, it is easier to compute \( \rho(G) \) in (2.53) rather than \( ||G^n|| \) in (2.51). For a nonstationary method the rate of convergence is given by

\[ r(G_n) \overset{\Delta}{=} -\log \rho(G_n) \]  
(2.54)

This rate also depends on the definition of \( \rho(G_n) \) [Young (1971) p. 299].

It should be noted that the iterative methods apply to either the systems of equations given explicitly or generated one or more equations at a time. They may be classified into two categories, point iterative and block iterative, depending on the number of equations (or systems) involved in the explicit nature of the calculation.

2.5.1 Point Iterative Methods

The structure of the basic iterative methods involves a partition (splitting) of \( A \) into the form

\[ A = C_L + D + C_U \]  
(2.55)

where \( C_L \) and \( C_U \) are strictly lower and strictly upper triangular matrices, and \( D = \text{diag } A \). The methods converge if \( A \) has properties (e), (f), (h), and (i) of Sec. 2.3.1. This requirement is usually

\[ ^1 \text{ Properties (e) and (f) are not necessary [Redish (1961) Ch. 2].} \]
satisfied by a reasonably posed elliptic problem.

2.5.1.1 Jacobi Method

Substituting (2.55) into \( Au = \kappa \) yields

\[
u = -D^{-1}(C_L + C_U)\psi + D^{-1} \kappa.
\]

(2.56)

Comparing (2.56) with (2.43) shows that

\[
J \equiv -D^{-1}(C_L + C_U) \quad \text{and} \quad M \equiv D^{-1}.
\]

(2.57)

Since \( D \) has nonvanishing diagonals, \( D^{-1} \) exists and 2.42 holds, thus the method is completely consistent with (2.32). The convergent process

\[
\psi^{(n)} = J\psi^{(n-1)} + M\psi
\]

(2.58)

is called the Jacobi method. It is seen that the vector \( \psi^{(n+1)} \) is replaced by the improved vector \( \psi^{(n)} \) in one step of the iteration (or simultaneously), i.e., the replacement must be deferred until all the components of \( \psi^{(n)} \) are computed. This is the reason that this method is also called the iteration by total step and the method of simultaneous displacements. The algebraic form of (2.58) is

\[
u^{(n)}_i = \sum_{j=1}^{N} d_{ij} \nu^{(n-1)}_j + c_i, \quad \forall i \in N,
\]

(2.59a)

or

\[
u^{(n)}_i = -\sum_{j=1}^{N} \frac{a_{ij}}{a_{ii}} \nu^{(n-1)}_j + \frac{b_i}{a_{ii}}, \quad \forall i \in N.
\]

(2.59b)

When solving field problems by the Jacobi method one can use the 5-point formula

\[
u^{(n)}_i = \frac{1}{4} \sum_{k=1}^{4} \nu^{(n-1)}_k, \quad \forall i \in N
\]

(2.60)
The method has comparatively slow convergence and requires storage for two vectors $\mathbf{u}^{(n)}$ and $\mathbf{u}^{(n-1)}$. The explicit convergence rates of the Jacobi and other methods are given in Appendix 5.

2.5.1.2 Jacobi Overrelaxation Method (JOR)

An extension of the Jacobi method is the simultaneous over-relaxation method (JOR method) in which a new vector $\mathbf{u}^{(n)}$ is obtained by the simultaneous projection of $\mathbf{u}^{(n)}$ and $\mathbf{u}^{(n-1)}$ according to the formula

$$\mathbf{u}^{(n)} = \mathbf{u}^{(n-1)} + \omega \mathbf{d}^{(n)} \quad (2.61)$$

where

$$\mathbf{d}^{(n)} = \mathbf{u}^{(n)} - \mathbf{u}^{(n-1)} \quad (2.62)$$

is the displacement vector, $\mathbf{d}^{(n)}$ is obtained from the Jacobi method and $\omega$ is a real overrelaxation parameter. Substituting (2.56) into (2.61) gives

$$\mathbf{u}^{(n)} = J_\omega \mathbf{u}^{(n-1)} + M_\omega \mathbf{d} \quad (2.63)$$

where

$$J_\omega = -\omega D^{-1}(C_L + C_U) + (1-\omega)I \quad \text{and} \quad M_\omega = \omega D^{-1} \quad (2.64)$$

For $\omega \neq 0$ the method is completely consistent. For $\omega = 1$, it reduces to the Jacobi method. Overcorrecting of $\mathbf{u}^{(n)}$ occurs if $\omega > 1$, and undercorrecting takes place for $\omega < 1$. The method converges for $0 < \omega \leq 2$ [Young (1971) p. 109].

2.5.1.3 Gauss-Seidel Method

This method is similar to the Jacobi scheme except that each computed element $u^{(n)}_i$ immediately replaces $u^{(n-1)}_i$. Eq. (2.57) becomes
\[ u^{(n)} = -D^{-1}C_L u^{(n)} - D^{-1}C_U u^{(n-1)} + D^{-1} \mathbf{r} \]  

(2.65)

or

\[ u^{(n)} = L u^{(n-1)} + M \mathbf{r} \]  

(2.66)

where

\[ L \equiv -(D + C_L)^{-1}C_U, \quad \text{and} \quad M \equiv (D + C_L)^{-1} \]

(2.67)

The method is completely consistent since \( \det (D + C_L) \) exists and therefore \( M \) exists.

The algebraic form of (2.66) is

\[ u_i^{(n)} = \sum_{j=1}^{i-1} d_{ij} u_j^{(n)} + \sum_{j=i+1}^{N} d_{ij} u_j^{(n-1)} + c_i, \quad \forall \ i \in N \]  

(2.68a)

or

\[ u_i^{(n)} = -\sum_{j=1}^{i-1} \frac{a_{ij}}{a_{ii}} u_j^{(n)} - \sum_{j=i+1}^{N} \frac{a_{ij}}{a_{ii}} u_j^{(n-1)} + \frac{b_i}{a_{ii}}, \quad \forall \ i \in N \]  

(2.68b)

This equation can be generated by the use of (2.60). The convergence is approximately twice as rapid as that of the Jacobi method (ref. Appendix 5). Only one vector need be available at each iteration. The method is also known as the method of Liebmann, successive displacements, or iteration by single step.

2.5.1.4 Successive Overrelaxation Method (SOR)

The Gauss-Seidel method can be accelerated by JOR in (2.61)

\[ u_i^{(n)} = (1-\omega) u_i^{(n-1)} + \omega \overline{u}_i^{(n)} \]  

(2.69)

where \( \overline{u}_i^{(n)} \) is computed by the Gauss-Seidel scheme. Substituting (2.65) into the vector form of (2.69) yields

\[ u_i^{(n)} = L_\omega u_i^{(n-1)} + M_\omega \mathbf{r} \]  

(2.70)
where \( L_\omega = (D + \omega C_L)^{-1} \) \( [(1-\omega)D + \omega C_U] \), and \( M_\omega = (D + \omega C_L)^{-1} \) \( (2.71) \)

If \( A \) is nonsingular and \( \omega \neq 0 \) the method is completely consistent.

Application of \( \omega = 1 \) reduces (2.70) to (2.66), \( \omega > 1 \) causes over-relaxation, and \( \omega < 1 \) underrelaxation.

The algebraic form of (2.70)

\[
  u_i^{(n)} = (1-\omega)u_i^{(n-1)} + \omega \left\{ \sum_{j=1}^{i-1} d_{ij}u_j^{(n)} + \sum_{j=i+1}^{N} d_{ij}u_j^{(n-1)} + c_i \right\}
\]

or

\[
  u_i^{(n)} = (1-\omega)u_i^{(n)} + \omega \left\{ -\sum_{j=1}^{i-1} a_{ii}u_j^{(n)} + \sum_{j=i+1}^{N} a_{ii}u_j^{(n-1)} + \frac{b_{ii}}{a_{ii}} \right\}
\]

The equation can be generated by (2.60) and (2.69). The SOR method requires one vector to be available. The process converges for \( 0 < \omega < 2 \) [Young (1971) p. 109; This reference is essentially devoted to the SOR method].

Another version of SOR is the modified SOR method (MSOR) [Young (1971) Ch. 8 and 10] in which the SOR procedure is applied to the "white" equations with one value of the relaxation factor \( \omega_1 \), and to the "black" equations with another value \( \omega_2 \) yielding

\[
  u_i^{(n)} = L_{\omega_1, \omega_2} u_i^{(n-1)} + \xi_{\omega_1, \omega_2}
\]

2.5.1.5 Other Point Iterative Methods

Several stationary, nonstationary and semi-iterative methods shall be discussed in this section.

a) Gradient Methods

It is practical to solve \( A\psi = \phi \), where \( A \) is symmetric and
positive definite, by minimizing the real-value quadratic form

\[ Q(\psi) = \psi^T A \psi - 2 \kappa^T \psi \]  

(2.74a)

\[ = (\psi^T A^{-1} \kappa) A (\psi^T A^{-1} \kappa) - \kappa^T A^{-1} \kappa \]  

(2.74b)

using a suitable gradient method because \( Q(\psi) \) attains its minimum value \((-\kappa^T A^{-1} \kappa)\) when \( \psi = A^{-1} \kappa \), the solution. Using the gradient of \( Q(\psi) \) at \( \psi^{(n-1)} \)

\[ -vQ(\psi^{(n-1)}) = -2(\Delta \psi^{(n-1)} - \kappa) \]  

(2.75)

leads to the iteration in the form

\[ \psi^{(n)} = \psi^{(n-1)} + s_{n-1} (\kappa - A \psi^{(n-1)}) \]  

(2.76)

where \( s \) is a positive number. For \( s_{n-1} = s \) Eq. (2.76) can be written

\[ \psi^{(n)} = G \psi^{(n-1)} + s \kappa \]  

(2.77)

where \( G = (I - sA) \). The method is completely consistent if \( s \neq 0 \) and \( A \) is nonsingular. The fastest possible convergence is obtained if the maximum eigenvalue of \( G \), \( \lambda_{\max} \), takes a minimum value, i.e.,

\[ \max_{i} |1 - s \mu_{i}| = \min \]  

(2.78)

where \( \mu_{i} \)'s are eigenvalues of \( A \). Various strategies for choosing \( s \) may be applied such that (2.78) holds [ref. Forsythe-Wasow (1960) p. 225].

This method is also called the stationary Richardson's method [Young (1971) p. 74].

b) Richardson's Method.

Richardson's method is based on Eq. (2.76). Since \( s_{n-1} \) is permitted to vary with \( n \), the method is nonstationary and can be written in the form
\[ y^{(n)} = G_{n-1} y^{(n-1)} + s_{n-1} y \]  
(2.79)

where

\[ G_{n-1} = \prod_{p=0}^{n-1} (I - s_p A) = P_n(A) \]  
(2.80)

and \( P(x) \) is a polynomial of degree \( n \) with zeros \( s_p^{-1} \) with the following constraint

\[ P_n(0) = 1 \]  
(2.81)

The strategies for selecting successive \( s_p \) are based on knowledge of the minimum, \( a \), and the maximum, \( b \), values of the eigenvalues of \( A \).

Richardson suggested a fairly uniform distribution of zeros \( s_p^{-1} \) of \( P_n(x) \) over \( a < x < b \), which makes \( |P_n(x)| \) small for \( x = [a,b] \). A better approach (for large \( n \)) is to obtain

\[ \max_{a<x<b} |P_n(x)| = \min \]  
(2.82)

by means of the Markov polynomial

\[ P_n(x) = \frac{T_n \left( \frac{b+a-2x}{b-a} \right)}{T_n(y_o)}, \quad y_o = \frac{b+a}{b-a} > 1 \]  
(2.83)

where \( T_n(y) = \cos(n \cos^{-1} y) \) is the ordinary Chebyshev polynomial on \( y = [-1, 1] \). It can be shown [Forsythe-Wasow (1960) p. 228] that

\[ \max_{a<x<b} |P_n(x)| \leq 2 \left[ y_o - (y_o^2 - 1)^{1/2} \right]^n \]  
(2.84)

from which the average rate of convergence,

\[ r(G_{n-1}) = -\log(\|e^{(n)}\|/\|e^{(o)}\|)^{1/n} \],

is bounded by
\[ r(G_{n-1}) \leq \log \left[ \gamma_o - \left( y_0^2 - 1 \right)^{1/2} \right] \leq 2p^{-1/2} \]  

(2.85)

where \( P \) is Todd's \( P \)-Condition number

\[ P = \frac{\max_{p} \mu_p}{\min_{p} \mu_p} < \frac{b}{a} \]  

(2.86)

The Richardson method is approximately twice slower than SOR, is more complicated than SOR, and requires more storage (see Appendices 4, 5, and 6). Furthermore, this method is sensitive to rounding errors, especially for large \( n \) [Young (1971) p. 365; Forsythe-Wasow (1960) p. 229].

It is evident that one cannot compute \( s_p \) exactly because it requires that \( n \rightarrow \infty \). Fixing \( n \) (e.g., \( n_F = 20 \)) leads to determining \( s_p(p=1, ..., n_F) \) from the tabulated zeros of the Chebyshev polynomials and then computing \( y_p^{(p)}(p=1, ..., n_p) \) from Eq. (2.76). If \( e^{(n_F)} \) is not sufficiently small, the cycle is repeated with the same values of \( s_p \). Such processes are called semi-iterative (SI).

c) Semi-Iterative Methods

For a given linear stationary method of first degree, one can find an associated nonstationary method (semi-iterative method) which converges faster than the given method. Thus, to solve \( A\varphi = A \) one needs a linear stationary iteration

\[ \varphi(n) = G\varphi(n-1) + \varphi \]  

(2.87)

and an associated nonstationary method

\[ \chi(n) = \sum_{m=0}^{n} a_n^{(m)} u(n) \]  

(2.88)

called a semi-iterative method with respect to the iterative method of
If \( a^{(m)}_n = 0 \) \( \forall m < n \), and \( a^{(n)}_n = 1 \) \( \forall n \), then \( \chi^{(n)} = u^{(n)} \). The objective of this method is to obtain a finite sequence of vectors \( \chi^{(1)} \) by means of (2.87), and then to combine these vectors by means of (2.88) in such a way that the new sequence of vectors \( \chi^{(1)} \) tends rapidly to the unique solution \( \bar{u} \) [Varga (1962) p. 133]. One of such methods, related to the Richardson procedure, is the cyclic Chebyshev semi-iterative (CCSI) method [Varga (1962) p. 128; Young (1971) Sec. 11.5].

It has been shown [Young (1971 Ch. 11)] that other semi-iterative methods, based on the Jacobi, JOR, Gauss-Seidel, SOR, and MSOR methods, can be constructed. Using a proper SI method can often yield an order-of-magnitude improvement in the convergence. Furthermore, when \( A \) is positive definite but does not possess Property A (SOR is not effective), the use of SI methods based on the Jacobi or Gauss-Seidel may still be effective.

d) **Symmetric SOR (SSOR) Method**

This method consists of two half iterations. The first half iteration is the same as the SOR or Gauss-Seidel method, and the second is the SOR or G-S with the equations in reverse order. Thus \( \chi^{(n-1/2)} \) is determined from \( \chi^{(n-1)} \) (forward) and then \( \chi^{(n)} \) from \( \chi^{(n-1/2)} \) (backward). This process is equivalent to squaring the iteration matrix of the SOR or G-S methods. The method is completely consistent. If \( A \) is positive definite and \( 0 < \omega < 2 \), the method converges. The SSOR method with the best \( \omega \) is usually slower than the SOR with the best \( \omega \) but the eigenvalues of the iteration matrix are always real and positive, therefore, the SSOR can be accelerated by the use of SI methods.
so that it converges faster than the SOR method (without semi-iteration). The symmetric Gauss-Seidel method was first used by Aitken (1950); the SSOR method is discussed by Young (1971) Ch. 15.

When the relaxation factor is \( \omega = \omega_1 \) for the first half iteration and \( \omega = \omega_2 \) for the second half iteration, then the resulting method is known as the \textit{unsymmetric SOR} (USSOR) method [Young (1971) Sec. 15.6]. When the symmetric and unsymmetric procedures are applied to the modified SOR (MSOR), the resulting methods are called the \textit{symmetric MSOR} (SMSOR) method and the \textit{unsymmetric MSOR} (USMSOR) method, respectively [Young (1971) Sec. 15.7]. However, the USSOR, SMSOR, and USMSOR methods cannot converge faster than the SOR method and, therefore, are not practical [Lynn (1964); Young-Kincaid (1969); Young (1970a); Kincaid (1972, 1973)].

2.5.1.6 Optimum Relaxation Factor

The SOR method converges for the overrelaxation factor in the range \( 0 < \omega < 2 \). The most rapid convergence occurs somewhere in the interval \( 1 < \omega < 2 \). A one dimensional search for this optimum value is not economical because the convergence rate is very sensitive to \( \omega \) (several decimal places are needed for a good estimate of the optimum relaxation factor \( \tilde{\omega} \)).

Young (1954) and Frankel (1950) gave the theory for choosing \( \tilde{\omega} \). Young's theory applies to matrices \( A \) possessing Property A. Let the iteration matrix \( J \) of the Jacobi method have eigenvalues \( \mu_1 \). If \( A \) has Property A, then \( \mu_1 \)'s are either zero or occur in pairs \( \pm \mu_1 \). If \( A \) possesses Property A and is consistently ordered, then \( \eta(\eta \neq 0) \) is an eigenvalue of the SOR iteration matrix \( L_\omega \) if and only if there exists
an eigenvalue \( \mu_1 \) of \( J \) such that

\[
(n+\omega-1)^2 = \omega^2 \mu_1^2 \eta
\]  

(2.89)

This fundamental relationship may yield \( \tilde{\omega} \) when \( J \) is symmetric (\( \mu_1 \)'s are real) and also for the case of complex \( \mu_1 \) [Forsythe-Wasow (1960) pp. 250-266; Young (1971) Ch. 6; Young-Edison (1970)]. When \( J \) is symmetric, then the optimum value \( \tilde{\omega} \) of \( \omega \), such that the spectral radius \( \rho(L_\omega) \) is a minimum, satisfies

\[
\tilde{\omega} = \frac{2}{1 + [1 - \rho^2(J)]^{1/2}}
\]  

(2.90)

For \( \omega = \tilde{\omega} \) all eigenvalues of \( L_\omega \) have modulus \( \tilde{\omega} - 1 \), that is,

\[
\rho(L_\omega) = \tilde{\omega} - 1
\]  

(2.91)

Fig. 2.12 shows the behaviour of \( \rho(L_\omega) \) as a function of \( \omega \) for various values of \( \rho(J) \). It is seen that for \( \omega \) slightly below \( \tilde{\omega} , \rho(L_\omega) \) changes rapidly (the slope approaches \( \infty \) at \( \tilde{\omega} \)). Evidently, an underestimation of \( \tilde{\omega} \) causes a much larger relative decrease in the rate of convergence than a corresponding overestimation of \( \tilde{\omega} \). For example, using \( \omega = 1.72 \) instead of 1.72945 for \( \tilde{\omega} \) results in relative decrease of 25% in the rate of convergence, while using \( \omega = 1.74 \) results in a relative decrease of only 5%.

Eq. (2.89) shows that for \( \omega = 1 \) (SOR + GS) \( \eta_1 = \mu_1^2 \) and hence \( \rho(L) = \rho^2(J) \) from which \( r(L) = 2r(J) \), i.e., the rate of convergence of the Gauss-Seidel method is twice that of the Jacobi method when \( J \) has Property A and is consistently ordered. Combining (2.91) and (2.90) may lead to \( \rho(L_\omega) \sim 1 - 2h \) as \( h \to 0 \), indicating that the convergence rate is
approximately 2\(h\), a substantial reduction in computing time with respect to the Gauss-Seidel method.

In practice, \(\rho(J)\) is not known in advance. Thus various practical methods for estimating \(\bar{\omega}\) basically differ in estimating \(\rho(J)\). It would be better to overestimate \(\bar{\omega}\) rather than to underestimate it by the same amount due to the slope of the curve in Fig. 2.12 which equals 1 for \(\omega > \bar{\omega}\) but is infinite for \(\omega = \bar{\omega}\). However, for \(\omega > \bar{\omega}\) the corresponding eigenvalues \(\mu\) are complex and one can estimate \(\bar{\omega}\) by approaching it from below.

Carré (1961) replaced (2.90) by

\[
\omega_o = 2(1 + \left[1 - \left(\frac{\eta_{\text{max}} + \omega - 1}{\eta_{\text{max}}^2 \omega^2}\right)^2\right]^{1/2})^{-1}
\]

(2.91)

which gives \(\omega_o\) in terms of the largest eigenvalue of \(L^\omega\) for any \(\omega\) in the range \(0 < \omega < \omega_o\). Carré's algorithms is given in Appendix 7.

Kulsrud (1961) uses essentially Carré's process except that he does not decreases the value of \(\omega_o\) computed from (2.91) (i.e., Eq. A7.8 is not used). Reid (1966) also uses Carré's procedure but he substitutes the criterion for terminating the optimization process (A7.10) by another one, involving a measure of convergence of all the vectors from the basic optimization cycle.

Another approach to the determination of \(\bar{\omega}\) [Randall (1968); Lloyd-McCallion (1968)] is based on the substitution of an equivalent rectangle for the irregular region and then computing an estimate of \(\bar{\omega}\). Although this estimate is known at the beginning of the SOR process, the fastest convergence cannot be guaranteed because the estimate is
inaccurate.

The simplest, one dimensional search for \( \omega \) [Niki (1971)] is shown to fail when solving mixed boundary-value problems or very large problems [Radley et al. (1971)].

Various methods are available for determining the optimum parameters for the semi-iterative methods [ref; Young (1971); Rigler (1965); Wachspress (1966) Ch. 5; Too (1963-4); Martin-Too (1961); Miranker (1972)].

2.5.1.7 Scannings in SOR

It has been shown [Young (1971) Th. 2.1] that all the consistently ordered matrices have the same eigenvalues. Furthermore, Varga (1969) proves Young's conjecture that no inconsistent ordering for Laplace's equation has a rate of convergence as great as that common to all consistent orders. However, Forsythe-Wasow (1960) p. 259 stated that although the asymptotic rate of convergence is the same for all consistent orderings, experimental results have shown that the actual reduction of error for \( \omega = 1 \) after approximately \( n(10^{-4})/4 \) iterations varies greatly among different consistent orderings \( n(10^{-4}) \) is the number of iterations required to reduce the initial error by a factor of the order of \( 10^{-4} \).

The phenomenon indicated by Forsythe-Wasow may be due to the fact that for less than \( n \) iterations, some eigenvalues responsible for the convergence do not achieve their asymptotic state and, hence, the asymptotic definition of the rate of convergence cannot apply for
n < m. The author of this thesis observed this phenomenon when solving large problems only.

Nevertheless, the difference in convergence, for small number of iterations, was principally observed for both large and small problems when using consistent orderings and nearly-consistent orderings, the latter yielding a substantially higher convergence. Fig. 2.13 shows a typical behaviour of a potential value at an internal point for three consistent orderings: raster, diagonal, and "white-black", and two nearly-consistent orderings: "spiral" and "contour" [Della Torre-Kinsner (1973b) and Kinsner-Della Torro (1973b)]. The potential variation was obtained by the Gauss-Seidel method, and recorded at the point \((M, N) = (8, 13)\) within the boundary shown in Fig. 2.14. The value of \(u\) for the contour scan is more than two times larger as compared to the consistent orderings. The difference diminishes when the number of iterations increases but the value of \(u\) for consistent orderings is still smaller than the values for the new scans. Thus the solution is obtained faster for the nearly-consistent scans.

The raster scan is shown in Fig. 2.15, the spiral scan in Fig. 2.16, a double spiral (pertinent to the SSOR method) in Fig. 2.17, and the contour scan in Fig. 2.18. Some comments on both spiral and contour scans are presented in Appendix 8. Several results obtained by the application of the raster, spiral and contour scans are presented in Sec. 5.6.
Fig. 2.13. Convergence of a potential value at the point $(8,13)$ of the problem shown in Fig. 2.14, for different scanning methods.

Fig. 2.14. Model problem associated with Fig. 2.13.
Fig. 2.15. Raster (natural) scan.
Fig. 2.18. Optimal contour scan.
2.5.2 Group and Block Iterative Methods

The point iterative methods involve explicit processes for modifying the approximate solution at a single point within $\bar{\Omega}$. It is natural to modify the solution at more than one point at a time. It implies that the individual components of $\mathbf{v}^{(i)}(k)$ are implicitly related to the other components of the same group $k$. Since each group (system) can be solved simultaneously, one system at a time, the resulting methods belong to implicit iterative methods. If the grouping of the equations into subsets is arbitrary (but fixed for the whole process of solution) then the associated methods are called group iterative methods. Procedures based on a partitioning\(^1\) are called block iterative methods.

Direct methods are usually involved in solving the subsystems because the subsystems are much smaller and may even be tridiagonal, reducing the number of multiplications from $O(n^3)$ (Gaussian elimination) to $O(n)$.

The point iterative methods are convertible to group (block) methods. Some commonly used block methods shall be briefly reviewed. Analysis of the group methods is given by Young (1971) Ch. 14 and Varga (1962) Sec. 6.4.

\(^1\) An ordered set $N = \{1, 2, \ldots, n\}$ is partitioned if $k$ disjoint subsets contain the members of $N$ such that $N_1 = \{1, 2, \ldots, n_1\}$, $N_2 = \{n_1, \ldots, n_2\}$, $\ldots$, $N_k = \{n_{q-1}, \ldots, n\}$ where $n_i \in N$ for $i \leq q$ and $n_i < n_i+1$. If the mesh points are numbered in the natural ordering, their grouping of the points in a row corresponds to a partitioning.
2.5.2.1 Line Iterative Methods

Let all the base points in \( \mathbf{n} \) have the natural ordering and let the partitioning of \( \mathbf{n} \) be determined by \( R \). Then each disjoint subset contains points from a row (line). Each subsystem of equations corresponding to a subset shall be solved by a suitable direct method, thus improving the values of the approximate solution simultaneously on an entire line of points.

To solve Laplace's equation on \( R \) using the equal-arm 5-point operator one should apply the following formulae to a line (see Appendix 2 for the numbering of points in a molecule)

\[
4u_o^{(n)} = u_1^{(n)} + u_2^{(n)} + u_3^{(n-1)} + u_4^{(n-1)}
\]

for the \textit{Jacobi line iteration},

\[
4u_o^{(n)} = u_1^{(n)} + u_2^{(n)} + u_3^{(n-1)} + u_4^{(n)}
\]

for the \textit{Gauss-Seidel line iteration}, or

\[
u_o^{(n)} = u_o^{(n-1)} + \omega[u_o^{(n)} - u_o^{(n-1)}]
\]

where

\[
\overline{u}_o^{(n)} = [\overline{u}_1^{(n)} + \overline{u}_2^{(n)} + \overline{u}_3^{(n-1)} + \overline{u}_4^{(n)}]/4
\]

for the \textit{successive line overrelaxation} (SLOR).

Since each point is coupled only with two other points on a given row and column, the corresponding \( N_1 \times N_1 \) matrix, for a given row containing \( N_1 \) points, is tridiagonal (this feature is preserved for the 9- and 13-point operators). The solution of the subsystem can then be obtained directly by means of the fast Thomas algorithm (see Appendix 9) [Ames (1969) Sec. 2.3]
The convergence rate of the line Jacobi method is the same as that of the point Gauss-Seidel method. The optimum SLOR [Evans (1964)] is faster than the corresponding optimum SOR by a factor of $\sqrt{2}$.

This method is similar to the Russian "method of lines" where a differential equation is approximated by a coupled system of ordinary differential equation analytically for each line [Berezin-Zhidkov (1965) Sec. 10.8]

### 2.5.2.2 Alternating Direction Implicit (ADI) Method

The SLOR method scans the lines (treated as units) from 1 to K and repeats this cycle at the next iteration. Convergence may be improved by first scanning the rows and then the columns. One iteration involves both complete scans (hence the name of the method).

The method has been introduced by Peaceman-Rachford (1955) (PRADI) and by Douglas-Rachford (1956) (DRADI). Both processes are similar. The PRADI iteration for Laplace's equation approximated by an equal-arm 5-point operator in a rectangular domain proceeds from $u_o^{(n)}$ to

$$u_o^{(n+1/2)} = u_o^{(n)} + \rho(n) \left[ u_1^{(n+1/2)} + u_2^{(n+1/2)} - 2u_o^{(n+1/2)} \right]$$

$$+ \rho(n) \left[ u_3^{(n)} + u_4^{(n)} - 2u_o^{(n)} \right] \quad (2.95)$$

by a single row iteration followed by a single column iteration determined from

$$u_o^{(n+1)} = u_o^{(n+1/2)} + \rho(n) \left[ u_1^{(n+1/2)} + u_2^{(n+1/2)} - 2u_o^{(n+1/2)} \right]$$

$$+ \rho(n) \left[ u_3^{(n+1)} + u_4^{(n+1)} - 2u_o^{(n+1)} \right] \quad (2.96)$$
The iteration parameters, \( p^{(n)} \), may depend upon \( n \) but are the same for both parts of an iteration. The matrices resulting from this formulation may be inverted by Thomas algorithm.

Various methods of determining optimum parameters are described [Wachspress - Habetler (1960), Birkhoff et al. (1962), Hadjidimos (1971)]. The ADI method is systematically treated by Varga (1962) Ch. 7, Wachspress (1966) Ch. 6, Young (1971) Ch. 17.

### 2.6 Matrix Eigenvalue Equation

The finite difference approximation to boundary value problems lead to Eq. (2.32). Many physical problems (see Sec. 1.11) belong to eigenvalue problems whose solutions must satisfy the differential equation

\[
L_{2m}[u] = \lambda L'_{2n}[u], \quad u \in \mathbb{R}
\]  

(2.97a)

and the boundary conditions

\[
M'_i[u] = \lambda M_i'[u], \quad i = 1, 2, \ldots, m, \quad u \in \Gamma
\]  

(2.97b)

The operators \( L_{2m} \) and \( L'_{2n} \) are linear homogeneous differential operators of order \( 2m \) and \( 2n \), respectively, with \( m > n \). If \( \lambda \) does not appear in (2.97b) then the eigenvalue problem is self-adjoint if for any two functions \( \psi, \phi \) which satisfy the boundary conditions the following relations hold.

\[
< L[\psi], \phi > = < \phi, L[\psi] >
\]  

(2.98a)

\[
< L'[\psi], \phi > = < \phi, L'[\psi] >
\]  

(2.98b)
If for any such function \( \psi \)
\[
< L[\psi], \psi > = \begin{cases} 
\neq 0 , & \psi \neq 0 \\
= 0 , & \psi = 0 
\end{cases}
\]  \hspace{1cm} (2.99)

holds then the operator \( L \) is positive definite. The pointed brackets denote an inner product of the functions located on both sides of the comma. The simplest form of inner product is the integral of the product over the domain of the problem.

A discretization of such problems yields the matrix eigenvalue equation

\[
A_\psi = \lambda B_\psi \]  \hspace{1cm} (2.100)

The matrix \( B \) is the identity matrix when (2.100) is generated by the use of a finite difference operators. Methods of solution of (2.100), when \( B \) is a general or banded symmetric matrix, will be discussed in conjunction with the finite element method.

### 2.6.1 Direct Methods

The triangulation processes of \( A \) and back substitution were sufficient for solving (2.32) by direct methods. To solve (2.100) more powerful methods are required for determining all or specific eigenvalues of \( A \). In addition, methods for obtaining the corresponding eigenvector are needed. The various methods differ depending on the requirement for the complete eigensolution (all eigenvalues and eigenvectors) or the partial eigensolution (specific eigenvalues and/or eigenvectors). The methods are based on the reduction of \( A \) to a diagonal or tridiagonal form by similarity transformations.
(unitary for Hermitian matrices and orthogonal for real symmetric matrices) of the form

\[ R^H A R = \text{diag} \{ \lambda_i \} = A \]  

(2.101)

where \( R \) is a unitary matrix\(^1\) (or an orthogonal matrix\(^2\)). The resulting eigenvalues are real.

2.6.1.1 Jacobi Algorithm

This algorithm reduces the original symmetric matrix \( A \) to diagonal form by a sequence of plane rotations [Young-Gregory (1973) Sec. 14.2]. Theoretically, the algorithm requires an infinite number of plane rotations; in practice, the process is terminated when the off-diagonal elements are negligible to working accuracy.

Let the pivotal element \( a_{pq} \) for the plane rotation in the \((p,q)\) plane have magnitude greater than the average magnitude of the off-diagonal elements of \( A^{(n)} \), then

\[ \lim_{n \to \infty} A^{(n)} = A \]  

(2.102)

where the sequence of \( \{A^{(n)}\} \) is generated by

\[ A^{(n)} = R^{(n)} A^{(n-1)} R^{(n)T} \]  

(2.103)

The angle \( \theta \) of the plane rotation is chosen so as to reduce the pivotal element \( a_{pq}^{(n-1)} \) of \( A^{(n-1)} \) to zero. Thus the matrix \( R^{(n)} \) is

---

\(^1\) A matrix \( R \) is unitary if \( R R^H = I \)

\(^2\) A matrix \( R \) is orthogonal if \( R R^T = I \)
\[ R_{pp} = R_{qq} = \cos \theta, \quad R_{pq} = R_{qp} = -\sin \theta \]  

(2.104)

\[ R_{ii} = 1 \quad (i \neq p, q), \quad R_{ij} = 0 \quad \text{otherwise} \]  

(2.105)

for \( p < q \). \( A^{(n)} \) differs from \( A^{(n-1)} \) only in rows and columns \( p, q \).

Explicit formulae for the transformation are given by

\[
\begin{align*}
\alpha_{ip}^{(n)} &= \alpha_{ip}^{(n-1)} \cos \theta + \alpha_{iq}^{(n-1)} \sin \theta = \alpha_{pi}^{(n)} \\
\alpha_{iq}^{(n)} &= -\alpha_{ip}^{(n-1)} \sin \theta + \alpha_{iq}^{(n-1)} \cos \theta = \alpha_{qi}^{(n)}
\end{align*}
\]

(2.106)

\[
\begin{align*}
\alpha_{pp}^{(n)} &= \alpha_{pp}^{(n-1)} \cos^2 \theta + 2\alpha_{pq}^{(n-1)} \cos \theta \sin \theta + \alpha_{qq}^{(n-1)} \sin^2 \theta, \\
\alpha_{qq}^{(n)} &= \alpha_{pp}^{(n-1)} \sin^2 \theta - 2\alpha_{pq}^{(n-1)} \cos \theta \sin \theta + \alpha_{qq}^{(n-1)} \cos^2 \theta,
\end{align*}
\]

(2.107)

and

\[
\alpha_{pq}^{(n)} = (\alpha_{pp}^{(n-1)} - \alpha_{pq}^{(n-1)}) \cos \theta \sin \theta + \alpha_{pq}^{(n-1)} (\cos^2 \theta - \sin^2 \theta) = \alpha_{qp}^{(n-1)}
\]

(2.108)

Since \( \alpha_{pq}^{(n)} = \alpha_{qp}^{(n)} \) are to be annihilated, Eq. (2.108) yields

\[
\tan 2\theta = \frac{2\alpha_{pq}^{(n-1)}}{\alpha_{pp}^{(n-1)} - \alpha_{pq}^{(n-1)}}, \quad p < q.
\]

(2.109)

In order to avoid unnecessary computation of \( \theta \) from (2.108) the following formulae should be used

\[
\cos \theta = \left[ \frac{1}{2} \left( 1 + \frac{\beta}{(\alpha^2 + \beta^2)^{1/2}} \right) \right]^{1/2},
\]

(2.110a)
\[ \sin \theta = \frac{\alpha}{2 \cos \theta \left( \alpha^2 + \beta^2 \right)^{1/2}} \]  
(2.110b)

where
\[ \alpha = 2a_{pq}^{(n-1)} \operatorname{sgn}(a_{pp}^{(n-1)} - a_{qq}^{(n-1)}) \]  
(2.111a)

and
\[ \beta = |a_{pp}^{(n-1)} - a_{qq}^{(n-1)}| \]  
(2.111b)

In summary,

a) Compute \( \alpha \) and \( \beta \) from (2.111);

b) Compute \( \cos \theta \) and \( \sin \theta \) from (2.110)

c) Compute \( A^{(n)} \) from \( A^{(n-1)} \), using (2.106) and (2.107) and setting
\[ a_{pq} = a_{qp} = 0 \] (instead of using (2.108))

This process is essentially iterative because an annihilated pivot at one rotation is usually made non-zero by subsequent rotations. A better version of this algorithm (the threshold Jacobi algorithm) [Wilkinson (1965) p. 277] is given in Appendix 10.

2.6.1.2 Givens' Algorithm

This algorithm [Wilkinson (1965) p. 282] removes the iterative nature of the Jacobi algorithm by preserving the off-tridiagonal zeros resulting from the annihilation of the pivots. The resulting matrix is obtained in a finite number of steps and is tridiagonal.

There are approximately \( 4n^3/3 \) multiplications in the complete reduction to the tridiagonal form as compared with \( 2n^3 \) multiplications in one sweep of the Jacobi process. The computation of the eigenvalues of a symmetric tridiagonal matrix is not difficult and hence this method is
very effective.

2.6.1.3 Householder's Algorithm

In this method [Householder (1958); Wilkinson (1965) p. 290] replaces the plane rotations of Givens' algorithm by the orthogonal similarity transformations (only one transformation per row and column of A). These transformations are more complicated than Givens' plane rotations but the complete process requires approximately half the computation of Givens' method. The resulting matrix is again tridiagonal. The method is endorsed by Wilkinson (1960, 1962), Ortega (1967), and Martin et al. (1968a, 1971).

The tridiagonalization in the Givens or Householder method prepares a given matrix for the final eigensolution, i.e., obtaining eigenvalues and eigenvectors.

2.6.1.4 Sturm Sequence and Bisection

The general concept of a Sturm sequence shall be introduced first, and then it will be used to locate the eigenvalues of a symmetric tridiagonal matrix.

Let the sequence of real polynomials $P_0(x), P_1(x), \ldots, P_m(x)$ have the following properties with respect to an open interval $(a,b)$:

(i) if $x_0 \in (a,b)$ and $P_k(x_0) = 0$, then $P_{k-1}(x_0) P_{k+1}(x_0) < 0$;
(ii) $P_0(x) \neq 0$ for any $x \in (a,b)$,

This sequence of polynomials is known as a Sturm sequence in the interval $(a,b)$. Fig. 2.19 shows an arbitrary Sturm sequence. Note that roots $\alpha_1$ and $\gamma_2$ must be in the interval $(\beta_1, \beta_2)$ regardless of
their relation, i.e., \( a_1 < \gamma_2 \) or \( \gamma_2 < a_1 \).

![Diagram](image)

**Fig. 2.19. Arbitrary Sturm sequence \( P_i(\lambda) \) for \( i = 0,1,2,3 \).**

The property (i) implies that the roots of \( P_i(\lambda) = 0 \) are distinct and are separated by the roots of \( P_{i-1}(\lambda) = 0 \). It is possible to determine the sign of each \( P_i(c) \) \( \forall c \in (a,b) \). If \( P_i(c) = 0 \), for some \( i \), then, in Wilkinson's convention [see Wilkinson (1965) p. 300],

\[
\text{sgn} [P_i(c)] = -\text{sgn} [P_{i-1}(c)] \quad (2.112)
\]

Note that there are other conventions [Young-Gregory (1973) p. 912]. Since no two consecutive \( P_i(c) \) can be simultaneously zero, the sign of \( P_i(c) \) is always defined.

Take \( c \) located as in Fig. 2.19. Then \( P_0(c) \sim + \), \( P_1(c) \sim - \), \( P_2(c) \sim - \), and \( P_3(c) \sim - \). It is apparent that the number \( s(c) \) of sign agreements in the sequence is equal to the number of roots of \( P_3(\lambda) = 0 \) which are greater than \( c \). When \( c = \gamma_2 \) in Fig. 2.19, then \( P_0(c) \sim + \), \( P_1(c) \sim - \), \( P_2(c) \sim - \), and \( P_3(c) \sim + \) (using 2.112) and hence \( s(c) = 1 \). Indeed, there is only one root of \( P_3(\lambda) = 0 \), \( \gamma_1 \), greater than \( \gamma_2 \). A general analysis of the sign-agreement property is given by Wilkinson (1965) p. 300.
Having specified this important property of a Sturm sequence, one can proceed to the Givens procedure for finding the eigenvalues of a symmetric tridiagonal matrix $T$. The assumption that all the off-diagonal elements of $T$ are non-zero is still general, because whenever such an element is zero, the matrix is block tridiagonal and hence each block can be treated separately. The characteristic polynomial of $T - \lambda I$ can be written in the form

$$
P_n(\lambda) = \det (T - \lambda I)
$$

$$
= \begin{vmatrix}
(d_1 - \lambda) b_1 \\
\quad b_1 (d_2 - \lambda) b_2 \\
\quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad b_{n-2} (a_{n-1} - \lambda) b_{n-1} \\
\quad \quad \quad \quad \quad \quad b_{n-1} (a_n - \lambda)
\end{vmatrix}
$$

(2.113)

Let $P_i(\lambda)$ be the determinant of the leading principal submatrix (of order $i$) of $T - \lambda I$, then

$$
P_1(\lambda) = d_1 - \lambda
$$

$$
P_2(\lambda) = (d_2 - \lambda) P_1(\lambda) - b_1^2
$$

$$
P_3(\lambda) = (d_3 - \lambda) P_2(\lambda) - b_2^2 P_1(\lambda),
$$

(2.114)

and a general recurrence formula is

$$
P_i(\lambda) = (d_i - \lambda) P_{i-1}(\lambda) - b_i^2 P_{i-2}(\lambda)
$$

(2.115)

where $P_{-1}(\lambda) \overset{\Delta}{=} 0$, $P_0(\lambda) \overset{\Delta}{=} 1$, and $b_o \overset{\Delta}{=} 0$. It can be shown (Givens)
that if \( b_i \neq 0 \) (i = 1, ..., n-1) then the roots of each \( P_i(\lambda) = 0 \) are
distinct and are separated by the roots of \( P_{i-1}(\lambda) = 0 \). Thus the se-
quence of polynomials in (2.115) constitutes a Sturm sequence. Therefore,
one can locate all or specific eigenvalues of \( T \) by the use of
the Sturm sequence count when evaluating the characteristic polynomials
of the leading principal submatrices \( T_i \).

Since \( \rho(\cdot) \leq ||\cdot||_B \forall B \), hence all of the eigenvalues of \( T \)
(roots of \( P_n(\lambda) = 0 \)) lie within \([-||T||_\infty, ||T||_\infty]\) where

\[
||T||_\infty = \max_i \{|b_i| + |d_i| + |b_{i+1}|\}.
\]  

(2.116)

Further assume that

\[
\lambda_1 > \lambda_2 > \ldots > \lambda_k > \ldots > \lambda_n.
\]  

(2.117)

Using bisection of the interval \([-||T||_\infty, ||T||_\infty]\) and (2.115) one can
find whether the sought \( \lambda_k \) lies to the left or to the right of the
midpoint. Then, one repeats the process say \( m \) times until

\[
s(c_L) > k, s(c_u) < k \text{ for } c_L < c_u
\]  

(2.118)

which means that \( (c_L, c_u)^m \) contains only the eigenvalue \( \lambda_k \). The
interval \( (c_L, c_u)^m \) is \([-||T||_\infty, ||T||_\infty]/2^m \) wide.

The process of bisection can be continued until the eigenvalue
is located in the predetermined interval \( (c_L, c_u)^{(p)} \) which is reached
after \( p + m \) iterations, \( (c_L, c_u)^{(p)} = (c_L, c_u)^{(m)}/2^p \). At \( r \)th step of
this final stage, the procedure is as follows

a) Compute the midpoint

\[
\frac{c_m^{(r)}}{2} = \frac{1}{2} \left( c_u^{(r-1)} - c_L^{(r-1)} \right);
\]  

(2.119)

b) Compute the sequence
and determine \( s(c_m(r)) \).

If \( s(c_m(r)) \geq k \), then \( c_k(r) = c_m(r) \), \( c_u(r) = c_{u-1}(r) \)

If \( s(c_m(r)) < k \), then \( c_k(r) = c_{k-1}(r) \), \( c_u(r) = c_{u-1}(r) \)

The relation (2.118) is preserved on the new interval \((c_k(r), c_u(r))\).

This algorithm has only linear convergence but is very stable.

If the eigenvalues are pathologically close the algorithm has to be modified [Wilkinson (1965) p. 306]

The algorithm has been applied to different problems [Barth et al. (1967); Peters-Wilkinson (1969); Gupta (1972, 1973)].

The Jacobi algorithms are well suited for obtaining all eigenvalues simultaneously and, in addition the complete orthogonal set of eigenvectors (if \( n \) is not large). The Givens and Householder algorithms give the eigenvalues one at a time but do not yield the corresponding eigenvectors.

2.6.1.5 QR Algorithm

For non-Hermitian (non-symmetric) matrices other methods have to be applied. The power method [Young-Gregory (1973) p. 915] yields the dominant eigenvalue \( \lambda_1 \) with its corresponding eigenvector. The inverse power method [Young-Gregory (1973) p. 919, Wilkinson (1965) p. 619] yields any eigenvalue and eigenvector. Another powerful method which yields all the eigenvalues of \( A \) is the QR algorithm [Wilkinson (1965) Ch. 8, Young-Gregory (1973) p. 921] which is defined by the iteration
\[ A_i = Q_i R_i \]
\[ A_{i+1} = Q_i^{-1} A_i Q_i \quad i = 1, 2, \ldots \]
\[ = R_i Q_i \quad (2.122) \]

where \( R_i \) is upper triangular and \( Q_i \) is a unitary matrix. If \( A_1 \) in (2.122) has eigenvalues satisfying \( |\lambda_1| > |\lambda_j| > 0 \) (\( j = 2, \ldots, n \)) then the sequence \( \{A_i\} \) produced by the QR algorithm converges essentially to an upper triangular matrix (which displays the eigenvalues on the main diagonal). Wilkinson (1965, p. 524) indicated that this algorithm is practical only when applied to a matrix in (upper) Hessenberg form (almost triangular form). Thus, a matrix \( A \) should be first condensed to upper Hessenberg form [Young-Gregory (1973) p. 924] and then the eigenvalue can be obtained by the use of the QR algorithm.

2.6.1.6 Inverse (Wielandt) Iteration

Once very accurate eigenvalues are determined by the above methods,

---

1 If \( AA^H = I \), then \( A \) is unitary

2 A matrix \( A \) has Hessenberg form if

\[
A = \begin{bmatrix}
  a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\
  a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\
  a_{32} & a_{33} & \cdots & a_{3n} \\
  \vdots & \vdots & \ddots & \ddots \\
  a_{n-1} & a_{nn} \\
\end{bmatrix}
\]
the corresponding eigenvectors can be computed. It is possible, at
least for the tridiagonal, to write explicit formulae for these eigenvectors [Wilkinson (1965) Sec. 5.48]. However, instabilities of the
explicit expressions may cause catastrophic errors in the eigenvectors. Another method, inverse iteration, appears to be the most
satisfactory of known methods [Wilkinson (1965) Sec. 5.60] for computing
the eigenvectors of T-matrices.

Let A from \( A \chi = \lambda \chi \) be reduced to T

\[
A = P T P^T
\]

\[
 PP^T = I
\]

(2.123)

and if \( \lambda \) is an eigenvalue of T with corresponding \( \chi \), then \( T \chi = \lambda \chi \)
and so

\[
\lambda (P \chi) = P T \chi
= P T P^T (P \chi)
= A (P \chi)
\]

(2.124)

Eq. (2.124) shows that \( \lambda \) is an eigenvalue of A with corresponding
eigenvector

\[
\chi = P \chi
\]

(2.125)

The justification of using the intermediate step (2.125) lies in the
fact that computing \( \chi \) for tridiagonal matrices is not difficult. If
\( \lambda_e \) is a good approximation to \( \lambda \), then the inverse iteration is character-
ized by the sequence
\[(T - \lambda_e I)\chi^{(1)} = \chi^{(o)} , \quad \chi^{(1)} = \frac{1}{n^{(1)}} \chi^{(1)}\]
\[(T - \lambda_e I)\chi^{(2)} = \chi^{(1)} , \quad \chi^{(2)} = \frac{1}{n^{(2)}} \chi^{(2)}\]
\[\vdots\]
\[(T - \lambda_e I)\chi^{(k)} = \chi^{(k-1)} , \quad \chi^{(k)} = \frac{1}{n^{(k)}} \chi^{(k)}\]

where
\[n^{(k)} = \max_i |y_i^{(k)}| = \|\chi^{(k)}\|_\infty ,\]  \hspace{1cm} (2.127)

and \(y^{(o)}\) is an arbitrary normalized (i.e., \(\|\chi^{(o)}\|_\infty = 1\)) vector. This process converges \((\chi^{(k)} \rightarrow \chi)\) with mild assumptions on \(\lambda_e\) and \(\chi^{(o)}\) [Wilkinson (1965) Sec.5.53-60]. It is noted that \(\lambda_e\) is constant in the process. Since each step of (2.126) involves the solution of a system of linear algebraic equations whose coefficient matrix is tridiagonal, either Gaussian elimination or triangular decomposition can be readily used to obtain \((T - \lambda_e I)^{-1}\). The inverse iteration is basically valid also for non-Hermitian matrices [Young-Gregory (1973) p. 942]

This technique is used not only for eigenproblems with \(B = I\) but for a general \(B\) [Martin-Wilkinson (1967); Peters-Wilkinson (1969); Dubrulle (1972); Gupta (1973); Bathe-Wilson (1973); Dong et al. (1973)]

2.6.2 Iterative Methods

Application of the direct methods in the solution of the general eigenvalue problem \(Ay = \lambda B y\) is economical when \(A\) and \(B\) are of moderate size and possibly symmetrical. (see next chapter). The use of the finite difference operators often leads, however, to matrices that are large and also nonsymmetric. For example, discretization of Helmholtz's equation \((v^2 + k_e^2)u = 0\) produces a matrix eigenvalue problem
\[(A - \lambda I)u = 0\]  \hspace{1cm} (2.128)

where \(\lambda = (\hbar k_c)^2\). When using the 5-point Laplacian (Appendix 2), the explicit form of (2.128) is given by

\[-u_1 - u_2 + (4 - \lambda)u_{o1} - u_3 - u_4 = 0 \hspace{0.5cm} \forall \hspace{0.2cm} i \in n\]  \hspace{1cm} (2.129)

suitably amended for boundary conditions. (Note that the signs in (2.129) are changed in order to use the concept of a positive definite matrix rather than a negative definite matrix). Even though the matrix is sparse (at most 5 nonzero elements), the order \(N\) is usually large for high accuracy.

Iterative methods employed in the solution of (2.128) are essentially based on the classical inverse power method which closely relates to the Wielandt inverse iteration (Sec. 2.6.1.6). The inverse power method is in turn related to the power method characterised as follows.

Let the matrix \(B\) have \(n\) eigenvalues \(\eta\) such that \(\vert \eta_j \vert < \vert \eta_1 \vert\), \((j = 2, 3, \ldots, n)\) so that \(\eta_1\) is the dominant. Let \(\{\chi_1, \chi_2, \ldots, \chi_n\}\) be the set of linearly independent normalized vectors. If this set is a basis then a linear combination of these vectors defines a new vector \(\chi\)

\[\chi = c_1\chi_1 + c_2\chi_2 + \ldots + c_n\chi_n.\]  \hspace{1cm} (2.130)

One can define a sequence \(\{\chi^{(1)}\}\)

\[\chi^{(1)} = B\chi^{(0)}\]

\[= c_1\eta_1\chi_1 + c_2\eta_2\chi_2 + \ldots + c_n\eta_n\chi_n\]
\[ \chi^{(2)} = B \chi^{(1)} \]
\[ = B^2 \chi^{(0)} \]
\[ = c_1 n_1^2 \chi_1 + c_2 n_2^2 \chi_2 + \ldots + c_n n_n^2 \chi_n \]
\[ \chi^{(k)} = c_1 n_1^k \chi_1 + c_2 n_2^k \chi_2 + \ldots + c_n n_n^k \chi_n \]
\[ \vdots \]
\[ \chi^{(k)} = \eta_1^k [c_1 \chi_1 + c_2 \frac{n_2}{n_1} \chi_2 + \ldots + c_n \frac{n_n}{n_1} \chi_n] \]
\[ = \eta_1^k [c_1 \chi_1 + \chi^{(k)}]. \] (2.132)

It is seen that \( \chi^{(k)} \rightarrow \chi \) as \( k \rightarrow \infty \), and its convergence depends on how rapidly the expressions \( c_j (n_j/n_1)^k \) approach zero. Thus the method converges to the dominant eigenvector with the corresponding eigenvalue
\[ \eta_1 = \frac{x_{i(k+1)}}{x_{i(k)}}, \quad \forall i \in n . \] (2.133)

If the vectors are normalized at each stage, then
\[ \chi^{(1)} = B \chi^{(0)}, \quad \chi^{(1)} = \frac{1}{m(1)} \chi^{(1)} \]
\[ \vdots \]
\[ \chi^{(k)} = B \chi^{(k-1)}, \quad \chi^{(k)} = \frac{1}{m(k)} \chi^{(k)} \]
\[ \vdots \] (2.134)

where the scaling factor, \( m^{(k)} \), is given by (2.127). Therefore, \( \chi^{(k)} \sim \chi_1 \) and \( m^{(k+1)} \sim \eta_1 \). The sequence \( \{m^{(k)}\} \) converges to the dominant eigenvalue and the sequence \( \{\chi^{(k)}\} \) to the eigenvector.

If one assumes that \( B \equiv (A-\lambda I)^{-1} \) then the sequence (2.134) is called
the inverse power method. The equation analogous to (2.132) becomes
\[
\chi^{(k)} = \frac{1}{(\mu_1-\lambda)^k} \left\{ c_1 \frac{\mu_1-\lambda}{\mu_1-\lambda} \chi_1 + \ldots + c_i \chi_i + \ldots + c_n \left[ \frac{\mu_1-\lambda}{\mu_n-\lambda} \chi_n \right] \right\}
\]
\[
= \frac{1}{(\mu_1-\lambda)^k} \left[ \mathbf{v}_i + \xi^{(k)} \right]
\]
where \( \mu_1 \) is the eigenvalue closest to \( \lambda \).

2.7 Summary

The detailed finite difference formulation presented in this chapter introduced the basic concepts associated with the discretization of a physical field problem by meshes. Various finite difference operators, including those applied in the discrete and continuous inhomogeneous media, have been discussed. Many direct and iterative methods for the solution of a matrix equation have been presented and analyzed. The standard and new scanning procedures used in iterative methods have been introduced.
CHAPTER 3

ACCELERATION OF ITERATIVE PROCESSES

3.1 Introduction

Extrapolation methods to accelerate convergence of the sequence of iterates are investigated. Transformation formulae derived from deterministic sequences are modified so that they may be used for their related stochastic sequences. The SER method, which is related to Aitken's $\delta^2$ process, is discussed. For linearly convergent sequences it is shown that SER not only will converge if the original sequence converges, but will converge to the same limit. The method is applicable to convergent and locally convergent vector sequences.

Recently the successive extrapolated relaxation (SER) method [Della Torre-Kinsner (1972a, 1972b, 1973b, 1973c)] has been introduced to accelerate the convergence of solutions to elliptic partial differential equations when solved by the method of finite differences. This method uses an essential modification of Aitken's formula [Aitken (1926, 1927, 1950)] (see Appendix 11) as its basis so that the convergence is assumed even when the original vector sequence converges linearly$^1$ rather than geometrically$^2$. We shall discuss acceleration formulae based upon other types of deterministic convergence, and how to modify them for related stochastic convergence.

---

1. A sequence $\{y^{(i)}\}$ converges linearly if $y^{(i)} = x^{(i)} + \sigma^{(i)}$ where $\{x^{(i)}\}$ converges geometrically and $\sigma^{(i)} \to 0$.

2. A sequence converges geometrically if, the ratio of any two successive errors is constant.
Special attention will be given to iterative finite difference solutions which lead to linearly convergent vector sequences to which the SER method applies.

3.2 Deterministic and Related Stochastic Sequences

Some numerical solutions to problems involve iterative processes. These processes generate sequences of answers which may converge to the solutions. It is noted that the better the initial approximation to the solution the faster the solution will be obtained. The sequence is terminated when some error criterion is met. This section concerns itself with how to extrapolate to the "correct" answer and this scalar or vector is used as a new starting approximation to the solution for the iterative process. Extrapolation can be applied regularly during the iterative process.

Suppose the sequence of answers were given by

\[ x^{(n)} = a \sum_{p=0}^{n} \frac{(1/2)^p}{p!} + b \]  \hspace{1cm} (3.1)

where a and b are unknown constants. It is known that the limit is given by

\[ x^{(\infty)} = a e^{1/2} + b \] \hspace{1cm} (3.2)

The factors a and b can be uniquely determined by examining the first two steps of the iterative process since

\[ x^{(0)} = a + b \] \hspace{1cm} (3.3)

\[ x^{(1)} = 1.5a + b \] \hspace{1cm} (3.4)
Therefore

\[ a = 2(x^{(1)} - x^{(0)}) \]  
\[ b = 3x^{(0)} - 2x^{(1)} \]  

and

\[ x^{(\infty)} = (3 - 2e^{1/2}) x^{(0)} - 2(1 - e^{1/2}) x^{(1)} \]  

(3.7)

A formula of this type relating the extrapolated value to the terms of a sequence will be referred to as the transformation formula of that sequence. Thus, the answer could be obtained from (3.7) after only two iterations.

In practice, a process that would ideally produce a sequence as in (3.1) is perturbed by noise, and the use of numerical methods. The perturbed process will be referred to as the related stochastic sequence to the deterministic sequence of (3.1). In particular, consider the sequence

\[ x^{(n)} = a \sum_{p=0}^{n} \frac{(1/2)^p}{p!} b + \sigma^{(n)} \]  

(3.8)

where \( \sigma^{(n)} \) is a random variable with the property that

\[ \lim_{n \to \infty} \sigma^{(n)} = 0. \]  

\[ \lim_{n \to \infty} \sigma^{(n)} = 0. \]  

(3.9)

A property of a related stochastic sequence such as (3.8) to a deterministic sequence, in this case (3.1), is that they have the same limit, (3.2).

We shall define the error function \( e^{(n)} \) by

\[ e^{(n)} = x^{(n)} - x^{(\infty)} \]  

(3.10)
It is noted that for a convergent sequence, such as (3.1)

\[ \lim_{n \to \infty} e^{(n)} = 0. \]  \hspace{1cm} (3.11)

We shall define the extrapolated error function \( f^{(n)} \) by

\[ f^{(n)} = y^{(n)} - x^{(\infty)} , \]  \hspace{1cm} (3.12)

where \( y^{(n)} \) is the value after using an extrapolation formula such as (3.7). It is noted that for the deterministic sequence \( f^{(1)} = 0 \), that is, there is no residual error after extrapolation from two iterates. However, for the stochastic sequence there will be a residual error.

If on the average

\[ \left| \frac{f^{(n)}}{e^{(n)}} \right| < 1 , \]  \hspace{1cm} (3.13)

then the sequence with extrapolation will converge faster than the original sequence.

For the related stochastic sequence given in (3.8) if one uses the transformation (3.7) for the related deterministic sequence then

\[ y^{(1)} = (3 - 2e^{1/2})(x^{(0)} - \sigma^{(0)}) - 2(1 - e^{1/2})(x^{(1)} - \sigma^{(1)}) . \]  \hspace{1cm} (3.14)

At this point

\[ e^{(1)} = x^{(1)} - x^{(\infty)} = (1.5 - e^{1/2})a + \sigma^{(1)} , \]  \hspace{1cm} (3.15)

and

\[ f^{(1)} = y^{(1)} - x^{(\infty)} = 2(1 - e^{1/2}) \sigma^{(1)} - (3 - 2e^{1/2}) \sigma^{(0)} . \]  \hspace{1cm} (3.16)
It is seen that if the \( \sigma \)'s are zero, the related stochastic sequence reduces to the deterministic sequence and the error after extrapolation is zero. If one examines the ratio in (3.13), then one finds that an improvement is obtained by extrapolating only if
\[
\sigma^{(0)} - .5a < \sigma^{(1)} < .129\sigma^{(0)} + .0647a \quad \text{for } a > 0
\] (3.17)
which is easily satisfied if the \( \sigma \)'s decrease to zero.

### 3.3 Geometric and Linear Convergent Sequences

Consider the geometrically convergent sequences given by
\[
x^{(n)} = a^n + b,
\] (3.18)
where \( |a| < 1 \). It is obvious that \( x^{(\infty)} = b \). The transformation equation for this sequence is referred to as Aitken's formula and is given by the following nonlinear transformation of first degree
\[
y^{(n)} = \frac{[x^{(n-1)}]^2 - x^{(n)} x^{(n-2)}}{2x^{(n-1)} - x^{(n)} - x^{(n-2)}}
\] (3.19)
(For other forms of Aitken's formula see Appendix 12).

The related stochastic sequence is linearly convergent and is defined by
\[
x^{(n)} = a^n + b + \sigma^{(n)}
\] (3.20)
when, in addition, the limitation is imposed that
\[
\lim_{n \to \infty} \sigma^{(n)} = 0
\] (3.21)

Since (3.19) is a nonlinear transformation direct application to (3.20) leads to three types of problems. First of all there is the possibility...
that the denominator of (3.19) may become very small and the extrapolation diverges. The second possibility is that the ratio of the first difference $\delta^{(n)}$ to the second difference $\delta^2(n)$ of the sequence is positive in which case (3.19) extrapolates in the wrong direction. Finally, if the successive first differences are of opposite sign, (3.19) does not yield the optimum extrapolation. The problems are illustrated in Fig. 3.1.

3.4 Physical Problems that Lead to Geometrically Convergent Sequences

When solving N-dimensional elliptic partial differential equations numerically by the method of finite differences one discretizes the problem by choosing an N-dimensional lattice and defines a vector $\chi$ whose dimension is equal to the number of lattice points. If the space is a hypercube with L lattice points on a side, then the dimension of $\chi$ is $L^N$. We shall now show that using the Jacobi method one generates a vector sequence $\{\chi^{(n)}\}$ which is linearly convergent if the coefficient matrix of the linear system corresponding to the elliptic partial differential equation is positive definite and the spectral radius of the iteration matrix is less than unity. An example of such a system is the solution of Laplace's equation using a five-point operator. A typical plot of a sequence of the potential for given boundary conditions is shown in Fig. 3.2.

The solution at a given lattice point is the superposition of the Green's function solutions for all the boundary points. A typical Green's function using the Jacobi method will be zero for the first
Fig. 3.1. Aitken's extrapolation of (a) convergent, (b) divergent, and (c) oscillating sequence.
Fig. 3.2. Typical potential sequence for given boundary conditions.
steps of the iterative process because it takes a finite number of iterations for the effect of the boundary condition to reach that lattice point. If there were no other boundary points, then Green's function would converge geometrically to the correct Green's function. The effect of the other boundary points is to reflect additional terms to the Green's function solution. Thus, the sequence consists of a super-position of an infinite number of waves of the type shown in Fig. 3.3 where for each subsequent wave the amplitude $a_k$ gets smaller and smaller and the characteristic number of iterations $n_{1k}$ and $n_{2k}$ get larger and larger.

In the limit, as $n \to \infty$ one of these waves will dominate so that the sequence will have an error of the form $a_k e^{-(n-n_{1k})/n_{2k}}$. We can write the error for small $n$ as $a_k e^{-n_{1k}/n_{2k}} + o(n)$ where $o(n)$ contains all the other waves. From this analysis it is seen that $\lim_{n \to \infty} o(n) = 0$. Therefore, the Green's function converges linearly. Since the superposition of linearly convergent sequences converges linearly, then $(\chi(n))$ converges linearly.

Since the only difference between the Jacobi and the Gauss-Seidel methods is a faster propagation of the effects of boundary conditions throughout the array, all the previous arguments hold except the corresponding values of $n_{1k}$ and $n_{2k}$ will be smaller. Similarly, if one uses finite-difference operators such that the coefficient matrix $A$ has diagonal dominance, then faster linear convergence will result.
In addition, if the operator has the Young's Property $A$, the corresponding iterative method can utilize the optimization of the accelerating
Fig. 3.3. Potential sequence due to a single boundary point.
factor such as used by Carré (1961) and still lead to a linearly convergent vector sequence which, in fact, converges faster.

3.5 Successive Extrapolated Relaxation (SER) Method

The SER method was developed [Della Torre-Kinsner (1973b)] to accelerate the convergence of iterative methods used to solve elliptic partial differential equations. We shall later show that these methods generate sequences of vectors which converge linearly to the vector solution. The problem of divergence introduced by (3.19) is removed by limiting the extrapolation. Backward extrapolation is detected and inverted. Oscillating sequences are not extrapolated. The resulting extrapolation formula is given by

\[
\gamma_i^{(n)} = \begin{cases} 
  x_i^{(n)} + 2(1 + \gamma)\delta_i^{(n)} & , \delta_i^{(n-1)} \delta_i^{(n)} > 0, |\delta_i^{(n)}| < |\delta_i^{(n-1)}| \\
  x_i^{(n)} + (3 + 2\gamma)\delta_i^{(n-1)} & , \delta_i^{(n-1)} \delta_i^{(n)} > 0, |\delta_i^{(n)}| > |\delta_i^{(n-1)}| \\
  x_i^{(n)} & , \delta_i^{(n-1)} \delta_i^{(n)} < 0
\end{cases}
\]

(3.22)

where

\[
\gamma = \kappa - 1
\]

(3.23)

and \(\kappa\) is a constant, and \(\delta_i^{(n)} = x_i^{(n)} - x_i^{(n-1)}\).

If certain severe conditions are placed on the \(\sigma_i\)'s, then (3.19) can be used successfully. We shall show, however, that (3.22) will converge more generally than (3.19).
3.5.1 Derivation of Basic Formula

The derivation of the basic formula (3.22) is somewhat heuristic in its nature. Nevertheless, this approach can easily be understood. A formal treatment is given in the following sections.

Consider a two dimensional field problem. The potential is computed at each mesh node denoted by a double subscript, \( k \) and \( \ell \). The superscripts in parenthesis relate to the iteration count. Some analogies and differences between the successive extrapolated relaxation method and the successive overrelaxation (SOR) method will be discussed. It will be assumed that a convergent sequence of iterates is available.

The SOR method is based on the principle that if after \( n \) iterations in a two dimensional array an interior point \( (k, \ell) \) has achieved the value \( x^{(n)}_{k,\ell} \) and examination of the relevant neighbours indicates that on the \( n+1 \) iteration the value should be \( x^{(n)}_{k,\ell} \) then the next value to be taken for \( x^{(n+1)}_{k,\ell} \) shall be

\[
x^{(n+1)}_{k,\ell} = x^{(n)}_{k,\ell} + \omega(x^{(n)}_{k,\ell} - x^{(n)}_{k,\ell})
\]  

(3.24)

When \( \omega = 1 \), Eq. (3.24) represents the Gauss-Seidel method. For \( 1 < \omega < 2 \) more rapid convergence is achieved. But there exists only one \( \omega \) which produces the largest rate of convergence. Optimization of \( \omega \) (see 2.5.14) becomes very difficult when the eigenvalues of the system matrix are complex [Young (1971)].

The SOR method is an essential modification of Aitken's \( \delta^2 \) process which can best be understood by examining Fig. 3.4. It is seen that for the \( (n-2) \) iteration, the value is given by \( x^{(n-2)}_{k,\ell} \), and subsequent values increase (or decrease) monotonically towards the desired
final value \( x_k^{(m)} \). (Monotonic behaviour is characteristic of the Gauss-Seidel method with only positive boundary values.) The difference between SOR and SER is that SOR tries to improve convergence by increasing (or decreasing) the slope of the curve in Fig. 3.4, while SER tries to predict \( x_k^{(m)} \) directly.

If one assumes that the approach to the asymptotic value is characterized by an exponential behaviour of the form

\[
x_k^{(n)} = x_k^{(m)} + (x_k^{(0)} - x_k^{(m)}) \alpha^n,
\]

then, after a minimum of three iterations one can extrapolate to the asymptotic value \( A_k^{(n)} \) which will be taken as the value of \( x_k^{(n+1)} \). The extrapolated value is given by Aitken's formula (see Eq. 3.19)

\[
A_k^{(n)} = \frac{[x_k^{(n-1)}]^2 - x_k^{(n-2)} x_k^{(n)}}{2x_k^{(n-1)} - x_k^{(n-2)} - x_k^{(n)}}.
\]

The SER method is basically applied to a convergent process, such as SOR, and after every third iteration (3.26) is applied simultaneously to all \( x_k^{(n)} \). The modifications to (3.26) are to permit extrapolation even when the convergence is linear rather than geometric and to make the convergence properties less sensitive to computation errors.

It is convenient to define

\[
\delta_1 = x_k^{(n-1)} - x_k^{(n-2)},
\]

\[
\delta_2 = x_k^{(n)} - x_k^{(n-1)},
\]

and

\[
\delta = \delta_1 + \delta_2.
\]
It is noted that the $\delta$'s vary with $k$, $l$, $n$, however, this notation is suppressed for convenience. Implicit in (3.26) are the conditions that

$$\delta_1 \delta_2 > 0$$

(3.30)

and

$$|\delta_2| < |\delta_1|$$

(3.31)

Violation of (3.30) indicates that $x_{k,l}^{(n)}$ is not a monotonic function of $n$. This condition occasionally occurs in relaxing an array, especially during approach to convergence. It has been found that faster convergence results if $x_{k,l}^{(n+1)}$ is taken to be $x_{k,l}^{(n)}$ rather than using (3.26) in this case.

If $|\delta_1 - \delta_2|$ is very small, even if (3.31) holds, the value extrapolated from (3.26) will not only be very large but will be highly susceptible to errors. It was found necessary in this case to limit extrapolation to $\kappa \delta$ where $\kappa$ is a constant experimentally found to be approximately 1.5. At first, the following modification of (3.26) was used

$$E_{k,l}^{(n)} = \begin{cases} 
    x_{k,l}^{(n)} + \kappa \delta & \text{if} \quad |A_{k,l}^{(n)} - x_{k,l}^{(n)}| \geq |\kappa \delta| \\
    A_{k,l}^{(n)} & \text{if} \quad |A_{k,l}^{(n)} - x_{k,l}^{(n)}| < |\kappa \delta|.
\end{cases}$$

(3.32)

(3.33)

This condition will be referred to as the hard limiting condition, and is indicated by the solid line in Fig. 3.5.

Faster convergence results when soft limiting is used. That is, as $|A_{k,l}^{(n)} - x_{k,l}^{(n)}|$ increases, limiting is introduced gradually by the
Fig. 3.5. Hard and soft limiting of extrapolation.
formula

\[ E_{k, \ell}^{(n)} = x_{k, \ell}^{(n)} + \kappa (1 - \mu) \delta \] (3.34)

where

\[ \mu = \frac{\delta_1 - \delta_2}{\delta_1 + \delta_2} \] (3.35)

This is illustrated by the dashed line in Fig. 3.5. The condition that \(|\delta_1 - \delta_2|\) is small corresponds to \(\mu\) being small. It is seen that in this case (3.34) properly reduces to (3.32). On the other hand, as \(\mu\) approaches one, (3.34) smoothly approximates (3.33). It is seen from Fig. 3.5 that for \(\mu > 1/3\) (3.34) yields a larger value than (3.33). Although many other functions were investigated to obtain soft limiting, the function chosen, though not ideal, requires little computing time.

If (3.31) is violated, then (3.26) cannot be applied because one is extrapolating in the wrong direction as shown in Fig. 3.6. This condition occurs most often at the beginning of a relaxation procedure since it takes a finite number of iterations for the effect of distant boundary conditions to propagate to a given point. In this case an extrapolation procedure that works well is derived from reflecting the extrapolation procedure about the \((n-1)\)-th point, as indicated in Fig. 3.6. In this case \(E_{k, \ell}^{(n)}\) is given by

\[ E_{k, \ell}^{(n)} = x_{k, \ell}^{(n-1)} + x_{k, \ell}^{(n)} - A_{k, \ell}^{(n)} \] (3.36)

This equation has the same problem as (3.26) when \(|\Delta_1 - \Delta_2|\) is small. Consequently, similarly to the way that (3.34) replaces (3.32) and (3.33), the following equation was derived

\[ E_{k, \ell}^{(n)} = x_{k, \ell}^{(n-1)} + (\kappa \mu + \kappa + 1) \delta \] (3.37)
Fig. 3.6. Forced extrapolation.
These results are summarized by

\[
E_{k,l}^{(n)} = \begin{cases} 
  x_{k,l}^{(n)} + 2(1 + \gamma)\delta_2 ; & \delta_1 \delta_2 \geq 0, \ |\delta_2| \leq |\delta_1| \\
  x_{k,l}^{(n)} + (3 + 2\gamma)\delta_1 ; & \delta_1 \delta_2 \geq 0, \ |\delta_2| < |\delta_1| \\
  x_{k,l}^{(n)} ; & \delta_1 \delta_2 < 0 
\end{cases} \tag{3.38}
\]

where

\[
\gamma = \kappa - 1 . \tag{3.39}
\]

It was found experimentally that a good choice for \( \gamma \) is 1/2.

Thus the procedure for approaching a solution is to perform the Gauss-Seidel procedure twice, storing only the values at the (n-2) -nd iteration in an extra array, then applying (3.38) to all the values simultaneously at the n-th iteration. This procedure is repeated until a solution is found to satisfactory accuracy. Although SOR could replace the Gauss-Seidel method in the above procedure, it was found that over-relaxation was not necessary at this point, since (3.38) normally over-estimates the extrapolated value.

It is seen that Eq. (3.38) is the same as (3.22) after the following changes in notation: \( E_{k,l} \equiv y_i, \ x_{k,l} \equiv x_i, \ \delta_1 \equiv \delta_i^{(n-1)}, \) and \( \delta_2 \equiv \delta_i^{(n)} \).

A more rigorous derivation of (3.38) will be indicated in Sec. 3.6.

### 3.5.2 Linear Convergence of Linear Iterative Methods

A linearly convergent vector sequence \( \{x_i^{(n)}\} \) can be expressed by

\[
x_i^{(n)} = D^{(n)} x_i^{(n-1)} + \xi_i , \tag{3.40}
\]
where $D^{(n)}$, a diagonal iteration matrix, may be written as

$$D^{(n)} = D_\alpha + D^{(n)}_\sigma = \begin{bmatrix} a_{11} + \sigma_{11}^{(n)} & \cdots & \sigma_{1n}^{(n)} \\ \vdots & \ddots & \vdots \\ \sigma_{n1}^{(n)} & \cdots & a_{nn} + \sigma_{nn}^{(n)} \end{bmatrix}$$

(3.41)

and

$$\chi^{(n)} = [x_1^{(n)} \ x_2^{(n)} \ \cdots \ x_N^{(n)}]^T$$

(3.42)

$$\xi = [c_1 \ c_2 \ \cdots \ c_N]^T$$

(3.43)

In order for the sequence to converge it is necessary that

$$|a_{ii}| < 1 \ \forall \ i \in N$$

(3.44)

and

$$\lim_{n \to \infty} \sigma_{ii}^{(n)} = 0 \ \forall \ i \in N$$

(3.45)

The matrix $D_\sigma$ is perturbed by the matrix $D^{(n)}_\sigma$, and hence the iteration matrix $b^{(n)}$ is not stationary. If $D^{(n)}_\sigma = 0$, the vector sequence $\{\chi^{(n)}\}$ converges geometrically.

When solving a linear system

$$A\chi = \zeta$$

(3.46)

we can construct a linear stationary iterative method of first degree,

$$F(\chi^{(n)}; A, \zeta)$$

of the form

$$\chi^{(n)} = M\chi^{(n-1)} + \xi$$

(3.47)

where the iteration matrix $M$ is not necessarily diagonal and $\xi$ is some vector. The commonly used methods such as the point Jacobi, the Gauss-
Seidel, the stationary Richardson, are completely consistent (see Sec. 2.5) with (3.46). The general convergence theorem states that the iterative method (3.47) is convergent for all initial vectors, $\xi^{(0)}$, if and only if the spectral radius of the iteration matrix $M$ satisfies the inequality

$$\rho(M) < 1.$$  \hfill (3.48)

Let $P$ be a matrix with certain properties, $D_\alpha$ a diagonal matrix, and $M_D$ an upper triangular matrix whose particular forms are the diagonal matrix $D_\alpha$ or the bidiagonal matrix.

**Theorem 1** The sequence of vectors $\{\xi^{(n)}\}$ generated by

$$\xi^{(n)} = M\xi^{(n-1)} + \xi$$

converges geometrically if and only if

$$PM^{-1} = D_\alpha.$$ 

A proof of this theorem is simple and is omitted here. It is noted that a diagonal matrix $M_D = D_\alpha$ can be obtained only if $M$ is a normal (or real and symmetric) matrix. In this case $P$ is a unitary (or orthogonal) matrix. If $M$ is Hermitian, then $M_D$ is real. For a general $M$, there might not exist a diagonalizing transformation. It is always possible, however, to find a unitary similarity transformation which transforms $M$ into an upper triangular form, or a more general transformation can be used to transform $M$ into a special type of bidiagonal form. The transformations preserve the eigenvalues of $M$. This is essential because the eigenvalues determine the convergence of $\{\xi^{(n)}\}$.

Let us examine the convergence of the vector sequence $\{\xi^{(n)}\}$ generated by (3.26) for the nonstationary process expressed by

$$\xi^{(n)} = M_D\xi^{(n-1)} + \xi, \quad n \geq 1.$$  \hfill (3.49)
For an iterative method, consistent with (3.46), the following holds:
if for some $n$ (not necessarily $n = 0$) $\chi^{(n)}$ is a solution $\chi^*$ of (3.46),
then $\chi^{(n+1)} = \chi^{(n+2)} = \cdots = \chi^*$. Thus

$$
\chi^{(n)} = M_D^{(n)} \chi^* + \epsilon,
$$
(3.50)

and we then get

$$
\chi^{(n)} = M_D^{(n)} \chi^{(n-1)}
= \chi^{(0)} P_{n1} M_D^{(p)}
$$
(3.51)

where

$$
\chi^{(n)} = \chi^{(n)} - \chi^*
$$
(3.52)

is the error vector. Let us define $\chi^{(n)}$ to be the error vector after transformation. Then

$$
\chi^{(n)} = \chi^{(n)} - \chi^*
$$
(3.53)

Theorem 2. Let $(x_1^{(n)})$ be the sequence of the $i$-th elements of the vector sequence $(\chi^{(n)})$ generated by

$$
\chi^{(n)} = M_D^{(n)} \chi^{(n-1)} + \epsilon, \quad n\geq 1
$$

and let $(y_1^{(n)})$ be the sequence defined by

$$
y_1^{(n)} = \frac{[x_1^{(n-1)}]^2 - x_1^{(n)} x_1^{(n-2)}}{2x_1^{(n-1)} - x_1^{(n)} - x_1^{(n-2)}},
$$

If the sequence $(x_1^{(n)})$ converges to a limit $x_1^*$, then not only does the sequence $(y_1^{(n)})$ converge but it converges to the same limit.
Proof

From (3.52), (3.53), and the definition of \( y_i^{(n)} \) it is seen that

\[
    f_i^{(n)} = e_i^{(n-2)} + \frac{[e_i^{(n-1)} - e_i^{(n-2)}]^2}{2e_i^{(n-1)} - e_i^{(n)} - e_i^{(n-2)}}.
\]

(3.54)

Application of (3.41) and (3.52) yields

\[
    \frac{f_i^{(n)}}{e_i^{(n-2)}} = \frac{(a_i + q_i^{(n-2)})(\sigma_i^{(n-1)} - \sigma_i^{(n-2)})}{(a_i - 1)^2 + \tau_i^{(n-1)}}
\]

(3.55)

or

\[
    \frac{f_i^{(n)}}{e_i^{(n)}} = \frac{\sigma_i^{(n-1)} - \sigma_i^{(n-2)}}{(a_i + q_i^{(n-1)})(\sigma_i^{(n-1)} - \sigma_i^{(n-2)})}
\]

(3.56)

where

\[
    \tau_i^{(n-1)} = (a_i + q_i^{(n-2)})q_i^{(n-1)} + (a_i - 2)\sigma_i^{(n-2)}.
\]

(3.57)

Since \( \lim_{n \to \infty} \sigma_i^{(n)} = 0 \) it follows that \( \lim_{n \to \infty} \tau_i^{(n)} = 0 \). In addition, \( |a_i| < 1 \). Therefore

\[
    \frac{f_i^{(n)}}{e_i^{(n)}} \to 0
\]

(3.58)

The following theorem refers to the particular extrapolation formula used in SER.

**Theorem 3** Let \( \{x_i^{(n)}\} \) be the sequence of the \( i \)-th elements of the vector sequence \( \{\xi^{(n)}\} \) generated by

\[
    \xi^{(n)} = M_D^{(n)} \xi^{(n-1)} + \xi,
\]

and let \( \{y_i^{(n)}\} \) be the sequence defined by

\[
    y_i^{(n)} = x_i^{(n)} + 2\kappa(x_i^{(n)} - x_i^{(n-1)})
\]
where $\kappa$ is a constant.

Then the sequence \( \{ y_i^{(n)} \} \) converges and it converges to the same limit as \( \{ x_i^{(n)} \} \) if and only if

$$
\kappa = \frac{a_i}{2(1 - \frac{a_i}{2})},
$$

where

$$
a_i = \lim_{n \to \infty} (a_{ii} + \sigma_{ii}^{(n)})
$$

**Proof** From (3.52), (3.53) and the definition of \( y_i^{(n)} \) it is seen that

$$
\frac{f_i^{(n)}}{c_i^{(n)}} = \frac{2\sigma_{i}^{(n-1)} + a_i(1 + 2\kappa) - 2\kappa}{a_i + \sigma_{i}^{(n-1)}}
$$

(3.59)

Since \( \lim_{n \to \infty} \sigma_{i}^{(n)} = 0 \) it follows that

$$
\lim_{n \to \infty} \frac{f_i^{(n)}}{c_i^{(n)}} = \frac{a_i(1 + 2\kappa) - 2\kappa}{a_i}
$$

(3.60)

In order for this expression to be zero, it is necessary that

$$
\alpha_i = \frac{2\kappa}{1 + 2\kappa}
$$

(3.61)

or

$$
\kappa = \frac{a_i}{2(1 - \frac{a_i}{2})}
$$

(3.62)

**Theorem 4** Let \( \{ x_i^{(n)} \} \) be the sequences generated by

$$
\dot{x}_i^{(n)} = M_i^{(n)} \chi_i^{(n-1)} + \xi
$$

convergent to the limits \( x_i^* \) for all \( i \) within a region \( R \),
and let \( \{y_{i}^{(n)}\} \) be the sequences defined by

\[
y_{i}^{(n)} = \frac{[x_{i}^{(n-1)}]^{2} - x_{i}^{(n)} x_{i}^{(n-2)}}{2x_{i}^{(n-1)} - x_{i}^{(n)} - x_{i}^{(n-2)}}
\]

or

\[
y_{i}^{(n)} = x_{i}^{(n)} + 2\kappa(x_{i}^{(n)} - x_{i}^{(n-1)})
\]

then the vector sequence \( \{\xi^{(n)}\} \) converges to the solution \( \kappa^{*} \).

**Proof**

Let \( \xi^{(n)}_{i} = (\lambda_{i} \cdot \xi^{(n)}) \lambda_{i} \).

where \( \lambda_{i} \) is the i-th unit vector.

Hence

\[
\xi^{(n)} = \xi_{1}^{(n)} + \xi_{2}^{(n)} + \ldots + \xi_{N}^{(n)}.
\]

(3.63)

It is seen that the vector sequence

\[
\{\xi^{(n)}\} = \{(\xi_{1}^{(0)} + \xi_{2}^{(0)} + \ldots + \xi_{N}^{(0)}) , (\xi_{1}^{(1)} + \xi_{2}^{(1)} + \ldots + \xi_{N}^{(1)}) ,
\]

\[
\ldots , (\xi_{1}^{(n)} + \xi_{2}^{(n)} + \ldots + \xi_{N}^{(n)}) , \ldots \}
\]

(3.65)

tends to

\[
\xi_{1}^{*} + \xi_{2}^{*} + \ldots + \xi_{N}^{*} = \kappa^{*}
\]

(3.66)

Then from Theorems 2 and 3 it follows that

\[
\{\xi^{(n)}\} \rightarrow \kappa^{*}
\]

(3.67)

3.5.3 **Convergence Analysis of Aitken's \( \delta^{2} \)-Process and SER**

An analysis of the bounds for the rate of convergence and perturbation from simple behaviour are made for Aitken's formula, SER
and SEOR. It is seen that the SER method converges at least as fast as the original sequence. For slowly convergent sequences, such as those in finite difference solution of differential equations, the improvement is very large and it is less susceptible to errors.

When applying Aitken's formula (3.19) to linearly convergent sequences (3.40) the ratio, $R$, of the error after to the error before application of (3.19) is given by (3.56). Fig. 3.7 is a plot of lines of constant $R$ for various values of $\sigma^{(n-1)}_i$ and $\sigma^{(n-2)}_i$ when $\alpha_i = .5$. It is noted that improvement in convergence occurs only in the region $|R| < 1$, that is, below the line labelled $R = -1$. If for example it is desired to have the extrapolated sequence converge twice as fast as the original sequence then one is limited to the region for which $|R| < 0.5$. In this case one is limited to the narrow strip between $R = 0.5$ and $R = -0.5$. The general nature of this diagram for other values of $\alpha$ is illustrated by Fig. 3.8.

Another way of looking at this analysis is to plot contours of constant $R$ as $\alpha_i$ and $\sigma^{(n-1)}_i$ are varied for the case where $\sigma^{(n-2)}_i = 0$. Again it is seen that improvement occurs only above the line $R = -1$. In fact if $\sigma^{(n-1)}_i < -.1$ convergence will be slowed down for all $\alpha$. The important thing to notice is the improvement obtained for slowly converging sequences, that $\alpha > .95$. In this case if $\sigma^{(n-1)}_i$ is -.0025 then $R = \infty$. Thus for sequences where extrapolation is most useful, that is for slowly convergent sequences, Aitken's formula cannot be applied!

If the same kind of plot as Fig. 3.9 is made for the extrapolation formula (3.22) with an error ratio given by (3.59), then one obtains
Fig. 3.7. Convergence improvement using Aitken's transform to linearly convergent sequences when $\alpha = 0.5$. 
Fig. 3.8. Convergence improvement using Aitken's transform for general $a$. 
Fig. 3.9. Convergence improvement using Aitken's transform for $\sigma_1^{(n-2)} = 0$. 
Fig. 3.10 for the case where $\gamma = 1/6$ ($\kappa = 7/6$). This choice of $\kappa$ optimizes the extrapolation process for sequence converging with $\alpha = .7$. Improvement will be obtained in convergence for 
-.23 < $\sigma_{i}^{(n-1)}$ < .7. However, if $\alpha$ is really .95 improvement will still be obtained if 
-.6 < $\sigma_{i}^{(n-1)}$ < .12. It is seen that the area where improvement occurs, especially for slowly convergent sequences, is much larger for SER than it is for Aitken's formula.

In successive extrapolated optimized relaxation (SEOR) the value of $\kappa$ is optimized as the process continues. For this method improvement contours are shown in Fig. 3.11. For a process in which $\alpha = .7$ one again obtains an allowable range for $\sigma_{i}^{(n-1)}$ of [-.23, .7]. However, if $\alpha = .95$ the allowable range for $\sigma_{i}^{(n-1)}$ is now [-.33, .95]. The dashed line on Fig. 3.11 indicates the allowable range for Aitken's formula if one wishes the extrapolated sequence to converge twice as fast as the original sequence. With this criteria SER will be better than Aitken's formula for $\alpha > .57$ whereas SEOR will be better than Aitken's formula for $\alpha > .31$.

Unlike the poles in Aitken's formulas as seen in Figs. 3.7, 3.8, and 3.9 the lines $R = \infty$ in Figs 3.10 and 3.11 do not indicate instability in the method. For this combination of $\alpha_{i}$ and $\sigma_{i}^{(n)}$, $\sigma_{i}^{(n)}$ is identically zero and $f_{i}^{(n)}$ is still finite. In fact, $f_{i}^{(n)}/e_{i}^{(n-1)}$ is finite over the entire $\alpha - \sigma$ plane indicating that both SER and SEOR are absolutely stable.

Another advantage of SEOR over SER is that the line $R = 0$ is $\sigma_{i}^{(n-1)} = 0$. This means that if at any step in the process the matrix
Fig. 3.10. Convergence improvement using SER ($\kappa = 7/6$) for linearly convergent sequences.
Fig. 3.11. Convergence improvement using optimized SER on linearly convergent sequences.
\( d^{(n)} \) is zero and all the diagonal elements of the matrix \( D_n \) are identical then the process extrapolates to the limit in one step as in the case of Aitken's formula.

3.5.4 **Upper and Lower Bounds for Generalized Perturbation Matrix**

The above two sections dealt with convergence properties of the SER method. The analysis was first restricted to one element of a vector and then generalized to the entire vector. That approach led to analytic expressions for errors which may be encountered during the computational process. In this section, a general error analysis is employed to give the lower and upper bounds for both the generalized perturbation matrix (formerly \( D^{(n)}_0 \) in (3.41)) and the extrapolation factor.

To solve a system

\[
A \xi = \eta
\]

one constructs an iterative method

\[
\xi^{(n)} = M \xi^{(n-1)} + \xi,
\]

where \( \xi \) is given by (3.43), and the iteration matrix \( M \) is not necessarily diagonal. The sequence \( \{\xi\} \) generated by (3.69) converges to the solution if the condition (3.48) is satisfied.

A general extrapolation process

\[
\xi^{(n)} = \xi^{(n)} + d^{(n)}
\]

applied to \( \{\xi^{(n)}\} \) can accelerate convergence of the sequence. Here \( d^{(n)} \) is the overstep vector produced by some particular extrapolation algorithm.

Both processes (3.69) and (3.70) are susceptible to computational errors. The most critical errors occur, however, in the background
process (3.69) because these vectors are extrapolated and the extrapolation takes place less frequently than the background process. Thus, once the possible numerical instabilities are removed from the extrapolation process, one can idealize the problem by assuming that the only errors may occur in the iterative process.

Similarly to Eq. (3.41), the contributing errors change Eq. (3.69) to the following form

\[ \chi^{(n)} = (M + M^{(n)}_{\sigma}) \chi^{(n-1)} + \xi \]  

(3.71)

where \( M^{(n)}_{\sigma} \) is the generalized perturbation matrix. Eq. (3.70) can be replaced by a specific (3.22) expression (but still with general coefficients)

\[ \chi^{(n)} = \chi^{(n)} + s(\chi^{(n)} - \chi^{(n-1)}) \]  

(3.72)

where \( s = s^{(n)} \) is a single-valued variable. Eqs (3.71) and (3.72) may be expressed in terms of the error vectors

\[ \xi^{(n)} = (M + M^{(n)}_{\sigma}) \xi^{(n-1)} \]  

(3.73)

and

\[ \xi^{(n)} = \xi^{(n)} + s(\xi^{(n)} - \xi^{(n-1)}) \]  

(3.74)

Substituting (3.73) into (3.74) yields

\[ \xi^{(n)} = (I + s[I - (M + M^{(n)}_{\sigma})^{-1}])\xi^{(n)} \]

\[ = (I + s[I - M^{-1}(I + P^{(n)})^{-1}])\xi^{(n)} \]

\[ = G^{(n)} \xi^{(n)} \]  

(3.75)

where

\[ P^{(n)} = M^{-1} M^{(n)}_{\sigma} \]  

(3.76)

and \( I \) is the identity matrix. Taking a norm of (3.75) leads to the
following inequality

$$|| \epsilon^{(n)} || \leq || G^{(n)} || + || \xi^{(n)} ||$$  \hspace{1cm} (3.77)

It is convenient to rewrite the error matrix as

$$G^{(n)} = (I + sI) - sM^{-1} (I + P^{(n)})^{-1}$$

$$= G_1^{(n)} - G_2^{(n)}$$  \hspace{1cm} (3.78)

Using the Minkowski and the Cauchy-Schwarz inequalities yields

$$|| G_1^{(n)} || \leq || I || + |s| || I ||$$

$$\leq 1 + |s|$$  \hspace{1cm} (3.79a)

$$|| G_2^{(n)} || = |s| \cdot || M^{-1} (I + P^{(n)})^{-1} ||$$

$$\leq |s| \cdot || M^{-1} || \cdot || (I + P^{(n)})^{-1} ||$$

$$\leq |s| \cdot || M^{-1} || \cdot \frac{1}{1 - ||P^{(n)}||}$$  \hspace{1cm} (3.79b)

The following lemma [Forsythe-Moler (1967) p. 110; Householder (1964) p. 54] has been used in (3.79b)

$$|| (I + P)^{-1} || \leq \frac{1}{1 - ||P||}$$  \hspace{1cm} (3.80)

if $||P|| < 1$ in some norm.

The error norm $|| \epsilon^{(n)} ||$ in (3.77) is not greater than $|| \xi^{(n)} ||$

if

$$0 \leq || G^{(n)} || \leq 1.$$  \hspace{1cm} (3.81)

Since

$$1 + |s| (1 - \frac{|| M^{-1} ||}{1 - ||P^{(n)}||}) > 0$$
\[
\left\| P^{(n)} \right\| \leq 1 - \frac{\left\| M^{-1} \right\|}{1 + |s|^{-1}} \tag{3.82a}
\]
and
\[
1 + |s| \left( 1 - \frac{\left\| M^{-1} \right\|}{1 - \left\| P^{(n)} \right\|} \right) \leq 1 \left\| P^{(n)} \right\| \geq 1 - \left\| M^{-1} \right\| \tag{3.82b}
\]
hence
\[
1 - \left\| M^{-1} \right\| \leq \left\| P^{(n)} \right\| \leq 1 - \frac{\left\| M^{-1} \right\|}{1 + |s|^{-1}} \tag{3.83}^1
\]

This expression determines the lower and upper bounds of the perturbation matrix because the eigenvalues of the inverse of a matrix are reciprocals of the eigenvalues of the matrix.

It is also important to determine the upper and lower bounds of the extrapolation factor \( s \) when \( \left\| P^{(n)} \right\| \) is known. Eq. (3.83) can be rewritten into the form
\[
1 - \frac{\nu(M)}{\left\| M \right\|} \leq \left\| P^{(n)} \right\| \leq 1 - \frac{\nu(M)}{\left\| M \right\| (1 + |s|^{-1})} \tag{3.84}
\]
where \( \nu(M) \) is the spectral condition number of a matrix \( M \) [Householder (1964) p. 81, 123; Young (1971) p. 85] defined by
\[
\nu(M) = \left\| M \right\| \cdot \left\| M^{-1} \right\| \tag{3.85}
\]
This condition number must satisfy the following relation
\[
\nu(M) \geq 1 \tag{3.86}
\]
with \( \nu(M) = 1 \) only when \( M \) is a unitary matrix, i.e., \( MM^H = I \).

Eq. (3.84) further indicates a well-defined closed interval \([a, b]\) for \( \left\| P^{(n)} \right\| \) with \( a < b \) because \( |s| \geq 0 \) and
\[
\frac{\nu(M)}{||M|| (1 + |s|^{-1})} \leq \frac{\nu(M)}{||M||}
\]  
(3.87)

It should be noted that the lower bound in (3.84) cannot be negative because if the spectral radius satisfies (3.48) then there exists some norm for which \( ||M|| < 1 \) [Householder (1964) p. 53]. At the same time, \( \nu(M) \geq 1 \). But this is in contrast with the requirement that \( || \cdot ||_n > 0 \). Thus the lower bound cannot be negative.

The right hand side inequality of (3.84) yields

\[
|s| \leq \frac{t}{\nu(M) - t}
\]  
(3.88)

where \( t = \frac{||M||}{(1 - || p^{(n)} ||)} \). Since \( ||M|| < 1 \) and \( || p^{(n)} || < 1 \) then \( t < 1 \). For \( \nu(M) = 4 \), \( || p^{(n)} || = 0.5 \), \( ||M|| = 0.8 \) we have \( t = 0.4 \) and \( |s| \leq 0.111 \). It is seen that even though the matrix \( M \) is well conditioned (\( \nu(M) = 4 \)) the extrapolation must be performed with caution for the perturbation matrix norm is large. If \( || p^{(n)} || \to 0 \), \( ||M|| \to 1 \), and \( \nu(M) \to 1 \), then \( |s| \gg 1 \) but is still finite since \( ||M|| < 1 \). Thus, if the permutation matrix norm has a large value (but \( || p^{(n)} || < 1 \)) or if the iteration matrix is ill-conditioned (\( \nu(M) \gg 1 \)) (\( ||M|| \ll 1 \), i.e., also \( \rho(M) \ll 1 \) [Householder (1964) p. 45]) then the extrapolation factor must be small.

3.6 Improvements of the SER Method

It was found that the value of \( \gamma \) which minimizes the number of iterations required to achieve 1% accuracy was suboptimal for achieving higher accuracy. In fact it is desirable to change \( \gamma \) as one proceeds
to a solution. This statement will be apparent from the following analysis.

One can gain considerable insight into the difficulties and power of Aitken's nonlinear transformation by examining the following forms of Aitken's formula (derived in Appendix 12)

\[
A_i^{(n)} = x_i^{(n-2)} + \frac{[\delta_i^{(n-1)}]^2}{\delta_i^{(n-1)} - \delta_i^{(n)}} \tag{3.89a}
\]

\[
= x_i^{(n-1)} + \frac{\delta_i^{(n-1)} \delta_i^{(n)}}{\delta_i^{(n-1)} - \delta_i^{(n)}} \tag{3.89b}
\]

\[
= x_i^{(n)} + \frac{[\delta_i^{(n)}]^2}{\delta_i^{(n-1)} - \delta_i^{(n)}} \tag{3.89c}
\]

\[
= x_i^{(n)} + \left[ \frac{1}{2} \left( 1 - \nu_i^{(n)} \right) \right]^2 [\nu_i^{(n)}]^{-1} (\delta_i^{(n)} + \delta_i^{(n-1)}) \tag{3.89d}
\]

\[
= x_i^{(n)} + [\gamma_i^{(n)}]^2 [1 - (\gamma_i^{(n)})^2]^{-1} (\delta_i^{(n)} + \delta_i^{(n)}) \tag{3.89e}
\]

\[
= x_i^{(n)} + \left[ \frac{1}{2} (1 - \nu_i^{(n)}) \right][\nu_i^{(n)}]^{-1} \delta_i^{(n)} \tag{3.89f}
\]

\[
= x_i^{(n)} + \gamma_i^{(n)} [1 - \gamma_i^{(n)}]^{-1} \delta_i^{(n)} \tag{3.89g}
\]

where:

\[
\delta_i^{(r)} = x_i^{(r)} - x_i^{(r-1)} \tag{3.90}
\]

\[
\nu_i^{(r)} = \frac{\delta_i^{(r-1)} - \delta_i^{(r)}}{\delta_i^{(r-1)} + \delta_i^{(r)}} \tag{3.91}
\]
and
\[
\gamma_i^{(r)} = \frac{\delta_i^{(r)}}{\delta_i^{(r-1)}}
\]  
(3.92)

Here \(\gamma_i^{(r)}\) is the ratio of successive differences and defines the rate of decay of the errors; similarly, \(\mu_i^{(r)}\) describes the behaviour of the errors. The first three forms of (3.89) are less susceptible to computational errors than the original Aitken's formula because they involve only one multiplication (instead of two in (3.26)). It is seen that erratic or infinite values of \(A_i^{(n)}\) will still occur when \(\delta_i^{(r)} + \delta_i^{(r-1)} \mu_i^{(r)} \approx 0\), or \(\gamma_i^{(r)} \approx 1\). The fact that \(\mu\) and \(\gamma\) are measures of errors implies, however, possible improvements of equations (3.89) which are identical to (3.26).

3.6.1 SER with Ultimate Frequency (Method I)

Let \(x_i^{(n)}\) be the \(i\)th component of a vector \(x_i^{(n)}\) at the \(n\)-th iteration. Then Eq. (3.89g) can be generalized to the form\(^1\)

\[
\xi_i^{(n)} = \chi_i^{(n)} = \xi_i - s(n)\xi_i^{(n-1)}
\]  
(3.93)

where

\[
\xi_i^{(n)} = \chi_i^{(n)} - \chi_i^{(n-1)},
\]  
(3.94)

\[
s(n) = \frac{\gamma(n)}{1 - \gamma(n)}.
\]  
(3.95)

and

\[
\gamma(n) = \frac{||\xi_i^{(n)}||_a}{||\xi_i^{(n-1)}||_a} \geq 0 \text{ since } ||\cdot||_a \geq 0.
\]  
(3.96)

Here ||cdot||_a is an \(\alpha\)-norm of the displacement vector \(\xi_i\). This method of extrapolation is very important because only two vectors, \(\chi_i^{(n)}\) and

\(^1\) Note that \(\gamma(n) \neq \gamma_i^{(n)}\) in general.
\( x^{(n-1)} \), are required by (3.93). The factor \( s^{(n)} \) may be substituted with \( s^{(n-1)} \) and computed in advance (at the (n-1)-st iteration in order to facilitate computation). If \( s^{(n)} \) is to be computed at the n-th iteration, the computation process will be longer (see Sec. 3.7). Although the extrapolation involves only two vectors, the extrapolated vector \( x^{(n)} \) is based on three vectors \( x_i \) because of \( s^{(n)} \). An extra storage is required for \( x^{(n-1)} \) because \( x_i \) is computed using the simultaneous (Jacobi type) technique regardless of the background method generating \( \{ x_i^{(n)} \} \). The ultimate (fastest) extrapolation frequency is one promotion every two iterations. This frequency may be regarded as an advantage of the method.

Any success of this method depends on whether \( \gamma^{(n)} \) defined by (3.96) can approach unity or not. This problem will be solved in the following section.

3.6.2 Error Spectrum

Assume that an iteration matrix \( M \) in (3.47) has real eigenvalues \( \xi_i \) and corresponding eigenvectors \( x_i \), for \( \forall i \in N \) where \( N \) is the dimension of \( M \). Then

\[
\xi_i^{(0)} = \sum_{\forall i} a_i x_i
\]  

(3.97)

and

\[
\xi_i^{(n)} = \sum_{\forall i} \xi_i^n a_i x_i = x_i^{(n)} - x_i
\]  

(3.98)

where the scalar coefficients \( a_i \) are uniquely determined by the choice of the iteration parameters. Since for a convergent method

\[
\max_i \xi_i \equiv \xi_1 < 1
\]  

(3.99)
it follows from (3.98) that for \( n \gg 1 \)

\[
\xi^{(n)} \approx \xi_1^n a_1 \chi_1
\]  

(3.100)

One can establish a similar relation for the displacement vector \( \dot{\xi}^{(n)} \).

Thus the ratio of two successive displacement vector norms is \( (n>1) \)

\[
\frac{||\dot{\xi}^{(n)}||}{||\dot{\xi}^{(n-1)}||} \approx \frac{||\xi_1^n a_1 \chi_1||}{||\xi_1^{n-1} a_1 \chi_1||}
\]

\[
\leq ||\xi_1||
\]  

(3.101)

It follows from (3.101), (3.99), and (3.96) that

\[
\gamma^{(n)} < 1
\]  

(3.102)

3.6.3 Weighted SER (Method II)

The difference between Method I and II lies in the determination

of the extrapolation factor \( s^{(n)} \) in (3.95) and, specifically, the

definition of \( \gamma^{(n)} \). Let \( \chi \) be a vector defining the relative weights.

Then

\[
s^{(n)} = \frac{\gamma^{(n)}}{1 - \gamma^{(n)}} = \frac{\langle \chi, \dot{\xi}^{(n)} \rangle}{\langle \chi, \dot{\xi}^{(2)}^{(n)} \rangle}
\]  

(3.103)

where \( \dot{\xi}^{(2)}^{(n)} \) is the second order displacement vector

\[
\dot{\xi}^{(2)}^{(n)} = 2\dot{\xi}^{(n-1)} - \dot{\xi}^{(n)} - \dot{\xi}^{(n-2)},
\]  

(3.104)

and \( \langle \cdot, \cdot \rangle \) is the inner product, e.g.,

\[
\langle \chi, \chi \rangle = \chi^T \chi
\]  

(3.105)
Eq. (3.103) implies many possible choices for \( \chi \). The proper choice of \( \chi \) should be determined by the type of convergence of the background iterative scheme and the necessity to avoid cancellation of the denominator of (3.103). The first weighted SER method results from the assumption that \( \chi = \hat{\chi} \). As it will be seen, this method requires that the iteration matrix is symmetric with positive eigenvalues less than 1 (i.e., the system matrix must be symmetric positive definite).

Using the orthogonality of the eigenvectors \( \chi_i \) of \( M \)

\[
\chi_m^T \chi_n = \delta_{mn},
\]

(3.106)

where \( \delta_{mn} \) is the Kronecker delta (0 for \( m \neq n \), and 1 for \( m = n \)), and substituting (3.98) into (3.103) yields

\[
s(n) = \frac{\sum \xi_i b_i}{\sum \left(1 - \xi_i\right) b_i}
\]

(3.107)

where

\[
b_i = \xi_i^2 \left(1 - \xi_i\right)^2 a_i^2 > 0
\]

(3.108)

It follows from (3.103) and (3.107) that

\[
\sum \left(\gamma(n) - \xi_i\right) b_i = 0
\]

(3.109)

Eq. (3.109) shows that \( \gamma(n) \) must lie between the largest and smallest eigenvalues of a symmetric \( M \).

When \( M \) is unsymmetric then one can use \( \chi = \hat{\chi}^2 \) and (3.107) becomes

\[
s(n) = \frac{\sum \xi_i \left(1 - \xi_i\right) b_i}{\sum \left(1 - \xi_i\right)^2 b_i}
\]

(3.110)
In this case cancellation cannot occur in the denominator because the denominator is \( \| \xi_0^2 \|^2_E \) where \( \| \cdot \|^2_E \) is the Euclidean norm.

The error in the extrapolated vector \( \chi^{(n)} \) for both methods can be obtained from (3.93) and (3.98)

\[
\xi^{(n)} = \chi^{(n)} - \bar{\xi} = \sum_{\mathcal{V}_i} d_i \xi_i^n a_i \bar{\chi}_i
\]

where

\[
d_i = (1+s)\xi_i^2 - s\xi_i = \frac{\xi_i(\xi_i - \gamma^n)}{1-\gamma(n)}
\]

Thus the error components are scaled by a factor \( d_i \). It is seen that (3.112) is a parabola with respect to \( \xi_i \). It depends on the extrapolation factor \( s \). When \( s = 0 \) then \( d_i \) min = 0 for \( \xi_i = 0 \). Otherwise the quadratic passes through \((\xi_i, d_i) = (0,0), (\gamma(n),0), \) and \((1,1)\) shown in Fig. 3.12. It is apparent from Fig. 3.12 that for a symmetric iteration matrix with positive eigenvalues at least some error components must be reduced when the extrapolation is applied. The greatest reduction should occur for the error component associated with the largest eigenvalue, for the convergence depends on it. It should be noted that \( d_i < -1 \) for \( \gamma(n) > 2\sqrt{2} - 2 = 0.828 \).

It is also seen that the extrapolation should not be applied in the ordinary way (every two or more iterations) when \( M \) has eigenvalues in the range -1 to 1 since \( d_i \) will magnify the error components for \( \xi < 0 \). In order to "square" the iteration matrix and obtain only positive eigenvalues one must apply one of the symmetric iteration schemes (see Sec. 2.5.1.5.d), that is, the extrapolation should be estimated from \( \chi^{(n)}, \chi^{(n+2)} \), and \( \chi^{(n+4)} \).
Fig. 3.12. Error spectrum for weighted SER.
The extrapolation can be carried again with the ultimate frequency (every two iterations). The convergence rate can be estimated by setting \( \gamma^{(n)} = 2/2 - 2 \) and \( \xi_1 = 1 - \alpha \) in Eq. (3.112). For small \( \alpha \)

\[
d_1 = \frac{\xi_1(\xi_1 - \gamma^{(n)})}{1 - \gamma^{(n)}} = 1 - 2(2+\sqrt{2})\alpha + O(\alpha^2)
\]  

(3.113)

This convergence rate is higher by a factor \( 2+\sqrt{2} \) than two iterations without extrapolation.

If a symmetric iteration matrix has some \( \xi_1 < -1 \) then the background iteration process diverges. But the application of the weighted SER with its ultimate frequency ensures convergence of the overall process!

3.6.4 Other Weighted SER Methods (Method III)

Now, after the above discussion, we are in the position to justify the use of a constant \( \kappa \) or an approximating function \( \kappa(1-u) \) in the SER method. Other approximating functions will be indicated.

Consider Eq. (3.89f) in its vectorial form

\[
\psi^{(n)} = \psi^{(n)} + \frac{1}{2} \sum_{i=1}^{N} \frac{1-\mu_{i}^{(n)}}{\mu_{i}} \delta_{i}^{(n)}, \quad i \in N.
\]  

(3.114)

We may also write

\[
\psi^{(n)} = \psi^{(n)} + \frac{1}{2} \frac{1 - \| \mu_{i}^{(n)} \|}{\| \mu^{(n)} \|} \delta^{(n)}.
\]  

(3.115)

where

\[
\mu = [\mu_{1}, \mu_{2}, \ldots, \mu_{N}]^{T},
\]  

(3.116)

and each \( \mu_{i} \) is given by (3.91). The simplest approximation of the extrapolation factor is a constant
\[
\begin{align*}
s &= \frac{1}{2} \frac{1 - \| b_i^{(n)} \|}{\| b_i^{(n)} \|} \quad \text{set} \quad \kappa \\
\frac{1 - \| b_i^{(n)} \|}{\| b_i^{(n)} \|} &= 2\kappa
\end{align*}
\]

(3.117)  
(3.118)

This value is being used in (3.22) and corresponds to soft limiting.

The hard limiting is obtained when the factor in (3.89d) is approximated by a constant above a certain value of the normalized value of the promotion

\[
\frac{1}{4} \frac{[1-\mu_i^{(n)}]^2}{\mu_i^{(n)}} \quad \text{set} \quad \kappa \quad \text{for} \quad \left| \frac{y_i^{(n)} - x_i^{(n)}}{\delta_i^{(n)} + \delta_i^{(n-1)}} \right| > \kappa
\]

(3.119)

The soft limiting is achieved when the factor in (3.89d) is approximated by a function

\[
\frac{1}{4} \frac{[1-\mu_i^{(n)}]^2}{\mu_i^{(n)}} = \kappa(1 - \mu_i^{(n)})
\]

(3.120)

Other approximating functions such as \( \kappa_1 - \kappa_2 \mu_i^{(n)} \) and \( \kappa_1 - (\kappa_2 - \kappa_3 \mu_i^{(n)}) \mu_i^{(n)} \) have also been used. It is seen that all the approximations remove the singularity at \( \mu_i^{(n)} = 0 \). At the same time they change, however, the nonlinear transformation into a linear, less powerful, transform.

In order to preserve the powerful feature of the transformation \( \kappa \) should be controlled during the iteration-extrapolation process as indicated by (3.62), (3.96), (3.103), and (3.118).
3.7 Sequences as Mathematical Transients

Fig. 3.13 shows various plots of typical numerical scalar sequences \( \{x^{(n)}\} \) versus \( n \). The smooth curves passing through \( x^{(n)} \) resemble physical transients which can be expressed by

\[
p(t) = y + \sum_{i=1}^{k} a_i e^{a_i t}, \quad a_i \neq 0 \tag{3.121a}
\]

where the \( a_i \) are arbitrary complex numbers, or equivalently

\[
p(t) = y + \sum_{i=1}^{k} a_i q_i^t, \quad q_i \neq 0, 1 \tag{3.121b}
\]

where \( a_i \)'s are the amplitudes, \( q_i \)'s are the quotients, \( y \) is the base of the transient, and \( k \) is its order. The variable \( t \) can be considered as time. The analogy between \( t \) and \( n \) (Fig. 3.13) makes it possible to regard \( \{x^{(n)}\} \) as mathematical transients expressed by

\[
x^{(n)} = y + \sum_{i=1}^{k} a_i q_i^n, \quad q_i \neq 0, 1 \tag{3.122}
\]

Analysis of such a transient can determine its order, its spectrum of \( a_i \)'s and \( q_i \)'s, and - the most desired quantity - the base which is the limit of \( \{x^{(n)}\} \) for \( n \to \infty \) if \( |q_i| < 1 \). In general, the order \( k \) of \( \{x^{(n)}\} \) may be finite or infinite, and also the spectrum may be finite or continuous. But in the computation of fields the spectrum is dominated by a small number of components and \( k \) is finite. Thus, every sequence is a \( k \)'th order transient locally. This is shown in Fig. 3.14.

Consider the \( 2k + 1 \) successive elements of \( \{x^{(n)}\} \) centered around \( x^{(n)} \). Then each element \( x^{(j)} \) can be expressed as

\[
x^{(j)} = y^{(n)} + \sum_{i=1}^{k} a_i^{(n)} (q_i^{(n)})^j \text{ for } q_i^{(n)} \neq 0, 1, n > k,
\]

and \( n - k \leq j \leq n+k \). \( \tag{3.123} \)
Fig. 3.13. Sequences as mathematical transients.

Fig. 3.14. Third-order nonlinear transformation based on $2k+1=7$ elements of $\{x^{(n)}\}$. 
If \( \{x^{(n)}\} \) is nearly of \( k \)'th order than the local bases \( y^{(n)} \) obtained from some transformation may converge to the total base of \( \{x^{(n)}\} \) more rapidly. In the previous sections we have shown that it is possible when the transient is assumed to possess geometric or linear convergence locally, and the local bases are computed by SER. It is now desired to find other transformations of the same type but with higher \( k \).

3.8 Higher Order Nonlinear Transformations

Examination of Aitken's nonlinear transformation reveals that its form may be derived from a ratio of two determinants (another derivation follows from Eq. 3.18; see Appendix II)

\[
y^{(n)} = \frac{\Delta x}{\Delta} = \frac{x^{(n-1)}x^{(n+1)} - [x^{(n)}]^2}{x^{(n-1)} - 2x^{(n)} + x^{(n+1)}}
\]

(3.124)

where

\[
\Delta x = \begin{vmatrix} x^{(n-1)} & x^{(n)} \\ \delta^{(n)} & \delta^{(n+1)} \end{vmatrix}
\]

(3.125a)

\[
\Delta = \begin{vmatrix} 1 & 1 \\ \delta^{(n)} & \delta^{(n+1)} \end{vmatrix}
\]

(3.125b)

and \( \delta^{(r)} \) is the displacement value (see Eq. 3.94). This transformation involves three elements of \( \{x^{(n)}\} \) symmetrically placed with respect to \( x^{(n)} \). Since \( k = 1 \), this is the first order transform. (see Eq. 3.123).
If \( k > 1 \) (Fig. 3.14), then a general formula for \( y_k^{(n)} \) can be derived as (see Appendix 11)

\[
y_k^{(n)} = \frac{\Delta x_k}{\Delta_k}
\]

where

\[
\Delta x_k = \begin{vmatrix}
 x(n-k) & \cdots & x(n-1) & x(n) \\
 \delta(n-k+1) & \cdots & \delta(n) & \delta(n+1) \\
 \vdots & \vdots & \vdots & \vdots \\
 \delta(n) & \cdots & \delta(n+k-1) & \delta(n+k)
\end{vmatrix}
\]

and

\[
\Delta_k = \begin{vmatrix}
 1 & \cdots & 1 & 1 \\
 \delta(n-k+1) & \cdots & \delta(n) & \delta(n+1) \\
 \vdots & \vdots & \vdots & \vdots \\
 \delta(n) & \cdots & \delta(n+k-1) & \delta(n+k)
\end{vmatrix}
\]

It is seen that \( \Delta x_k \) and \( \Delta_k \) are identical except for the first row. The determinant \( \Delta x_k \) can be easily constructed: The first row consists of the elements \( x^{(j)} \) ranging from \( (n-k) \) to \( n \); the second row displacements ranging from \( (n-k+1) \) to \( (n+1) \), and each consecutive row consists of the same number of displacements with the count incremented by 1.

The transform (3.126) can be written in operator form as
\[ y_k^{(n)} = T_k[x^{(n)}] \quad (3.128) \]

Expanding the determinants in (3.126) by their first rows yields

\[ T_k[x^{(n)}] = {c^{(n-k)}x^{(n-k)}} + \ldots + {c^{(n-1)}x^{(n-1)}} + {c^{(n)}x^{(n)}} \quad (3.129) \]

where the c's are the appropriate cofactors, dependent upon the elements \( x^{(j)} \). Thus the nonlinear weighted combination of the \( x \)'s produces an estimation of the local base \( y_k^{(n)} \). It is the \( k \)'th order nonlinear transform.

An interesting property of (3.129) is that for any \( r = 0, 1, \ldots, k \) the elements \( x^{(j)} \) can be shifted to the right (the element count can be increased by \( r \)) without a change in the weights \( c^{(j)} \), that is

\[ y_k^{(n)} = {c^{(n-k)}x^{(n-k+r)}} + \ldots + {c^{(n-1)}x^{(n-1+r)}} + {c^{(n)}x^{(n+r)}} \]

\[ (3.130) \]

The validity of (3.130) can be shown by replacing the first row of \( \Delta_{x_k} \) by the sum of \( (r+1) \) rows of \( \Delta_{x_k} \) and again expanding by the new first row.

It should be noted that the transformation (at least for \( k = 1 \)) can be iterated, that is

\[ y_k^{(n)} = T_k[x^{(n)}] \quad , \quad n \geq k (1.131a) \]

\[ z_k^{(n)} = T_k[y_k^{(n)}] = T_k^2[x^{(n)}] \quad , \quad n \geq 2k (1.131b) \]

\[ c_k^{(n)} = T_k[z_k^{(n)}] = T_k^3[x^{(n)}] \quad , \quad n \geq 3k (1.131c) \]

\[ \vdots \]
This iterated transformation (with $k = 1$) has been successfully tested on a simple optimization problem, in determining the unimodal interval of a function (see Ch. 5).

It should be noted that the higher order nonlinear transformations are very powerful but can be used only for scalar sequences due to the enormous number of arithmetic operations in (3.126). Even the second order transform may be prohibitive for vector sequences.

3.9 Other Extrapolation Formulae

Some operations on Aitken's formula ($T_1$) lead to the following two transformations (see also Appendix 13)

$$y^{(n)} = T_L[x^{(n)}] = \frac{x^{(n)} - \gamma x^{(n-1)}}{1 - \gamma} \quad (3.132)$$

where

$$\gamma = \lim_{n \to \infty} \frac{\delta^{(n)}}{\delta^{(n-1)}} \quad (3.133)$$

and

$$z^{(n)} = T_{1c}[x^{(n)}] = \frac{\epsilon y^{(n)} - x^{(n)}}{\epsilon - 1} \quad (3.134)$$

where

$$\epsilon = \lim_{n \to \infty} \frac{\delta^{(n)}}{\delta^{(n)} y} \quad (3.135)$$

and

$$\delta^{(n)} = \alpha^{(n)} - \alpha^{(n-1)} \quad (3.136)$$

The transformation, $T_{1c}$, is specially useful for very slowly convergent monotonic sequences. Both transformations are amenable for vector sequences. Since $\gamma$ and $\epsilon$ are not known in advance, they can be estimated
from

\[ \gamma^{(n)} = \frac{|\delta^{(n)}|}{|\delta^{(n-1)}|} \]

and

\[ \epsilon^{(n)} = \frac{|\delta_x^{(n)}|}{|\delta_y^{(n)}|} \]  

(3.138)

(3.139)

The derivation of the \( T_L \) transform (Eq. 3.132) follows the proposition of Lyusternik (1947) (see Appendix 13). The \( T_L \) transform is probably the most fundamental of all the extrapolation techniques.

Another class of extrapolation formulae will be presented in the next Chapter. That class of transformations is pertinent to the reduction of the discretization error.

3.10 Conclusions

One can obtain a transformation formula that extrapolates the limit of a deterministic sequence after a finite number of steps. For the related stochastic sequence a modified formula can be obtained which can be applied after a finite number of steps to yield a value close to the limit and thus speed the convergence.

Aitken's formula is an example of transformation for a geometrically convergent sequence. It can be applied to a linearly convergent sequence only for a very restricted range of \( \sigma \)'s. For example, if \( \alpha + \sigma \ll 1 \) and the sequence of \( \sigma \)'s is slowly and monotonically varying to zero, then the Aitken's \( \delta^2 \) process will accelerate the convergence.
Solutions to elliptic partial differential equations lead to linearly convergent vector sequences. The SER method has been applied to Laplace's equation [Della Torre-Kinsner (1973b)] and to Helmholtz's equation [Della Torre-Kinsner, (1973c)] and have achieved significant reductions in computing time by finite difference methods. It is shown here that the SER method converges if the original sequence converges.

An analysis of Aitken's formula, SER and SEOR indicates that SEOR is the best extrapolation technique, is absolutely stable and gives considerable convergence improvement even in the presence of disturbances. The improvement obtained with SER is still considerable as compared with Aitken's formula if the assumed value of α is reasonably close to the actual value. Aitken's formula for extrapolation is essentially unstable in the presence of disturbances and is not particularly suitable for slowly converging sequences when it is needed most.

A general error analysis gave the upper and lower bounds for the generalized perturbation matrix associated with the iteration matrix, and also an expression indicating the sensitivity of the extrapolation factor to the perturbation matrix and the condition number. These results indicate that if the perturbation matrix is large, the extrapolation must be performed with care even though the iteration matrix is well conditioned. The extrapolation factor must also be small when the iteration matrix is ill-conditioned.

Other forms of Aitken's $\delta^2$ process with only one multiplication, instead of two multiplications, have been presented. Still other forms
readily lead to certain approximations which are used in the SER method.

Modification of the basic SER method lead either to schemes with ultimate frequency (one extrapolation every two iterations) or to methods in which the extrapolation factor is a weighted value, depending upon the type of convergence of the background iterative scheme and the need to avoid singularities from that factor. Only two weighted methods, employing the first or second difference vector as the weighting vector, are discussed. Other weighted SER methods are also possible. An error spectrum analysis of these methods indicates the improvement in the total convergence rate for symmetric iteration matrices with positive eigenvalues. Acceleration of the convergence of methods with non-symmetric matrices is also possible.

A concept of mathematical transients has been developed to facilitate the derivation of higher order nonlinear transformations of the Aitken transform type. These transforms are very powerful but, due to their complicated structure, they should be used for scalar sequences only. An iterated transformation (the first or higher order) may be applied to filter out the successive exponentials and finally reduce the transient to its base (or limit).

Two other transformations related to Aitken's formula, $T_L$ and $T_{1c}$, can be used for very slowly convergent both scalar and vector sequences.
3.11 Remarks.

The initial sections of this chapter are based on the following papers: Della Torre-Kinsner (1972a, 1972b, 1973b, 1973c, 1974). The SER method has been extended and applied to various problems in Della Torre-Kinsner (1973a, 1973c, 1973d) and Kinsner-Della Torre (1972, 1973c, 1973d, 1974a, 1974b).

The $\delta^2$- process was first used by Thiele [Bodewig (1959)] in 1909 for the summation of infinite series. It was rediscovered by Aitken (1926) and applied to functions of a single variable. Many other authors rediscovered the same transformation. Samuelson (1945) noted that in the iterative solution of the equation $x = f(x)$ the iterates have nearly geometric convergence. Aitken (1950) suggested an iterative application of his transform to scalar sequences. Higher order nonlinear transformations were discussed by Hartree (1949), Lubkin (1952), Shanks (1954), and others.

Aitken's extrapolation was applied by many authors. Some of them, including some modifications can be found in Kronrod (1960), Evans (1963, 1964), Overholt (1965), Irons-Tuck (1969), Jennings (1971), Boyle-Jennings (1973), Tuff-Jennings (1973).

The Wynn extrapolation (the $e$-algorithm) is closely related to the $T_m^1$ transformation [Wynn (1956)]. It also applies to slowly convergent and divergent sequences. The algorithm has been applied by many authors [Wynn (1962), Pyle (1967), Gekeler (1971, 1972)].

An interesting extrapolation technique, the dominant eigenvalue method, was suggested by Orbach-Crowe (1971) and its extension by Crowe (1972) for nonlinear processes whose convergence is eventually...
Various convergence acceleration techniques for different optimization methods were discussed by Potapova (1970) Demyanov (1971), Kahng (1972), Muda (1973).

A survey of the development of extrapolation processes in numerical analysis as applied to polynomial and rational functions is given by Joyce (1971).

Extrapolation techniques based on the interval variation pertinent to the discretization error (such as the Richardson and Chebyshev methods) will be discussed in the next Chapter.

It is important to note that the name "successive extrapolated relaxation" refers to the extrapolation which is successively repeated and not to the procedure involved in a single extrapolation. A single extrapolation is performed in one step of the iteration (that is, simultaneously as in the Jacobi method - see Sec. 2.5.1.1). Some programming details of the SER method are discussed in Sec. 5.8.
CHAPTER 4

AN ITERATIVE APPROACH TO FINITE ELEMENT METHODS
IN FIELD PROBLEMS

4.1 Introduction

This chapter presents a formulation of finite element methods which makes it possible to use iterative rather than direct techniques for solutions. Combined iterative-direct schemes are also presented. Several finite element formulations are derived for the Laplace, Poisson and Helmholtz equations. These formulations permit iterative solutions. The convergence of vector sequences generated by the iterative method is accelerated using successive extrapolation and other methods. Accuracy and convergence of the solutions are discussed.

The theoretical background of the finite element method has been given by Aubin (1972). Other authors [Oden (1972); Zienkiewicz (1971)] have introduced some practical aspects of the method as applied to structural mechanics. Silvester (1969) and Richards-Wexler (1972), Pontoppidan (1971), Andersen (1972), Silvester et al. (1973) discussed the method as applied to electromagnetic field problems. Convergence of this method, as a function of the relative size of the discretizing elements and the order of the approximating polynomials, is discussed in many recent mathematical and technical journals. In particular, explicit discretization errors are given by Aubin (1972), Oden (1972), Zlamal (1968), and Yamamoto-Tokuda (1971). Some experimental results are given by Richards-Wexler (1972) and Silvester (1969).
The variational formulation of waveguide problems, using complete polynomials, leads to the general eigenvalue problem

$$A\chi = \lambda B\chi$$

(4.1)

where $A$ and $B$ are symmetric positive definite $n \times n$ band matrices, $\lambda$ is the eigenvalue(s), and $\chi$ the eigenvector(s) associated with the particular eigenvalue(s). It is noted that for the $H$ modes $A$ is only semidefinite.

The finite difference formulation of these problems also leads to (4.1), however, the matrix $B$ is the identity matrix, and $A$ is not necessarily symmetric.

A variety of methods for solving (4.1) have been presented [Davies (1972); Ng (1972)]. The finite difference solutions most frequently employ iterative techniques and the finite element solutions almost exclusively use direct methods. One of the most efficient direct methods is first to perform the Choleski triangular decomposition of $B$ which reduces (4.1) to the form

$$C\chi = \lambda \chi$$

(4.2)

with

$$\chi = L^T \hat{\chi}$$

(4.3)

then the Householder tridiagonalization is applied to $C$ and the eigenvalues are located by Sturm sequences and bisection. Finally, the eigenvectors are computed by the Wielandt inverse iteration [Wilkinson (1965)]. It is probably the most efficient method when $A$ and $B$ are full matrices and when all or most of eigenvalues are desired. However,
when $A$ and $B$ are banded matrices, the matrix $C$ is still full since $L^{-1}$ is a full matrix. Therefore, the advantage of the banded form is lost, and even for large computers the upper bound of the number of elements is of the order of 40 [Silvester (1969c)]. We shall discuss a method which takes advantage of the band form of the matrices and obtains either specific eigenvalues or those between prescribed limits.

There are other methods for solving large systems of the form of (4.2) or

$$A\chi = \lambda \chi$$  \hspace{1cm} (4.4)

which use an external slow memory in conjunction with the internal random memory. However, the solutions require a large amount of computer time (up to 4,000 sec. for a system of 2,500 equations on an IBM 360 with IBM 2314 disk packs [Cantin (1971)].

Iterative methods for solving eigenvalue problems may be divided into two categories. The first methods use the fact that eigenvectors of a system form a linearly independent set which spans an $n$ dimensional space. The methods in the second category use the property that the generalized Rayleigh quotient

$$Q = \frac{\chi^T A \chi}{\chi^T B \chi}$$  \hspace{1cm} (4.5)

is equal to an eigenvalue and is stationary when $\chi$ is the corresponding eigenvector. A method using this property with Fletcher-Powell iteration has been described by Brandbury-Fletcher (1966). These methods combined with deflation or orthogonalization yield, however, only partial
eigensolutions, i.e., the dominant and several closest eigenvectors. An iterative method for the complete eigensolution will be presented.

4.2 Finite Element Formulation

Let \( R \) be either a simply or multiply-connected bounded region in an \( n \) dimensional space, \( \mathbb{V}^n \), with boundary \( \Gamma \). The boundary, \( \Gamma \), consists of a finite number of closed, nonintersecting hypersurfaces \( \Gamma_h (h = 0, \ldots, \tau) \) such that \( \Gamma_h \subseteq \Gamma \). Fig. 4.1 shows a two dimensional, nonconvex, multiply-connected region \( R \).

It can be shown [Forsythe-Wasow (1960), p. 182] that the solution of the inhomogeneous Helmholtz equation, \( \nabla^2 \phi + \lambda \phi = g \), subject to the associated natural boundary conditions, \( \phi \frac{\partial \phi}{\partial n} \) equal to zero, [Courant-Hilbert (1953), p. 208; Hazel-Wexler (1972)] is equivalent to minimizing the following functional

\[
F = \frac{1}{2} \int_R (|\nabla \phi|^2 + \lambda \phi^2) \, dR - \int_R \phi g \, dR
\]

(4.6)

where \( \phi \) is the function that is sought, \( g \) is the corresponding source density function, and \( \lambda \) is a constant. Instead of finding the true \( \phi \) over \( R \) the finite-element formulation employs, in general, the subdomains, \( \tilde{E}_e \), such that each \( \tilde{E}_e \subseteq \tilde{R} \). Any proper assembly of elements into a connected model, \( \tilde{R} \), gives an approximate global solution \( \phi \) over the region \( \tilde{R} \).

It is noted that so far there is no restriction on the shape of an element, the type of the approximating algebraic function and the connectivity of the elements. That is, the elements can form simplexes
Fig. 4.1. A general nonconvex two-dimensional multiply connected region with two types of boundary, placed in global coordinates $X,Y$. Triangulation of the region with $M$ elements and $N$ global nodes. Local coordinates $x,y$ are connected with an element $E_0$. 
or multiplexes in the rectangular Cartesian or curvilinear systems, and the space in which the finite element is determined need not even be Euclidean. It also means that one can consider any individual element to be completely isolated from the collection and approximate a function over the element only in terms of the values at its nodes. Furthermore, these nodal values are independent of the ultimate location of the element in the connected model and are also independent of the behaviour of the function within the other finite elements. Thus, it is possible to develop different finite element schemes for the same problem. In the iterative solutions, a proper embedding of the elements expedites the iteration processes.

The discrete model of \( R \) is constructed in several steps: First, a finite number, \( N \), of global nodes is identified in \( R \) and labelled \( P_\xi \) \((\xi = 1, 2, \ldots, N)\). These points may lie in \( R \) or on \( \Gamma \), and are contained in another region \( \bar{R} \), with boundary \( \bar{\Gamma} \), which approximates \( R \). Then, a finite number, \( M \), of elements \( E_\xi \) is defined on the global points. Now, each element is a closed region, and the elements are disjoint. The global nodes associated with a given element are called the local nodes \( P_k \) \((k = 0, 1, \ldots, L)\). Assembling the elements into \( \bar{R} \) completes the process. This assembly employs a grouping of \( m \) elements adjacent to a specific node. There may be \( J \) adjoining nodes.

The procedure for developing a finite element model function, \( \phi \), of a continuous function, \( \varphi \), is closely related to that for \( \bar{R} \) since \( \bar{R} \) is the domain of the function \( \phi \).

For simplicity, only two dimensional regions and triangular elements will be considered. The finite element model function, \( \phi_e \),
within an element shall be represented by the first order complete polynomial of the form

$$\phi_e = a_1 + a_2 x + a_3 y,$$  \hspace{1cm} (4.7)

where \(x\) and \(y\) are the local Cartesian coordinates, and the \(a\)'s are coefficients to be determined by the finite element formulation. Derivations given in the following sections refer to homogeneous, isotropic media.

4.2.1 Laplace's Equation

If \(\lambda = 0\) and \(g = 0\), the differential equation reduces to the Laplace equation. The functional (4.6) is then reduced to

$$F = \frac{1}{2} \iint_{\bar{R}} |\nabla \phi|^2 \, d\bar{\Omega}.$$  \hspace{1cm} (4.8)

In this section, a formula is obtained which minimizes (4.8) and depends only on the angles of the triangular elements.

Let \(E_e\) be a triangle in \(\bar{R}\) with vertices and their associated angles as shown in Fig. 4.2. The subscripts of all the entities in Fig. 4.2 refer to a triangle in \(\bar{R}\) and, at the same time, to a triangle in a group of adjoining triangles (see Fig. 4.1). The local coordinates are located with the \(x\)-coordinate coincident with \(\xi_{0j}\), and \(y\)-coordinate intersecting node \(P_0\). When \(\phi_0\) is perturbed and all other nodal values are kept constant, the functional changes only in the \(m\) triangles adjoining node \(P_0\). If one denotes by \(F_0\) the part of the functional associated with these triangles then
Fig. 4.2. One of \( n \) triangles adjacent to a node \( P_0 \). The origin of local coordinates lies on \( P_1 P_2 \).
\[ F_0 = \sum_{j=1}^{m} \left[ |\nabla \phi_x|^2 + |\nabla \phi_y|^2 \right] A_j, \quad (4.9) \]

where \( \phi_i \) is the component of the gradient in the \( i \) direction \( (i = x, y) \), and \( A_j \) is the area of the triangle. Due to (4.7) one has

\[ \nabla \phi_y = -\frac{\phi_0}{h_{0j}} + \frac{\phi_j}{h_j} \cos \beta_j + \frac{\phi_{j+1}}{h_{j+1}} \cos \xi_j \]

\[ = \frac{1}{\xi_j} \left[ -\frac{\phi_0}{\sin \xi_j} + \frac{\sin \beta_j}{\sin \theta_j} (\phi_j \cot \beta_j + \phi_{j+1} \cot \xi_j) \right]. \quad (4.10) \]

The area of a triangle is given by

\[ A_j = \frac{1}{2} h_{0j} \xi_{0j} \]

\[ = \frac{1}{2} \ell_j \frac{\sin \xi_j \sin \theta_j}{\sin \beta_j} \quad (4.11) \]

By hypothesis

\[ \frac{\partial F}{\partial \phi_g} = 0, \quad \forall g \quad (4.12) \]

everywhere in the domain \( \bar{R} \). For the specific nodal point \( F_0 \), (4.12) can be written using (4.9) in the form

\[ \frac{\partial F}{\partial \phi_0} = \frac{\partial F}{\partial \phi_0} = \sum_{j=1}^{m} 2A \frac{\partial \nabla \phi_y}{\partial \phi_0} \nabla \phi_y \, \text{sgn} \, 0. \quad (4.13) \]
Note that the $x$-component of the gradient vanishes in (4.13) since it is constant for this perturbation. Substituting (4.10) and (4.11) into (4.13) yields

$$
\phi_0 = \frac{\sum_{j=1}^{m} \left( \phi_j \cot \beta_j + \phi_{j+1} \cot \xi_j \right)}{\sum_{j=1}^{m} \left( \cot \beta_j + \cot \xi_j \right)} \quad (4.14a)
$$

$$
= \frac{1}{k_0} \sum_{j=1}^{m} \left( \phi_j k_{j+1} + \phi_{j+1} k_j \right) \quad (4.14b)
$$

where $k_0$ is a constant for a collection of adjoining triangles (all the $k$'s may be computed once for a given geometry), and $\phi_j$, $\phi_{j+1}$ are the nodal values of the approximating function $\phi_0$.

It is readily seen from (4.14) that if $\beta_j = \xi_j$ for all $j$ then (4.14) reduces to

$$
\phi_0 = \frac{1}{m} \sum_{j=1}^{m} \phi_j \quad (4.15)
$$

which is identical to the finite difference formula for an equilateral polygonal mesh. For the square mesh shown in Fig. 4.3, (4.14) becomes the ordinary 5-point Laplacian operator for all nodes no matter how the triangulation of $R$ is performed, and the value $\phi_0$ is the average of the four adjacent values. This implies that the optimum interpolation formula for a finite difference solution using the 5-point operator is the linear interpolation formula. It is noted that there are many different linear
Fig. 4.3. Five-point Laplacian operators resulting from various triangulations of the region:
(a) two types of nodes with 4 adjacent triangles $\bigcirc$, and 8 adjacent triangles $\Box$;
(b) one type of nodes with 6 adjacent triangles.
interpolations for a given set of nodes as shown in Fig. 4.3, however, the optimum values of the nodes is the same in all these cases. For a complete solution of a given problem a consistent manner of interpolation must be used over the whole region $\bar{R}$.

Other implications of (4.14) such as the accuracy and the numerical stability will be discussed in connection with the accuracy of the finite element method.

4.2.2 Poisson's Equation

In this section, the finite element formula will be derived for Poisson's equation using the coordinates of the nodal points instead of the angles associated with the triangles. In this case, the local system of coordinates may be placed anywhere as shown in Fig. 4.4.

For Poisson's equation, the functional (4.6) to be minimized for natural boundary conditions reduces to

$$ F = \frac{1}{2} \iint_{\bar{R}} |v_\phi|^2 \, d\bar{R} - \iint_{\bar{R}} \phi g \, d\bar{R} . \tag{4.16} $$

$F$ is minimized using (4.12). Since a variation of the function $\phi_0$ at a specific node can only affect the energy in the adjoining triangles then, to minimize the functional, it is sufficient to consider these triangles only. Thus

$$ \frac{\partial F}{\partial \phi_0} = \sum_{j=1}^{m} \frac{2}{\partial \phi_0} (r_j + s_j) \phi_0 = 0 , \tag{4.17} $$

where $r_j$ and $s_j$ correspond to the first and the second integral in (4.16)
Fig. 4.4: One of n triangles adjacent to a node $P_0$, placed in general local coordinates $x, y$. 

$\phi_0$, $\phi_1$, $\phi_2$
over the \( j \)-th triangle, and are given by

\[
\begin{align*}
\mathbf{r}_j & = \iint_{E_0} dE_0 \int_0^1 e d\mathbf{e}, \\
n_j & = \iint_{E_0} - g_j \phi_0 dE_0 = -g_j \iint_{E_0} \phi_0 dE_0,
\end{align*}
\] (4.18) (4.19)

where \( e \) denotes the field (function) intensity at any point within the triangle. The form of (4.18) simplifies the analysis of Poisson's equation in inhomogeneous media.

Differentiating and integrating (4.18) yields

\[
\frac{\partial^2 E_j}{\partial \phi_0} = \iint_{E_0} dE_0 \frac{\partial}{\partial E} \int_0^E e d\mathbf{e} \frac{\partial E}{\partial \phi_0}
\]

\[
= A_j E \frac{\partial E}{\partial \phi_0}.
\] (4.20)

Using (4.7) this leads to

\[
E = \left[ \left( \frac{\partial \phi_0}{\partial x} \right)^2 + \left( \frac{\partial \phi_0}{\partial y} \right)^2 \right]^{1/2}
\]

\[
= \left( \alpha_2^2 + \alpha_3^2 \right)^{1/2}.
\] (4.21)
Hence

\[ E \frac{\partial E}{\partial \phi_0} = \frac{1}{2} \frac{\partial E^2}{\partial \phi_0} \]

\[ = \alpha_2 \frac{\partial \alpha_2}{\partial \phi_0} + \alpha_3 \frac{\partial \alpha_3}{\partial \phi_0} \quad (4.22) \]

If \( \alpha_2 \) and \( \alpha_3 \) are expressed in terms of the local nodal values, \( \phi_k \), and their coordinates from (4.7) then

\[ \alpha_2 = \frac{1}{2A_j} \left[ \phi_0(y_1 - y_2) + \phi_1(y_2 - y_0) + \phi_2(y_0 - y_1) \right] \]

\[ = \frac{1}{2A_j} \sum_{k=0}^{2} \phi_k [y_k - y_1] \quad (4.23) \]

and

\[ \alpha_3 = \frac{1}{2A_j} \sum_{k=0}^{2} \phi_k [x_k - x_1] \quad (4.24) \]

where \( k = (\ell + 1) \mod (3) \) and \( i = (\ell + 2) \mod (3) \). Also

\[ A_j = \frac{1}{2} \begin{vmatrix} 1 & x_0 & y_0 \\ 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \end{vmatrix} \quad (4.25) \]

Substituting (4.23), (4.24), (4.22), and (4.25) into (4.20) gives the following expression

\[ \frac{\partial \alpha_j}{\partial \phi_0} = \frac{1}{2} \left[ \alpha_2(y_1 - y_2) + \alpha_3(x_2 - x_1) \right] \quad (4.26) \]
Summing over the $m$ adjoining triangles yields

$$\sum_{j=1}^{m} \frac{\partial r_j}{\partial \phi_0} = \phi_0 \sum_{j=1}^{m} k_{j0} + \sum_{j=1}^{m} (\phi_1 k_{j1} + \phi_2 k_{j2}), \quad (4.27)$$

where the constants $k_{j2}$, which are different for each triangle, are given by

$$k_{j0} = \frac{1}{4A_j} \left[ (y_1 - y_2)^2 + (x_2 - x_1)^2 \right], \quad (4.28a)$$

$$k_{j1} = \frac{1}{4A_j} \left[ (y_1 - y_2)(y_2 - y_0) + (x_2 - x_1)(x_0 - x_2) \right], \quad (4.28b)$$

$$k_{j2} = \frac{1}{4A_j} \left[ (y_1 - y_2)(y_0 - y_1) + (x_2 - x_1)(x_1 - x_0) \right]. \quad (4.28c)$$

The second component, $s_j$, is independent of the shape of the triangle since one can rewrite (4.19) as

$$s_j = -A_j \phi_j \phi_c, \quad (4.29)$$

where $\phi_c$ is the function at the centroid of the triangle, located at $(\frac{1}{3} \sum x_k, \frac{1}{3} \sum y_k)$, and given by

$$\phi_c = \frac{1}{3} \sum_{k=0}^{2} \phi_k. \quad (4.30)$$
Therefore, for a given triangle

\[
\frac{\partial s_j}{\partial \phi_0} = -\frac{1}{6} A_j g_j ,
\]  

(4.31)

and for the \( m \) triangles

\[
\sum_{j=1}^{m} \frac{\partial s_j}{\partial \phi_0} = -\frac{1}{6} \sum_{j=1}^{m} A_j g_j = k_0 ,
\]  

(4.32)

where \( k_0 \) is a constant which may be computed once for a group of adjoining triangles in numerical solutions.

Substituting (4.27) and (4.32) into (4.17) gives

\[
\phi_0 = -\frac{1}{m} \left[ \sum_{j=1}^{m} \left( \phi_{1j} + \phi_{2k} \right) + k_0 \right].
\]  

(4.33)

It can be shown that for a square grid with a mesh length \( h \) (4.33) reduces to the 5-point operator

\[
\phi_0 = \frac{1}{4} \left[ \phi_1 + \phi_2 + \phi_3 + \phi_4 + h^2 g \right].
\]  

(4.34)

4.2.3 Helmholtz's Equation

The derivation of a formula for the homogeneous Helmholtz equation subject to the natural boundary conditions follows the approach outlined previously. For convenience, let the vertex 0 of the triangle
be placed in the origin of the local coordinates. It will be shown that the potential, \( \phi \), at an arbitrary global node, \( P_g \), expressed in terms of both the potentials at the adjacent nodes and the local (or global) coordinates of these nodes is given by

\[
\phi_0 = \frac{1}{\sum_{j=1}^{m} (k_{j0} - \lambda k_{j3})} \sum_{j=1}^{m} \left[ \lambda (\phi_1 + \phi_2) k_{j12} - (\phi_1 k_{j1} + \phi_2 k_{j2}) \right] \quad (4.35)
\]

where

\[
k_{j0} = \frac{1}{4A_j} \left[ (x_2 - x_1)^2 + (y_2 - y_1)^2 \right], \quad VE_j, \quad (4.36a)
\]

\[
k_{j1} = -\frac{1}{4A_j} \left[ (x_2 - x_1)x_2 + (y_2 - y_1)y_2 \right], \quad VE_j, \quad (4.36b)
\]

\[
k_{j2} = -\frac{1}{4A_j} \left[ (x_1 - x_2)x_1 + (y_1 - y_2)y_1 \right], \quad VE_j, \quad (4.36c)
\]

\[
k_{j3} = \frac{1}{6} A_j, \quad (4.36d)
\]

\[
k_{j12} = \frac{1}{2} k_{j3}, \quad (4.36e)
\]

and the area of the triangle is

\[
A_j = \frac{1}{2} (x_1 y_2 - x_2 y_1), \quad VE_j. \quad (4.37)
\]

Equations (4.36a-c) and (4.37) are analogous to (4.28a-c) and (4.24) in the simplified positioning of the triangle.
For clarity, Equation (4.35) may be rewritten as

\[ \phi_0 = \frac{\lambda k_3(\phi) - k_2(\phi)}{k_1 - \lambda k_0} \quad (4.38) \]

where

\[ k_1 = \sum_{j=1}^{m} k_{j0} \quad (4.39a) \]

\[ k_2(\phi) = \sum_{j=1}^{m} (\phi_1 k_{j1} + \phi_2 k_{j2}) \quad (4.39b) \]

\[ k_0 = \sum_{j=1}^{m} k_{j3} \quad (4.39c) \]

\[ k_3(\phi) = \sum_{j=1}^{m} (\phi_1 + \phi_2) k_{j12} \quad (4.39d) \]

For an internal node of a periodic lattice, as shown in Fig. 4.5, Equation (4.39) reduces to

\[ k_1 = 4 \quad (4.40a) \]

\[ k_2(\phi) = -(\phi_1 + \phi_2 + \phi_3 + \phi_4) \quad (4.40b) \]

\[ k_0 = \frac{1}{2} h^2 \quad (4.40c) \]

\[ k_3(\phi) = \frac{1}{6}(\phi_1 + \phi_2 + \phi_3 + \phi_4 + \phi_5 + \phi_6) k_0 \quad (4.40d) \]

and (4.38) then yields
Fig. 4.5. Seven-point finite element operator for the dominant mode solution of Helmholtz's equation.
\[
\phi_0 = \frac{1}{4 - \frac{\lambda}{h^2}} \left[ \frac{h^2}{12} \lambda (\phi_1 + \phi_2 + \phi_3 + \phi_4 + \phi_5 + \phi_6) + \phi_1 + \phi_2 + \phi_3 + \phi_4 \right]
\]

\[
= \frac{1}{8 - \frac{\lambda}{h^2}} \left[ (24 \frac{\lambda h^2}{6}) (\phi_1 + \phi_2 + \phi_3 + \phi_4) + \frac{\lambda h^2}{6} (\phi_5 + \phi_6) \right].
\]  

(4.41)

It is seen from (4.40) that if \[ \sum_{j=1}^{6} \phi_j \] is approximated by \[ 6\phi_0 \] then (4.40) becomes identical to the 5-point finite difference operator:

\[
\phi_0 = \frac{1}{4 - \lambda h^2} (\phi_1 + \phi_2 + \phi_3 + \phi_4),
\]  

(4.42)

which in turn is a special case of a more general finite difference approximation when \((\phi_1 + \phi_2) k_{j12} \approx \phi_0 k_{j3} \).

4.3 Iterative Solutions of \( A\phi = \lambda B\phi \)

An iterative solution to the Helmholtz equation will be given in this section. General solutions of \( A\phi = \lambda B\phi \) have been discussed in Chapter 2.

4.3.1 Dominant Mode

Equation (4.38) is one of the \( N \) equations of the general matrix eigenvalue equation (4.1)

\[
A\phi = \lambda B\phi.
\]  

(4.43)

That is, application of (4.38) or (4.41) to all the \( N \) nodes in \( \bar{\Omega} \) leads to (4.43). It is noted that in (4.43) there is no restriction implied
in the ordering of nodes, however, for iterative techniques some orderings are preferable, as in finite difference methods. The simultaneous and the successive point or block iterative methods (the point or block Jacobi and the point or block SOR methods, respectively) as well as the direct methods can be applied to (4.43) (see Chapter 2). It is also seen from (4.43) that when \( N \) increases then the sparseness of \( A \) and \( B \) increases, thus the iterative methods become more effective.

Equation (4.38) can be rewritten

\[
\phi_g k_{g1} + k_{g2} = \lambda (\phi_g k_{g0} + k_{g3}) , \tag{4.44}
\]

where \( k_{g_i} \) for \( i = 0, \ldots, 3 \) are given by (4.39). Since (4.44) corresponds to the \( g \)-th equation of the system (4.43), the following holds

\[
\lambda = \frac{\sum_{g=1}^{N} \phi_g (\phi_g k_{g1} + k_{g2})}{\sum_{g=1}^{N} \phi_g (\phi_g k_{g0} + k_{g3})} . \tag{4.45}
\]

This is a representation of the generalized Rayleigh quotient (4.5) which can be used to obtain an improved dominant eigenvalue estimate using the iteration scheme shown in Table 4.1.

It is evident from (4.38) and (4.41) that the finite element model, \( \bar{K} \), should be based on a uniform mesh, if possible. The coefficients associated with \( P_g \) should be computed once and stored in a convenient form. Setting initial values of \( \phi \) should follow certain rules so as to obtain the closest approximation of the expected solution. Various
Table 4.1. A flow diagram of an iterative scheme.
iterative techniques may be used in the basic iteration cycle. The expressions (4.38) and (4.41) can be used directly in the Jacobi and the Gauss-Seidel methods. The SOR method requires an estimation of the overrelaxation parameter.

Since the Rayleigh quotient is extremal for the largest eigenvalue, then the method outlined above always converges to the dominant eigensolution. Numerical results of such a solution have been reported [Ahmed-Dahl (1969a)]. It is possible to obtain several higher eigensolutions when using this method [Peters-Wilkinson (1969)], however, the procedure requires a repeated orthogonalization of the lower eigenvectors. It means that these eigenvectors have to be stored, and the error accumulation decreases the accuracy of the successive eigenvectors.

4.3.2 Higher Order Eigensolutions

An iterative method combining the power of the finite element method and the finite difference scheme is presented. An iterative application of (4.38) and (4.45) leads to the dominant eigensolution. However, more information can be retrieved from (4.43) which is constructed iteratively by the application of (4.38) or (4.41), that is, all the eigenvalues of (4.43) can be found without storing the matrices A and B and transforming it to the reduced form (4.2). This is accomplished by the scheme based on a method described by Peters and Wilkinson (1969). The eigenvectors are computed during the subsequent stage by a finite difference scheme, which uses a modified Tee's 17-point operator [Silvester (1970c)].
4.3.2.1 Eigenvalue Computation

The method of computing eigenvalues of (4.43) [Peters-Wilkinson (1969)] is numerically stable, fast, and gives extreme economy of storage. It uses the Sturm sequences to locate the eigenvalues, and combines interpolation with bisection to accelerate the iterative process of finding the specific eigenvalues. In order to use the Sturm sequence count one has to compute all the leading principal minors of $A_n - \lambda B_n$. The minors are not computed explicitly, but only the sign of the $\det (A_r - \lambda B_r)$ with respect to the sign of the $\det (A_{r+1} - \lambda B_{r+1})$ has to be determined. It is noted that the Choleski decomposition of $A_n - \lambda B_n$ cannot be used because this matrix is not in general, positive definite.

There are two methods [Wilkinson (1965), p. 236] which may be used to obtain the required minors. They basically reduce the first $r$ rows as far as possible before introducing row $r+1$. This feature yields extreme economy of storage because for a band of full width $2m+1$ only $m+1$ rows are involved at each stage of the reduction and hence only $(m+1)(2m+1)$ extra locations are required. This is independent of the number of nodal points $N$. There is another important property of this method, that is, any individual eigenvalue can be computed without reference to any of the others.

Obtaining all Eigenvalues Between Prescribed Limits

When all the eigenvalues are required between given limits $a_s$ and $b_t$ where $a_s < b_t$ then these values are taken as upper and lower bounds for the eigenvalues and the method of bisection [Wilkinson (1965), p. 302] combined with the Sturm sequence count is applied until an interval $[a, b]$ is found which contains only one eigenvalue. Then the eigenvalue
is computed in \([a, b]\) to the required accuracy. This is accomplished by finding the zero of either the function

\[
f(\lambda) = \det(A_n - \lambda B_n),
\]

or

\[
g(\lambda) = \frac{\det(A_n - \lambda B_n)}{\det(\lambda_{n-1} - \lambda B_{n-1})}.
\]

Both functions have the same zero, the second being more suitable when \(f(\lambda)\) has a multiple zero or zeros which are pathologically close. The method of bisection is accelerated by the use of successive linear interpolation

\[
\lambda(r+1) = \frac{\lambda(r)f(\lambda(r-1)) - \lambda(r-1)f(\lambda(r))}{f(\lambda(r-1)) - f(\lambda(r))}.
\]

At the beginning of this iteration three values of \(\lambda\) are required. They are taken from the interval \([a, b]\) determined by the Sturm sequence count based on \(f(\lambda)\). The iteration process for finding one eigenvalue is terminated when the following criterion is satisfied

\[
(\lambda(r) - \lambda(r-1)) < \epsilon,
\]

where

\[
\epsilon = 4\epsilon_1|\lambda(r)| + \epsilon_2,
\]

\(\epsilon_1\) is the machine precision, and \(\epsilon_2\) is the permissible absolute error in the eigenvalue of smallest modulus.
Obtaining Specific Eigenvalues

When specific eigenvalues are required, for a given \( \lambda_s \) and \( \lambda_t \) such that \( \lambda_s \leq \lambda_{s+p} \leq \ldots \leq \lambda_t \), then the upper and lower bounds for \( \lambda_t \) and \( \lambda_s \) must be obtained. It is noted that not all eigenvalues in the range need be computed. The bounds are obtained by the Sturm sequence which is computed for

\[
a^{(i)} = -(4^i c), \quad i = 0, \ldots
\]  

(4.51a)

until a lower bound is found for \( \lambda_s \), and

\[
b^{(i)} = 4^i c, \quad i = 0, \ldots
\]

(4.51b)

until an upper bound is obtained for \( \lambda_t \). The constant \( c \) in (4.51) is defined by

\[
c = \frac{\max \sum |a_{ij}|}{\max \sum |b_{ij}|}
\]

(4.52)

It is seen from (4.52) that the row norms of the matrices \( A \) and \( B \) can be computed in one sweep of all the nodal points.

The outlined method of computing the zeros of the function \( f(\lambda) \) is more efficient than the Newton method [Peters-Wilkinson (1969)] and the method used in the PDSOR [Beaubien-Wexler (1971a, 1971b)]. It is apparent that this scheme can be used in the finite difference methods as well.
4.3.2.2 Computation of Eigenvectors

In order to obtain eigenvectors corresponding to eigenvalues higher than the dominant value, one could apply operator squaring in the same sense as in Beaubien-Wexler (1971a) and Silvester (1970c) for finite-difference methods. This would lead to an all finite element eigensolution. Here, a new finite difference scheme is proposed to compute higher eigenvectors to greater accuracy once the initial solution is obtained by the finite element method.

For reasons discussed by Silvester (1970c), a 17-point operator is used in this method for solution of \((v^2 + k^2)k = 0\). The use of such an operator has been indicated by Della Torre-Kinsner (1973c) in connection with higher waveguide mode solutions. This operator is obtained by a linear combination of one of Tee's 17-point biharmonic operators with the 5-point Laplacian, according to the formula

\[
\varphi_i = v^4 + 2k_{i-1}^2 v^2 + k_{i-1}^4,
\]

(4.53)

where the eigenvalue to be determined is

\[
\lambda_i = -(k_{i-1}^2 - k_i^2)^2,
\]

(4.54)

and for the dominant eigensolution \(\lambda_1 = -k_1^2, \varphi_1 = v^2\) in the scheme

\[
\varphi_i^{(n+1)} = \lambda_i \varphi_i^{(n)}.
\]

(4.55)

The resulting operator is shown in Fig. 4.6 where \(m = k_{i-1}^2\). Thus the 17-point finite difference representation of (4.55) is given by
\[
\frac{1}{108} \times
\]

\[
s = 216 \, m - 177
\]

\[
t = 108m^2 - 864m + 640
\]

Fig. 4.6. A 17-point finite difference operator. It is positive definite, has Property A and improved diagonal dominance.
\[ \phi_g = \frac{-108}{t} \left[ 8 \sum_{j=1}^{4} \phi_j + 11 \sum_{j=5}^{8} \phi_j + 3 \sum_{j=9}^{16} \phi_j \right]. \] (4.6)

If the SOR method is used to obtain \( \phi_g \),

\[ \phi_g^{(n+1)} = \phi_g^{(n)} + \omega(\phi_g^{(n+1)} - \phi_g^{(n)}) \]

\[ = (1 - \omega) \phi_g^{(n)} + \omega \phi_g^{(n+1)}, \] (4.7)

where \( \omega \) is the accelerating factor, then for an interior point at least 3h distant from the Dirichlet or Neumann boundary conditions (4.6) can be used in (4.7). This formula generates a system of equations of the form (4.4) where \( A \) is positive definite for higher eigensolutions. The operator (4.6) has Young's Property A [Young (1971) p. 148] thus allowing an optimization of \( \omega \) to be fully employed. In addition, it possesses an improved diagonal dominance with a higher convergence rate. The operator has a discretization error of order \( O(h^2) \).

According to Young (1971) p. 114, for every fixed value of \( \omega \) the convergence rate is maximized if the equations and the unknowns of (4.4) are permuted into a consistent ordering. There are different consistent orderings but the only type of this class of orderings which can be applied with the operator (4.6) is the red-black ordering [Young (1971) p. 159].

A special treatment has to be given to the nodal points less than 3h distant from \( \Gamma \). In order to avoid the necessity of using an unequal arm 17-point operator which complicates the logic of programming a given problem, a possible technique is to use a triple layer of
boundary points with the same boundary values as \( \phi \) on \( \Gamma \). Alternatively, one can switch to other operators when the internal nodes less than \( 3h \) distant from \( \Gamma \) are encountered in the scanning procedure, that is, for points \( 2h \) distant one can switch to the 13-point operator given by Beaubien and Wexler (1971b), and for nodes separated from \( \Gamma \) by \( h \) or lying on \( \Gamma \) the 5-point operator can be used.

Since the 13-point operator does not possess Property A then optimization based upon the explicit analysis of the eigenvalues of the iteration matrices of \( \omega \) is not possible. Also, the 5-point operator leads to the dominant eigensolution (even though it does posses the Property A). This treatment is better than the triple extended layer because the ratio of boundary points to all the interior points is \( \sim \frac{1}{N} \) where \( N \) is the total number of points, and hence the effect of operator switching decreases when \( N \) increases. This approach does not require extra storage for the triple boundary layer outside \( \Gamma \). Also, the triple layer implies \( \partial \phi / \partial n \) and \( \partial^2 \phi / \partial n^2 \) are both zero.

4.3.2.3 Combined Finite-Element-Difference Method

Other techniques which combine the direct-finite-element method, as presented by Silvester (1969b), with the finite difference method outlined above are also attractive. The finite element method, when used with high order approximating polynomials, is accurate and fast because assembling of the matrices uses invariant numerical matrices that are stored in the computer and are independent of the problem to be solved.
This method gives a complete eigensolution. However, the storage limitations put an upper bound on the accuracy of the solutions.

In order to obtain higher accuracies one can first obtain a finite element solution (over a nonuniform mesh), then impose either a uniform or graded fine mesh and subsequently compute initial nodal values of $\phi$ on the fine mesh from the finite element solution, and finally proceed with an iterative scheme using the 17-point operator to obtain a better approximation of $\phi$. This combined method is extremely efficient since the finite difference method starts from a very good initial eigenvector and initial eigenvalue.

4.4 Acceleration of Convergence

There exist two general approaches to the problem of accelerating the convergence of the finite element method to reduce the computation time required for a complete or partial eigensolution to a given accuracy. The first one refers to accelerating the direct solution of $A_\varepsilon \phi = \lambda B_\varepsilon \phi$ as given in existing programs [Silvester (1969c)] and the other refers to accelerating the iterative methods as described in the previous section.

It is noted that convergence is understood here as the diminishing of some norm of the error between the initial solution and the approximate solution $\phi$ on $\mathbb{R}$ rather than between $\phi$ and the true solution $\varphi$ on $\mathbb{R}$ when either the largest side or the minimum angle of all of the triangles is varied.
4.4.1. Wilkinson's Algorithm

In the first approach, the most efficient algorithms [e.g., Peters-Wilkinson (1969)] for obtaining eigenvalues of the generalized matrix eigenvalue problem, in which the matrices are of a band-type and only specific eigenvalues are required, should be used in the direct methods. The eigenvectors are then computed by the Wielandt inverse iteration (see Chapter 2).

4.4.2. Successive Extrapolation

The other technique relates to accelerating convergent vector sequences generated by iterative methods. The SER formula (see Eq. 3.22) and the discussion of Chapter 3 applies here. It might be recalled that a ten-fold improvement in convergence rate results if the errors are less than 4 percent and the approximation to $\alpha$ is within 5 percent.

When a linearly convergent vector sequence oscillates globally or locally then the effectiveness of the successive extrapolation method decreases. However, it is seen from (3.22) that if this method cannot improve the convergence of the original sequence then it does not decelerate it. Both oscillations and their frequencies of the vector sequences depend on the eigenvalues of matrices involved in the processes generating these sequences. If the eigenvalues are complex then a vector sequence oscillates. The successive extrapolation process (3.22) improves convergence of such sequences providing the period of the oscillations is longer than 5 to 10 iterations. These frequencies are
often lower in the iterative processes [Ashkenazi (1971)]. It is possible to remove the oscillations, at least partially, by the following method.

One term of a vector sequence is determined in two half iterations: the first sweep of all the nodal points produces \( \xi^{(n+1/2)} \) and the second scan, performed in an order opposite to the first sweep, gives \( \xi^{(n)} \). This process can be written as

\[
\xi^{(n+1/2)} = L_f \xi^{(n)} + \xi_f
\]

and

\[
\xi^{(n+1)} = L_b \xi^{(n+1/2)} + \xi_b
\]

where \( L \) is an iteration matrix, and the subscripts \( f \) and \( b \) correspond to forward and backward direction of a scan, respectively. When the SOR method is used in this process (including the Gauss-Seidel method) then it is called symmetric SOR method (SSOR) [see Ch. 2]. It can be shown that for any \( 0 < \omega < 2 \) the eigenvalues of \( L \) are real, positive and less than one. Hence oscillations are removed. This method may, in general, converge less rapidly than SOR, nevertheless the convergence of the well-behaved vector sequence generated by the SSOR method can be accelerated by the use of the successive extrapolation. This method and its variants have been derived and satisfactorily tested.

4.4.3. Chebyshev Acceleration

When some information concerning the distribution of the eigenvalues, \( \mu_L \), of the iteration matrix is available, and the eigenvalues are real then the Chebyshev polynomials [Young (1971) p. 301]
can be used to accelerate the convergence of different methods such as the Jacobi, SOR, Richardson's, and various second-degree iterative methods (see Ch. 2). This method assumes the convergence of a vector sequence upon which it should act. The strategy is to find a function which would not only describe the convergence property of the vector sequence based on the eigenvalues of \( \lambda \) but, at the same time, would cause some acceleration of the convergence by projecting the terms of the sequence towards its limit. Since for a convergent sequence the spectral radius of \( \lambda \) has to be less than one then the Chebyshev polynomials offer the means of weighting the projections. When the eigenvalues \( \lambda_i \) are distributed between \( 0 < \| \lambda_i \| < 1 \) where \( a \) is a constant then the appropriate linear combinations of some terms of the vector sequence \( \{ \hat{x} \} \) may be obtained by means of the three-term recurrence relation

\[
\hat{x}^{(n+1)} = \hat{x}^{(n)} + \frac{T_{n-2}(1/a)}{T_{n}(1/a)} [\hat{x}^{(n)} - \hat{x}^{(n-2)}]
\]  

(4.60)

where \( \hat{x}^{(n)} \) is the vector obtained by (4.58) and \( T_n \) is the Chebyshev polynomial of degree \( n \) in \( x \) defined by \( T_n(x) = \cos(n \cos^{-1} x) \) in the range \( |x| < 1 \).

The Chebyshev acceleration of Richardson's method,

\[
\hat{x}^{(n+1)} = \hat{x}^{(n)} + \beta^{(n)} \hat{x}^{(n)}
\]  

(4.61)

where \( \hat{x}^{(n)} \) is the residual vector at \( n \)-th iteration has the form of the second-degree Richardson's method,

\[
\hat{x}^{(n+1)} = \hat{x}^{(n)} + \beta^{(n)} \hat{x}^{(n)} + \gamma^{(n)} \hat{x}^{(n)}
\]  

(4.62)
where \( \mathbf{\hat{c}}^{(n)} = \mathbf{c}^{(n)} - \mathbf{c}^{(n-1)} \) is the displacement vector and the coefficients are given by

\[
\beta^{(n)} = \frac{4 \cosh nt}{b - a \cosh (n+1)t}, \tag{4.63a}
\]

\[
\gamma^{(n)} = \frac{\cosh (n-1)t}{\cosh (n+1)t}, \tag{4.63b}
\]

where \( \cosh t = \frac{b + a}{b - a} \), and \( a, b \) are the smallest and largest eigenvalues of \( L \). This process is nonstationary since the values of \( \beta \) and \( \gamma \) depend on \( n \).

The Chebyshev acceleration of SSOR when applied for a biharmonic problem causes its convergence to be twice as fast as SOR [Engeli et al. (1959)]. Young (1971) estimates an order faster convergence when this type of acceleration is applied to some methods.

A serious limitation on this method is posed by the lower bound, \( a \), for \( \mu \). It is easy to find a close estimate to the upper bound, \( b \), but determination of \( a \) is very difficult. It appears that the effectiveness of the method depends mainly on \( a \).

4.4.4. Gradient Methods

General properties of these methods have been discussed in Sec. 2.5.1.5. Let \( f(\mathbf{\hat{c}}) \) be a quadratic function defined by

\[
f(\mathbf{\hat{c}}) = \mathbf{\hat{c}}^T A \mathbf{\hat{c}} - 2 \mathbf{\hat{c}}^T \mathbf{p} + \mathbf{p}^T A^{-1} \mathbf{p} \tag{4.64}
\]

where \( \mathbf{\hat{c}}^T \) denotes transpose of \( \mathbf{\hat{c}} \), and \( A \) and \( \mathbf{p} \) correspond to the quantities in (4.4). When \( A \) is symmetric and positive definite then the \( f(\mathbf{\hat{c}}) \) takes
its minimum value, zero, at $A^{-1}p$. Various minimization methods of $f(\varphi)$ have been discussed by Bandler (1969). Interesting combinations of iterative methods (accelerated Jacobi, SSOR) with some simple gradient methods and with the Chebyshev acceleration have been discussed, and corresponding numerical results on the biharmonic operator have been presented by Engeli et al. (1959).

The major feature of some of the gradient iterative methods is that they lead to the solution in a finite number of steps. One can use this powerful feature of the gradient methods by combining the best minimization techniques (see Bandler (1969)) with the finite element method for Laplace's, Poisson's equations or several eigensolutions of the Helmholtz equation. Such a combined method should start from a direct or iterative solution of the finite element equations. The process then could proceed with an iterative refinement by a finite difference method on a fine mesh, however, this intermediate step is not necessary. Finally, the process uses one of the most efficient gradient methods to find the minimum of (4.64) or another objective function associated with the given problem. These minimization iterations would start from an initial vector close to the solution thus the minimum would be obtained quickly and the process would be reliable since (4.64) represents, in this case, a set of concentric hypersurfaces. It is noted, however, that the very efficient gradient methods require existence of the first and second derivatives of a function. The model function, $\varphi$, constructed in the finite element is continuous but the first derivative is discontinuous at the boundary between the elements,
thus the second derivative may be indeterminate at these boundaries. This restriction can be removed by redefining the function to be continuous with continuous derivatives. This model of $\xi$ describes the actual $\xi$ more accurately than the finite element model, and, therefore, the gradient vector containing the first partial derivatives and the Hessian matrix with second partial derivatives can be constructed.

The number of iterations can be reduced considerably when using the gradient methods but the amount of work involved in computing the gradient vector and the Hessian matrix may actually decrease the efficiency of the method below that of other simpler methods.

4.4.5. Mesh Refinement

Another technique for accelerating the convergence is mesh halving. In general, an application of bisection to all the sides of triangles or rectangles as shown in Fig. 4.7 leads to a finer mesh. Since the convergence depends on the starting values at the nodal points then a quick solution on a coarse mesh with small number of nodes can serve as the starting values for the next cycle of iterations. This mesh halving can be repeated leading to a better first approximation to $\varphi$ (see next section). The solution obtained by mesh halving is faster than that carried on the finer mesh only.

It is noted that the overrelaxation factor in SOR or the extrapolation factor in SER depends on the mesh size; it usually decreases when the mesh size decreases. Each mesh halving by bisection multiplies the number of elements by 4 implying an upper bound on the maximum number
Fig. 4.7. Element subdivision by bisection of its sides.
of elements by the available storage.

The advantage of using bisection rather than other techniques of the mesh refinement is that the bisection maintains the same angles in the new elements. This fact removes the necessity of recomputing all the coefficients, \( k \), in (4.14). The "old" nodes have exactly the same \( k \)'s.

Suppose the discretization error of the particular algorithm used in the finite-element solution of \( \chi \) is of order \( ||h||^p \) where \( ||h|| \) is the Chebyshev or some other norm of the side length of all the triangles, and \( p \) is some number. Let \( \{\phi_k\} \) be a sequence of solutions, generated by the same algorithm, using a corresponding sequence \( \{h_k\} \) of decreasing \( h \) on the same region \( \bar{R} \). When

\[
\{h_k\} \to 0 ,
\]

then

\[
\{\phi_k\} \to \chi ,
\]

(4.66)

Since \( \{\phi_k\} \) converges monotonically to \( \chi \), the following relation holds

\[
\frac{||\phi_{k+1}||}{||\phi_k||} = \frac{||h_{k+1}||^p}{||h_k||^p} ,
\]

(4.67)

where the discretization error vector \( \phi_k \) at \( k \)-th subdivision of the elements is given by

\[
\phi_k = \chi_\infty - \chi_k ,
\]

(4.68)

Any norm of \( \phi_k \), including the vector itself, satisfies (4.67) since \( p \) depends on the choice of the norm. Hence, one can extrapolate the
sequence (4.66) from two or more successive solution vectors, $\hat{\chi}_k$, obtained on two or more triangulations with decreasing size of the elements. For only two solutions ($k=1,2$) one obtains from (4.67) and (4.68) the following extrapolation formula

$$\hat{\chi}_E = \frac{\hat{\chi}_1 \|h_2\|^P - \hat{\chi}_2 \|h_1\|^P}{\|h_2\|^P - \|h_1\|^P}$$

(4.69)

where $\hat{\chi}_E$ indicates the extrapolated vector, not necessarily $\chi$. This is caused by various factors affecting ($\hat{\chi}_k$) such as the cumulative round-off errors, and the residual error of $\hat{\chi}_k$. A better extrapolation formula results from three solutions ($k=1,2,3$)

$$\hat{\chi}_E = \frac{\hat{\chi}_1 s_1 + \hat{\chi}_2 s_2 + \hat{\chi}_3 s_3}{s_1 + s_2 + s_3}$$

(4.70)

where

$$s_\ell = \frac{\|h_{\ell}\|^P - \|h_n\|^P}{\|h_n\|^P}$$

(4.71)

and $\ell, m, n$ are an even permutation of $1, 2, 3$.

The extrapolated values, $\hat{\chi}_E$, in general, do not result in $\chi$ even though $p$ is known in advance. However, when the cumulative round-off error is small and when the change in $\|h_k\|$ is appropriately chosen, then a significant improvement in accuracy is achieved by this extrapolation. An assessment of $p$ will be given in the following section.

4.5 **Accuracy and Convergence**

At the beginning of the previous section, two concepts of
convergence were introduced: (i) the convergence of $\chi$ to $\psi$, and (ii) the convergence of successive estimates of $\chi$ to the model function solution $\chi$. Similarly, there are two types of errors: (i) the accuracy of the model function $\chi$ related to the original function $\psi$, i.e., discretization error, and (ii) the accuracy of the estimate solution $\hat{\chi}$ with respect to the actual function $\chi$, i.e., residual error. This distinction has been neglected in some cases and the finite element results presented were not as accurate as expected [Fuchs-Erdelyi (1973)].

4.5.1. **Discretization Error**

Recall that, by assumption, the function $\psi$ is continuous with continuous partial derivatives in a region $\Omega$ bounded by $\Gamma$. When triangulation of $\Omega$ is employed and the first order polynomial is used to approximate $\psi$ within each triangle then the model polyhedral function $\hat{\psi}$ is continuous with continuous first derivatives within each triangle and discontinuous at junctions between the triangles. The second partial derivatives vanish within each triangle and may be, in general, infinite at the junctions. We shall now show that a given function $\psi$ and its first partial derivatives may be approximated arbitrarily closely by a polyhedral function $\hat{\psi}$ based on a suitable triangulation of $\Omega$.

The strategy of assembling $\hat{\psi}$ (see Sec. 4.2) guarantees the continuity of $\hat{\psi}$ and an appropriate "fitting" of $\phi$ to $\psi$. Therefore, only an arbitrary, closed triangular element, $E_c$, is considered in the proof. Let $E_c \subset \Omega$ and the vertices $P_i(x_i, y_i) \in E_c$. Let $\phi_i \in \psi$, that is, the
values of the model function at the vertices coincide with the exact
function $\varphi$. This result is obtained by minimization of the functional
(4.8) for each nodal point and for the linear interpolation over $E_e$.
For higher approximations more points have to satisfy this coincidence
condition. It is noted that this minimization process is similar to
the least squares. It might be possible to construct another functional
(objective function) whose least $p$th minimization could lead to a better
approximation.

The accuracy of the approximation $\phi_e$ given by (4.7) and its
first derivative, $\phi'_e$, to $\varphi$ and $\varphi'$ depends on the smoothness of $\varphi$. Let
the second derivatives of $\varphi$ be bounded by

$$\left| \frac{d^2 \varphi}{d \xi^2} \right| \leq M$$  \hspace{1cm} (4.72)

where $M$ is an arbitrary positive number and the derivative is calculated
at any point along an arbitrary straight line as shown in Fig. 4.8a.

Since

$$\frac{d^2 \varphi}{d \xi^2} \lambda_k = \frac{\partial \varphi}{\partial \xi_i} \frac{\partial \varphi}{\partial \xi_j} \lambda_i \lambda_j$$

$$= \varphi'_{ij} \lambda_i \lambda_j$$  \hspace{1cm} (4.73)

where $\xi_k$ are general coordinates and $\lambda_k$ are unit vectors associated
with these directions, then (4.72) is equivalent to

$$|\rho'_{ij} \lambda_i \lambda_j| \leq M.$$  \hspace{1cm} (4.74)
Fig. 4.8. Triangular element in general coordinates (a), and related hexagon (b).
The angle, $\beta$, between $\xi_i$ and $\xi_j$ is arbitrary. Hence

$$|\rho'_{ij} \quad \xi_i \xi_j | \leq M.$$  \hfill (4.75)

Let the error function over the element $E_e$ be defined as

$$e = \phi - \varphi.$$  \hfill (4.76)

For convenience, the subscripts $e$ are omitted in (4.76). Note that $e = 0$ at the vertices $P_i$. Since $\phi$ is a linear approximation then

$$e_{ij}' = -\rho'_{ij}.$$  \hfill (4.77)

Substituting (4.77) into (4.75) yields

$$|e_{ij}' \quad \xi_i \xi_j | \leq M.$$  \hfill (4.78)

Since $e = 0$ at any vertex there exists a point, $Q$, on any side $h_i$ of the triangle such that

$$e_h' \quad h_i = 0$$  \hfill (4.79)

where $h_h$ is a unit vector along $h_i$. Let $P$ be an arbitrary point such that $P \in E_e$. Then

$$\left( e_{h} \quad h_h \right)_P = \int_Q \frac{d}{dx} \left( e_{h} \quad h_h \right) d\xi = \int_Q e_{h} \quad h_h \quad h_h d\xi = \left( e_{h} \quad h_h \quad h_h \right)_P$$  \hfill (4.80)

Substituting (4.80) to (4.78) gives

$$|e_{h} \quad h_h |_P \leq M_h \text{ or } |\nabla e_{i} \quad \xi_i |_P \leq M_h$$  \hfill (4.81)
where \( h \) is the largest side of the triangle.

Eq. (4.81) is essential in this analysis. It indicates that the orthogonal projection of \( \nu e \) on any side of the triangle cannot exceed \( M_h \). This formula bounds the longest side of the triangle to the upper bound of an element of the gradient error vector at any point within the element. It is now apparent that the gradient vector can be introduced into (4.81). The maximum allowable gradient error vector is connected with the angles of the triangle in such a way that its projection is bounded by \( M_h \). This relation can be found geometrically by examining Fig. 4.8b. The graph is constructed in the following sequence. First, a circle is drawn of radius \( M_h \). Then a hexagon is circumscribed on this circle such that each pair of the sides of the hexagon is parallel to the given triangle in Fig. 4.8a. It is seen that the hexagon is symmetrical and it has all the angles of the triangle well specified. In order for an orthogonal component of the gradient error vector to be bounded by \( M_h \) (the radius of the inscribed circle), the gradient vector itself cannot lie outside the hexagon, if its beginning is adjoined to the centre of the circle. The vertices of the hexagon lie at distances

\[
\frac{M_h}{\cos \frac{1}{2} \theta_i}, \quad i = 0, 1, 2.
\]  

(4.82)

It follows that

\[
|\nu e| \leq \frac{1}{\cos \frac{1}{2} \theta} M_h
\]

(4.83a)

or

\[
|\nabla (\phi - \psi)| \leq \frac{1}{\cos \frac{1}{2} \theta} M_h
\]

(4.83b)
where \( \theta \) is the maximum angle in the triangle.

The bound for the approximation of the function itself can be obtained by integration of the derivative of the error function in any direction along this direction, starting from a vertex of the triangle where \( c = 0 \) and \( e_1' = 0 \).

\[
e_{L}^{p} = \int_{c}^{p} e_{L}^{'} dz .
\]

To get the upper bound, integration has to be performed in the direction of the maximum change of the error function. Hence

\[
\| \phi - \psi \|_{p} = \| \nabla (\phi - \psi) \|_{\max} h.
\]

Substituting (4.83) into (4.85) gives

\[
\| \phi - \psi \| \leq \frac{1}{\cos \frac{1}{2} \theta} M h^2
\]

If one defines \( h \) and \( \theta \) as the largest side and the largest angle of all triangles being associated with one triangle, then the following conclusions hold. The approximation for the derivatives is of order \( h \) and the approximation for the function is of order \( h^2 \). Eq. (4.83) and (4.86) indicate that the function \( \psi \) and its derivatives \( \psi' \) can be approximated by the first order polynomials based on a triangular mesh arbitrarily close by taking a sufficiently fine triangulation. It is noted that flat triangles give a poor approximation, since \( \cos \frac{1}{2} \theta \to 0 \).

Eq. (4.86) shows that the approximate solution \( \phi \) converges to the exact solution \( \psi \) with the same order of the convergence rate as in the case of the finite difference approximation. However, there is an
important difference between the two methods, evident from this analysis, that is, this order of the convergence rate in (4.79) requires a bounded second derivatives of $\varphi$ whereas, in the finite difference formulation, a bound on the third order derivatives is required.

Eq. (4.86) can be generalized by removing the assumption that the largest angle $\theta$ is associated with the triangle having the longest side. In this case, however, the derivation becomes complicated since the convergence has to be expressed in terms of a norm in the Sobolev space of functions having generalized derivatives up to the order $k$ inclusive, on a compact support [Aubin (1972) p. 138]. Using these norms, Zlámal (1968) has shown that for second order finite element approximation to the generalized second order partial differential equations the following is true

$$
|\psi - \varphi|_S \leq k \frac{1}{\sin \nu} M_3 h^2
$$

(4.87)

where the constant $k$ does not depend on the triangulation, $\nu$ is the smallest angle of all triangles, $h$ is the largest side of all triangles, and the subscript at $M$ indicates the order of the derivatives that have to be bounded. The subscript $S$ indicates a norm in the Sobolev sense.

The approximation by cubic polynomials leads to the following

$$
|\psi - \varphi|_S \leq k \frac{1}{\sin \nu} M_4 h^3
$$

(4.88)

For the same problems, the finite difference formulation gives the convergence rates of the order of $h$ and $h^2$, respectively.

For the biharmonic equation approximated by the fifth order polynomial one gets [Zlámal (1968)]
\[ | \theta - \varphi | \leq k \frac{1}{\sin^2 \vartheta} M_0 h^4. \quad (4.89) \]

It is noted that the largest angle \( \theta \) in (4.86) can describe a triangle better than the smallest angle \( \varphi \) since the latter indicates one small angle only, whereas the former always implies two small angles when \( \vartheta = 180^\circ \).

### 4.5.2 Residual Error and Error Criteria

In order to terminate an iterative process at some accuracy it is necessary to determine some measure of the accuracy. The simplest measure of the error is the residual measure \( e_R \) which indicates the fractional deviation of \( \mathcal{V}^2 \phi(n) \) from zero at \( n \)-th iteration

\[ e_R = \frac{|| \mathcal{V}^2 \phi(n) ||_E}{|| \mathcal{V} \phi(n) ||_E}. \quad (4.90) \]

This ratio employs the Euclidean norms with sieving function, and it can be shown that it reduces to

\[ e_R = \left| \sum_{g=1}^{N} \left( \frac{(\mathcal{V} \phi(n)_g)^2}{\mathcal{V} \phi(n)_g} \right)^{1/2} \right|. \quad (4.91) \]

Another measure is defined by

\[ e_F = \frac{\int_{\Gamma} \frac{\partial \phi(n)}{\partial n} |\Gamma| \, d\Gamma}{\int_{\Gamma_i} \frac{\partial \phi(n)}{\partial n} |\Gamma_i| \, d\Gamma_i}. \quad (4.92) \]
It shows the flux imbalance between the flux calculated along the entire boundary, \( \Gamma \), and the flux which enters the region across \( \Gamma_i \).

Since the exact solution \( \varphi \) is not known in general, it is impossible to relate the error norms \( e_R \) and \( e_F \) to an error norm of the exact solution, e.g.,

\[
e_{\varphi} = \frac{||\varphi^{(n)} - \varphi||}{||\varphi||},
\]

nevertheless some test problems for which the exact solution is known can qualitatively relate (4.91) and (4.92) to (4.93). The knowledge of this relation may be applied to other problems. It has been experimentally found [Richards-Wexler (1972)] that, for homogeneous and inhomogeneous Dirichlet, Cauchy, and Neumann problems, \( e_R \) and \( e_F \) are approximately proportional to \( e_{\varphi} \), and they decrease when the order of the approximating polynomial increases.

4.6 Conclusions

An iterative approach to the finite element method is presented in which no matrices and essentially only the solution vector has to be stored. Various techniques have been presented for assembling the system matrix. These are useful for iterative solutions and some are particularly useful in element subdivision. The relationship between the finite element, the finite difference, and the gradient methods is discussed.

Various techniques have been presented for convergence acceleration of the vector sequence to its limit which is the exact solution to the modelling problem (residual error), and accelerating the convergence of
the sequence of modelling solutions to its limit which is the exact solution of the field problem (discretization error). Of these techniques successive extrapolation is the most effective in obtaining the limit of a vector sequence. In order to obtain the limit of the sequence of modelling function one must use successive mesh refinement. A technique for element subdivision is presented which introduces new elements with no new angles thereby simplifying the obtaining of the coefficients of the new assembly matrix whose order is twice that of the original assembly matrix.

An analysis of discretization shows that the convergence rate varies as $h^2$. This permits to obtain an optimal extrapolation technique for estimating the exact solution with finite size elements.

An error analysis which has been carried out shows that it is possible by use of iterative techniques to attain higher accuracies than could be attained by use of the direct solution. For higher order matrices, required for high resolution and/or higher order waveguide modes, the iteration method presented here is more than an order of magnitude faster than direct methods.
4.7 Remarks

The material included in this Chapter is partially based on Della Torre-Kinsner (1973d) and Kinsner-Della Torre (1973d, 1974a).

Literature on finite element methods, both books and papers, grows rapidly. They still largely refer to structural mechanics although other areas are also being covered. Many works from the period 1967 to 1973 were cited in Chapter 1. It is worthwhile to mention two new books by Martin-Carey (1973) and Norrie-de Vries (1973), and a collection of papers presented at the Symposium on the Mathematical Foundation of the Finite Element Method with Applications to Partial Differential Equations edited by Aziz (1972). Numerous meetings and conferences are being currently held at different places, and are devoted to both the theoretical foundations and applications of the method.

Chebyshev acceleration is discussed by Wrigley (1963), Ehrlich (1972), and Bulirsch-Stoer (1966, 1967). Richardson's acceleration is applied by Durieu-Genin (1973) and Kronsjö-Dahlquist (1971).

The extrapolation technique based on the discretization error which is of order $O(|h|^p)$ is an extension of the technique suggested by Richardson-Gaunt (1927).

Iterative methods as applied to structural mechanics were reported by Boyle-Jennings (1973) and Tuff-Jennings (1973).
CHAPTER 5

EXAMPLES

5.1 Introduction

This chapter reports some results of the many problems solved by
the SER method. They include solutions of Laplace's equation with both
Dirichlet and Neumann boundary conditions in homogeneous and inhomogeneous
media. The application of the SER method to the solution of the Helmholtz
equation is discussed. A simple example illustrates the use of the iter-
ated extrapolation to scalar sequences, occurring in optimization.

Various scanning techniques pertinent to the solutions of field
problems are discussed and illustrated. Some programming remarks are
included. They refer to the SER method.

5.2 Laplace's Equation

Solutions to Laplace's equation subject to Dirichlet and Neumann
boundary conditions were obtained by the Gauss-Seidel method, SOR with
fixed \( \omega = 1.3 \), SOR with optimized \( \omega \) as described by Carré (1961) (optimi-
zed SOR), SER with variable \( \gamma \) and the Gauss-Seidel method as background,
and SER with variable \( \gamma \) and \( \omega \) on a CDC 6400 computer. The following error
criterion for terminating the iteration processes was applied to all the
methods

\[
|e^{(n)}_{\text{max}}| < \frac{\gamma^{(n)}(\ell \gamma^{(n)})}{1-\gamma^{(n)}}
\]

(5.1)
where
\[ \hat{\chi}(n) = \chi(n) - \chi(n-1), \]  
(5.2)
and
\[ |\delta_i^{(n)}| = \max_{\forall i} \delta_i^{(n)}, \quad i \in N, \]  
(5.3)

\[ \gamma(n) = \frac{|\delta_i^{(n)}|}{|\delta_i^{(n-1)}|}, \]

\[ = \frac{\sum_{\forall i} |\delta_i^{(n)}|}{\sum_{\forall i} |\delta_i^{(n-1)}|}, \]  
(5.4)

Various scanning techniques were used for each of these methods. Table 5.1 shows the number of iterations and the computing time required to reduce the maximum error given by (5.1) to both 1% and 0.1% for two problems. The first problem is shown in Fig. 5.1. It contains Dirichlet boundary conditions only. The other example is depicted in Fig. 5.2, and it has both Dirichlet and Neumann boundary conditions. Both regions shown in Figs 5.1 and 5.2 were discretized with 20 x 20 mesh points. It is seen that the extrapolation techniques require 66.6% fewer iterations and 60.6% less computing time with respect to the Gauss-Seidel method. The optimized SOR leads to 42.6% saving in number of iterations over the SOR (fixed \( \omega = 1.3 \)), however, due to the larger number of computational operations per iteration the computation time is only decreased by 41.1%. In each case the contour scan was the best scanning procedure. It is seen that SER with variable \( \gamma \) and \( \omega \) was superior to all presented relaxation techniques in both problems for 0.1% accuracy.
### TABLE 5.1

**COMPARISON BETWEEN VARIOUS FINITE DIFFERENCE METHODS**

<table>
<thead>
<tr>
<th>Method</th>
<th>Scan</th>
<th>Problem 1 (Fig. 5.1)</th>
<th>Problem 2 (Fig. 5.2)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1%</td>
<td>0.1%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Number of iterations</td>
<td>Time (sec)</td>
</tr>
<tr>
<td>Gauss-Seidel</td>
<td>Raster</td>
<td>100</td>
<td>6.2</td>
</tr>
<tr>
<td></td>
<td>Spiral</td>
<td>96</td>
<td>5.7</td>
</tr>
<tr>
<td></td>
<td>Contour</td>
<td>94</td>
<td>5.7</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SOR</td>
<td>Raster</td>
<td>54</td>
<td>3.5</td>
</tr>
<tr>
<td></td>
<td>Spiral</td>
<td>52</td>
<td>3.2</td>
</tr>
<tr>
<td></td>
<td>Contour</td>
<td>50</td>
<td>3.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Optimized SOR</td>
<td>Raster</td>
<td>31</td>
<td>2.1</td>
</tr>
<tr>
<td></td>
<td>Spiral</td>
<td>31</td>
<td>2.0</td>
</tr>
<tr>
<td></td>
<td>Contour</td>
<td>29</td>
<td>1.9</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SER</td>
<td>Raster</td>
<td>18</td>
<td>1.4</td>
</tr>
<tr>
<td></td>
<td>Spiral</td>
<td>20</td>
<td>1.5</td>
</tr>
<tr>
<td></td>
<td>Contour</td>
<td>18</td>
<td>1.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SER</td>
<td>Raster</td>
<td>20</td>
<td>1.6</td>
</tr>
<tr>
<td></td>
<td>Spiral</td>
<td>26</td>
<td>2.1</td>
</tr>
<tr>
<td></td>
<td>Contour</td>
<td>16</td>
<td>1.3</td>
</tr>
</tbody>
</table>

Note: The table compares the number of iterations and time taken for various methods to solve two problems. The data is presented for different scan settings and shows the efficiency of each method under these conditions.
Fig. 5.1. Dirichlet boundary value problem.

Fig. 5.2. Dirichlet and Neumann boundary value problem.
but the improvement over the optimized SOR, if at all, was marginal at 1% accuracy. This was caused by the initial convergence behaviour of the vector sequences generated by the background methods used in SER. Other examples are associated with the program for solving Laplace's equation [Kinsner-Della Torre (1973c)] and the contour scanning subprogram [Kinsner-Della Torre (1973b)] (see Fig. 5.6)

5.3 Helmholtz's Equation

Solutions to the Helmholtz equation

\[(\nabla^2 + k^2)x = 0\]  \hspace{1cm} (5.5)

will now be obtained by applying SER to the finite difference method as, reviewed by Beaubien-Wexler (1970, 1971). A lattice of points is defined over the space in which it is desired to find \(x\). Matrix \(\chi\) then has element values equal to the approximation to \(x\) at the lattice points. This matrix satisfies

\[(A - \lambda I) \chi = B\chi = 0\]  \hspace{1cm} (5.6)

where \(\lambda\) is the eigenvalue of the mode, \(I\) the unit matrix and \(A\) is the coefficient matrix related to the Laplacian operator. The matrix \(B\) is suitable for obtaining only the lowest mode of the Helmholtz equation since it is not positive definite for higher modes. It has been suggested [Beaubien - Wexler (1970, 1971)] to solve

\[B^TB\chi = C\chi = 0\]  \hspace{1cm} (5.7)

since \(C\) is positive definite and has the same eigenfunctions and eigenvalues as \(B\).
The iteration matrix \( M \) is defined as the matrix which takes the matrix \( \chi^{(n)} \) and generates \( \chi^{(n+1)} \). The \( M \) matrix is a function of the relaxation procedure, the eigenvalue and the coefficient matrix. For over-relaxation the iteration matrix will be denoted by \( M_{\omega, \lambda} \) where \( \omega \) is the over-relaxation factor and \( \lambda \) is the eigenvalue of \( A \) in (5.6).

The displacement vector is defined by

\[
\delta^{(n)} = \chi^{(n)} - \chi^{(n-1)}
\] (5.8)

A stationary iterative procedure generates successive displacement vectors or error vectors which satisfy the relationship

\[
\delta^{(n)} = M_{\omega, \lambda} \delta^{(n-1)} = (M_{\omega, \lambda})^{n-1} \delta^{(0)}
\] (5.9)
or

\[
\delta^{(n)} = (M_{\omega, \lambda})^{n-1} \delta^{(0)}
\] (5.10)

It can be shown that (5.9) leads to the expression

\[
\delta^{(n)} = \sum_{i=1}^{N} \xi_i^{n-1} \chi_i d_i
\] (5.11)

where \( \xi_i \) are the eigenvalues of \( M \), \( \chi_i \) the corresponding eigenvectors, \( d_i \) the scalar coefficients which are uniquely determined by the choice of the iteration parameters, and \( N \) the dimension of \( M \). It follows from (5.11) that, for a sufficiently large number of iterations

\[
\delta^{(n)} \approx \xi_1^{n-1} \chi_1 d_1 + \xi_2^{n-1} d_2 \chi_2
\] (5.12)

if one assumes that

\[
\xi_1 > \xi_2 > \xi_3 > \cdots > \xi_N
\] (5.13)

One can establish a similar relationship for the error vector. As \( \lambda \) approaches the exact value, \( \xi_1 \) must approach, but not exceed, unity for the
process to converge, and $d_1 \lambda_1$ must approach zero since the displacement
must be zero at the exact solution. Therefore, $d_1 = 0$ for exact $\lambda$.
However, since $\lambda$ is not exact, $d_1$ remains a small quantity near the
solution. It is seen from (5.11) that a stationary procedure converges
if all the eigenvalues of $M$ lie within the unit circle. This require-
ment is satisfied by the SOR method with $0 < \omega < 2$ which is used in SER.

The object of the extrapolation is to produce an extrapolated
vector $\chi^{(n)}$ which is closer to the solution than a vector $\chi^{(n)}$ resulting
from SOR. The three successive vectors, $\chi^{(n-2)}$, $\chi^{(n-1)}$, and $\chi^{(n)}$ are
obtained by the Jacobi, Gauss-Seidel or SOR methods. Then the extra-
polation is applied simultaneously to obtain the vector $\chi^{(n)}$. The extra-
polation is performed every three iterations, and if carried out less
frequently, the method is less efficient.

In practice, the optimum value of $\omega$ is not known. One dimensional
search has been used by Beauchien and Wexler (1970, 1971) for the optimum
value of the acceleration factor. More efficient methods of optimization
have been presented by Carré (1961), Kulrude (1961) Reid (1966).

From (5.11), for large $n$ the ratio of two successive displacement
vector norms is

$$
\frac{\| \chi^{(n)} \|}{\| \chi^{(n-1)} \|} \approx \frac{d_1 \xi_1^{n-1} x_1 + d_2 \xi_2^{n-1} x_2}{d_1 \xi_1^{n-2} x_1 + d_2 \xi_2^{n-2} x_2}
$$

(5.14)

Various norms of $\chi^{(n)}$ have been discussed [Carré (1961) and Kincaid (1972)].
Eq. (5.14) may be minimized with respect to the magnitude of the sub-
dominant eigenvalue, $\xi_2$, and other eigenvalues. An elegant method of
optimization is given by Carré (1961). This method requires the
system matrix to possess Young's Property A. The five-point finite
difference operator generates the matrix with the Property A. However,
this operator can be used only for the dominant mode in solving Helmholtz
equation since it is not positive definite for higher modes. The thirteen-
point operator does not possess this property. Therefore C'rre's optimi-
ization method is not applicable. Tee [Tee (1963)] derived two seventeen-
point operators which do possess Property A. In addition, Tee's second
operator [Silvester (1970)] possesses improved diagonal dominance, a
factor which accelerates convergence to the solution. However, the 17-
point operator causes more problems in logic for internal mesh points in
the vicinity of boundary conditions, especially Neumann boundary conditions.
The $M_{\omega,\lambda}$ matrix tends to be unsymmetric if the ratio of Neumann boundary
points to interior points is large. The eigenvalues $\xi_i$ of $M$ become
complex, and although the condition for convergence is not violated, the
optimization process, even the one-dimensional search, may be unreliable
[Reid (1971)]. For further discussion see Chapter 2.

Since

$$
\lim_{n \to \infty} \frac{|\xi^{(n)}|}{|\xi^{(n-1)}|} = |\xi_{sd}|
$$

(5.15)

where $\xi_{sd}$ is the subdominant eigenvalue of $M$, hence $\xi_{sd}$ can be estimated
at any iteration by the use of (5.4), yielding the variable $\gamma$ in the
SER method. Note that the variable $\omega$ is optimized at the same time.

The iteration was terminated by Collatz's criterion (Eq. 5.1). The
current value of $\gamma^{(n)}$ was used in (5.1) until a final estimate of
the overrelaxation parameter $\omega_{opt}$ was found. From that moment a final
estimate of $\gamma^{(n)}$ was computed from
\[ \gamma(n) = \omega_{\text{opt}} - 1. \quad (5.16) \]

If the thirteen-point operator is used, \( \gamma \) cannot be optimized by (5.4). In this case, one may estimate \( \gamma \) using a suboptimal accelerating factor, \( \omega_{\text{sopt}} \), resulting from a one-dimensional search. Reasonable estimates to the optimum value of \( \gamma \) are given by

\[ \gamma(n) = \omega(n)_{\text{sopt}} - 1 \quad (5.17) \]

It is noted that if mesh halving is applied during the relaxation process, \( \gamma(n) \) determined by (5.17) is overestimated for smaller mesh lengths. The problem arises from the fact that a mesh subdivision is applied when the error has been reduced by considerable amount and the vector \( \mathbf{X}^{(n)} \) is close to the solution. A new one-dimensional search for \( \omega_{\text{sopt}} \) starts from a new initial value of \( \omega \), usually larger than \( \omega_{\text{sopt}} \). Therefore, before \( \omega \) can stabilize, (5.17) gives too large a value of \( \gamma(n) \) and hence the application of SER results in overestimation of the extrapolated vectors. The problem may be overcome by

\[ \gamma(n) = \frac{\omega(n)_{\text{sopt}} - 1}{n} \quad (5.18) \]

where \( m = 2^5 \), and \( s \) is the number of mesh subdivisions performed.

The dominant mode for the waveguide whose cross-section is shown in Fig. 5.3 was solved on a CDC 6400. It took 100 iterations and 12.3 seconds to obtain the solution illustrated in Fig. 5.4. A five point formula was used for this process. The error criterion in this case was a residual ratio of \( 10^{-4} \).

Another problem solved by SER was obtaining the dominant and higher modes in a circular waveguide where the background procedure was
Fig. 5.3. Boundary conditions for the dominant TM mode in a ridge waveguide. Only one half of the symmetrical guide is shown.
Fig. 5.4. The dominant TM mode in a symmetrical ridge waveguide shown in Fig. 5.3.
Fig. 5.5. Placement of boundary data for a circular quadrant of radius 3.0. Characteristic dimension $a = 3.0$. The TE$_{31}$ mode with $k_v a = 4.1774$ generated by the boundary conditions is indicated with the dashed lines.
the PDSOR as implemented by Beaubien-Wexler (1971b) in their computer program. The SER algorithm was combined with the PDSOR algorithm. A quarter of a circular waveguide, shown in Fig. 5.5, has been discretized with 81 x 81 mesh points. A significant improvement resulted even though no attempts were made to optimize the PDSOR program for SER. For example, the 13-point operator, the one-dimensional search technique for optimizing \( \omega \), and the enormous storage allocation required by the PDSOR program were kept unchanged. The combined algorithm produced the same field values for each mode as those obtained by the PDSOR program alone. The computed parameters such as cutoff wavenumbers, mode impedances, and attenuation coefficients were the same. The extrapolation was applied only on two subdivisions because smaller mesh sizes corresponded to vectors which were very close to the solutions. The application of the SER method reduced computation time (approx. 5 to 6 minutes per mode) by 5 to 10%.

A discussion of the application of SER to the solution of Helmholtz's equation shows that no new conditions need be imposed on the equation.

5.4 Inhomogeneous Magnetic Media

The SER method was essentially developed and motivated by the need for improving the rate of convergence of the solutions of three-dimensional inhomogeneous magnetic field problems such as presented in this section. The method was applied to obtain the magnetic field within a magnetic cylindrical domain (bubble) supporting crystal due to magnetic poles which are formed in "T-bar", "Y-bar", and "Chevron"
bubble propagating circuits. These circuits, their dimensions, and sections of the basic two-dimensional periods, where the field was computed, are shown in Figs. 5.6 to 5.8. The Permalloy permeability is assumed to be 1000.

Solutions were obtained in two and a half minutes of which 37 seconds were required for the actual iterative solution to 0.01% accuracy of 17 x 37 x 10 mesh points. A CDC 6400 computer was used throughout. The resulting average magnetic field intensity (normalized to $4\pi M_0$) is shown in Figs. 5.9 to 5.11. The sequences (arranged as a page of English text) refer to transverse field rotation by $180^\circ$, every $30^\circ$ starting from $0^\circ$. It is seen that the minimum of the field follows the rotating field and hence a bubble must follow the propagating energy minimum. This preliminary analysis of the field distribution within crystals (such as orthoferrites or garnets) can be extended to obtain the high speed motion of a bubble for an arbitrarily shaped field access device and also to computer-aided design of optimum propagating structures.

Further discussion on the modelling and computation can be found in Kinsner-Della Torre (1972, 1974c)

5.5 Iterated Extrapolation

In certain applications of a uniform magnetic field a system of rectangular coils has structural advantages over the more usual circular and square Helmholtz coils, especially for a very large but flat volume of interest. The Helmholtz spacing for the limiting cases, such as the circular, square coils and a system of parallel wires, is known. An analytical solution for a rectangular coil system is not feasible. This
Fig. 5.6. T-bar two-dimensional bubble propagating Permalloy circuit.
Fig. 5.7. Y-bar structure.
Fig. 5.8. Chevron circuit.
Fig. 5.9. Sequence of computer plots showing normalized z-averaged z-component of magnetic field intensity in bubble crystals due to T-bar propagating structure as transverse field rotates counterclockwise through 180°.
Fig. 5.10. Same as Fig. 5.9 but for Y-bar propagating structure.
Fig. 5.11. Same as Fig. 5.9 but for Chevron propagating structure.
solution has been obtained [Kinsner (1974)] by means of the Davies, Swann and Campey method [Bandler (1971)] combined with the quadratic interpolation method and the iterated extrapolation discussed in Sec. 3.8. The quadratic interpolation technique was applied iteratively, producing a sequence of estimates to the optimum value. Although the convergence of a repeated quadratic interpolation is guaranteed its rate is very small. Since the function was very badly behaved, the sequence was oscillating. The application of a single nonlinear transformation (e.g., Aitken's formula) to obtain the limit of the sequence was not sufficient. Its repeated (iterated) use led to the solution in 10 times less iterations than the original sequence. The attained accuracy was $10^{-14}$. Triple iterations of Aitken's formula were used.

5.6 Scanning Procedures

Various scanning techniques were discussed in Sec. 2.5.1.7. The results reported in this section indicate the effect of a scanning on the number of iterations needed for a solution.

A comparison of the number of iterations for Laplace's equation with boundary conditions indicated in Fig. 5.12 (Problem 1) and Fig. 5.13 (Problem 2) solved by the Gauss-Seidel method and the SER method, applying raster, spiral and contour scans, is shown in Table 5.2. Dimensions of the arrays were 20 x 20 mesh points and the initial values of the points within the boundary were constant ($V=0$). In Problem 1 the spiral and contour scans generate the same orderings, and hence the number of iterations required to reduce the maximum error to 0.1% is the same. The relative reduction of the number of iterations produced by the SER
Fig. 5.12. Sample of boundary conditions and spiral scan (Problem 1).
Fig. 5.13. Sample of boundary conditions and contour scan (Problem 2).
### TABLE 5.2
NUMBER OF ITERATIONS REQUIRED TO REDUCE THE MAXIMUM ERROR TO 0.1% FOR VARIOUS SCANS

<table>
<thead>
<tr>
<th>Scanning techniques</th>
<th>Problem 1</th>
<th></th>
<th>Problem 2</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Gauss-Seidel</td>
<td>SER</td>
<td>Gauss-Seidel</td>
<td>SER</td>
</tr>
<tr>
<td>Raster scan</td>
<td>120</td>
<td>38</td>
<td>144</td>
<td>46</td>
</tr>
<tr>
<td>Spiral scan</td>
<td>110</td>
<td>16</td>
<td>100</td>
<td>40</td>
</tr>
<tr>
<td>Contour scan</td>
<td>116</td>
<td>16</td>
<td>138</td>
<td>36</td>
</tr>
</tbody>
</table>
method is greater than the reduction produced by the Gauss-Seidel method because the former takes the advantage of the faster propagation of the effect of the boundary conditions. The programs have been tested on a CDC 6400 computer. These tests indicate that both scans are superior to the raster scan and other consistent orderings such as the diagonal scan or the "white-black" scan, and, in general, the contour scan is the best one. This conclusion was further justified when the different scans were used in the SOR method with \( \omega \) optimized by techniques of Carré (1961), Kulsrud (1961), and Reid (1966).

5.7 **Norm Computation**

Experimental study was carried to determine the computational reliability of the factor \( \gamma \) used in the SER and SOR methods. Fig. 5.14 shows the \( \gamma \) variation as a function of the iteration count for two boundary value problems, illustrated in Figs. 2.12 (Problem 1) and 2.13 (Problem 2), which were solved by the Gauss-Seidel method. It is seen from Fig. 5.14 that both values of \( \gamma \) asymptotically approach constant values after some number of iterations. The SOR method with the over-relaxation factor \( \omega \) optimized by a suitable technique [Carré (1961)] effectively terminates the \( \gamma \) computation when the optimum \( \omega \) is found (the optimum \( \omega \) can be found if and only if the \( \gamma \) approaches a steady value). A similar termination process takes place when the SER method uses SOR, as the background iteration.

If, however, the background iteration does not use any technique to optimize its own parameters, then the \( \gamma \) must be terminated by the following procedure:
Fig. 5.14. Nearly-geometric convergence of $\gamma$. 
Fig. 5.15. Nearly-geometric convergence of errors $|e_Y|$ and $|e_{TY}|$. 
\[ |e^{(n)}_\gamma| = \left|\frac{\gamma^{(n)} - \gamma^{(n-1)}}{1 - \gamma^{(n)}}\right| < 0.05 \]  \hfill (5.19)

where \( \gamma^{(n)} \) is given by Eq. (5.4). The bounding value 1/20 in (5.19) appeared to be adequate for many problems. The value of \( \gamma \) has fairly steady behaviour when that bound is reached. Furthermore, formula (5.19) is somewhat better than the ordinary relative error

\[ |e^{(n)}_{\gamma\gamma}| = \left|\frac{\gamma^{(n)} - \gamma^{(n-1)}}{\gamma^{(n)}}\right| < 0.005 \]  \hfill (5.20)

The main reason for the termination of \( \gamma \) computation is not even the computer time saving but the fact that when the \( \gamma \) approaches some steady value, then the displacement vectors become small and the round-off error effect is greater.

The above experiment leads to another important conclusion, pertinent to the convergence analysis of iterative finite difference solutions of field problems (Sec. 3.4 and Sec. 3.5.2). Examination of Fig. 5.15, which is a semi-logarithmic plot of \( |e_\gamma| \) and \( |e_{\gamma\gamma}| \) versus the iteration count, reveals almost linear variation above some number of iterations. It implies almost geometric convergence of the processes.

5.8 Programming Remarks

The SER method requires only one extra array for the vector \( \chi^{(n-2)} \). This substantial saving of one array for \( \chi^{(n-1)} \) is achieved by the following procedure: The extrapolation is performed during the \( n \)th iteration and not after the \( n \)th iteration as implied by the simultaneous
technique which must be used in the extrapolation. The background iteration uses the neighbours of \( x_i^{(n-1)} \) and produces a new value of \( x_i^{(n)} \).

Thus, if one temporarily saves the \( x_i^{(n-1)} \), then all three required values \( x_i^{(n-1)} \), \( x_i^{(n)} \), and \( x_i^{(n-2)} \) from the extra array are available at a time.

The extrapolation takes place at this moment and the extrapolated value is stored in the extra array (and not in the working array), destroying the old value \( x_i^{(n-2)} \).

Checking the sign of \( \delta_i^{(n)} \delta_i^{(n-1)} \) is time consuming. It can be solved by the arithmetical IF statements, without any multiplication. Another solution can be obtained using the logical IF statement with logical variables.

If a problem requires a large storage, all the tags specifying the relative position of a point and its type (ordinary, Dirichlet, or Neumann) should be packed into one computer word (or set of words) together with the value of \( x_i^{(n)} \). This can be accomplished by a small subroutine written in assembly language or an extended version of Fortran, if available.

5.9 Conclusions

The examples presented in this Chapter do not cover all the topics discussed in previous Chapters. They do show, however, that the SER method and its extensions are useful in practical applications, are reliable, and do not require essential changes in the existing programs to be successfully incorporated into them. The method serves as an accelerator of the convergence rate of some iterative processes which exhibit geometric, nearly-geometric, or linear convergence.
CHAPTER 6

CONCLUSIONS

Various physical field problems, their analytical and numerical solutions have been reviewed and discussed. Basic concepts and definitions used explicitly or implicitly in this thesis have been presented in Chapters 1 and 2. The discretization error, the round-off error, and the residual error in numerical methods have been distinctly singled out.

A detailed finite difference formulation has been presented in order to obtain the scope of difficulties and advantages when the method is used in solving field problems. The discussion about various direct and iterative methods has revealed the apparent need and possibility of accelerating the convergence of vector sequences.

Some numerical results have indicated that Young's conclusions of the theory of consistent orderings cannot be freely adapted to the analysis of the vector sequence convergence acceleration because, in general, the consistent orderings do not yield the fastest convergence when applied within iterative methods. As a result of this observation, new nearly-consistent but faster scanning procedures have been implemented.

As a byproduct of the finite difference formulation, the five-point operator has been generalized to linear and nonlinear continuous inhomogeneous media. The main Chapter 3 has been devoted to various methods and techniques of accelerating scalar and vector sequences.
It has introduced a family of extrapolation methods applicable to deterministic and related stochastic sequences. An analysis and numerical results have indicated that the successive extrapolated relaxation (SER) method with variable coefficients is absolutely stable, and yields considerable convergence improvement even in the presence of numerical noise. The method is particularly suitable for acceleration of very slowly convergent sequences where other techniques fail. An explicit and a general error analyses have determined the sensitivity of SER with respect to the errors associated with particular elements of a sequence or with the generalized perturbation matrix of an iterative process generating the sequence. Conditioning of the background iteration has been included in the determination of the sensitivity.

Modifications of the basic SER method have led to schemes with ultimate frequency and weighted extrapolation factors, depending upon the type of convergence of the background iterative scheme and the need to avoid singularities. An error spectrum analysis has shown the improvement in the total convergence rate for symmetric iteration matrices with positive eigenvalues. These schemes are applicable to iterations with symmetric and nonsymmetric matrices.

Scalar and vector sequences may be treated as mathematical transients. This concept has readily led to higher order nonlinear transformations of the Aitken transform type. The transforms can be applied to successive elements of a sequence, similarly to SER, or repeatedly (iteratively) to some number of the elements determined by the order of the transform. Other formulae of the Lyusternik transform type have been presented. They are particularly applicable for very slowly con-
vergent sequences.

Finite element solutions to field problems should be obtained by the higher-order methods and the direct techniques for solving large sets of equations. For small problems, the first-order finite element method with iterative techniques cannot compete with the other approach but it may facilitate solutions to large problems because no matrices and essentially only the solution vector(s) are stored. A method which combines both approaches has been presented in Chapter 4. In order to obtain higher convergence rates of the iterative methods, employed in the finite element solutions, various acceleration techniques should be used. Such techniques which cause faster diminishing of both the residual and discretization errors have been presented.

An error analysis has resulted in explicit expressions for the upper bound of the discretization error as a function of the maximum side and the largest angle of all triangular elements. These expressions can be obtained by other methods of analysis; the method presented in the thesis is probably the simplest. The result leads to an optimal extrapolation technique for estimating the exact solution with finite size elements.

This work has left some problems for future investigation:

a) Both the original and improved SER methods can be further optimized with respect to oscillating vector sequences or to sequences generated by iterative processes with nonsymmetric matrices.

b) Although the higher-order extrapolation techniques can be applied to scalar and vector sequences, they are not economical when used repeatedly in iterative processes. To remove this inconvenience,
optimized weighted extrapolation factors ought to be found. Such factors may be based on the vector norm estimates of the eigenvalues of the iteration matrix, similarly to those employed in Chapter 3.

c) Is it possible and, if so, advantageous to apply different transforms to the same vector sequence simultaneously, depending upon the local convergence (or divergence) characteristics of the sequence?

d) Extrapolation techniques, similar to those presented in Chapters 3 and 4, can be applied to various optimization algorithms. A related work is being currently carried by Bandler (1973); a suggestion of such a possibility of extrapolation can be found in Charalambous' (1973) thesis. A simple example of the applicability of the idea has been presented in Chapter 5.

e) The combined finite element-finite difference method. When should the switching from one method to the other occur? What are the best direct and iterative methods to be used? What are the theoretical and practical implications of using the combined method? What is the actual advantage of using this method over the purely finite element or finite difference solutions? What are the optimum convergence accelerating techniques for the combined method?

f) The general n-dimensional finite difference operator (Sec. 2.2.9) can be applied not only to linear but also to nonlinear inhomogeneous media. This operator and appropriate acceleration techniques could be used in solving the real dynamic bubble propagation problem.

It might be useful to point out some of the guiding thoughts and contributions in this work:
(i) An explicit error analysis of Aitken's transform.

(ii) A sensitivity analysis of perturbed iteration matrices.

(iii) New extrapolation methods.

(iv) A simple discretization error analysis of the first order finite element methods.

(v) New nearly-consistent scanning procedures.

(vi) A general $n$-dimensional finite difference operator for continuous or piecewise continuous inhomogeneous static magnetic or dielectric media.
BIBLIOGRAPHY

A


Microwave Theory Tech., vol. MTT-17, pp. 294-301, June.


D


D.J. Evans (1964b). "The extrapolated modified Aitken iteration method


W. van Koppenfels and F. Stallman (1959), Praxis der Konformen
Abbildung, Berlin: Springer-Verlag.


J. G. Kretzschmar (1971). "Field configuration of the TM-c01 mode in an


L. Kronsjo and G. Dahlquist (1971). "On the design of nested iterations


H. E. Kulsrud (1961). "A practical technique for the determination of
the optimum relaxation factor of the successive over-relaxation


T. Larsen (1972). "On the relation between modes in rectangular, elliptical
and parabolic waveguides and a mode classifying system", IEEE Trans. Microwave Theory Tech., vol. MTT-20, pp. 379-84.

B. L. Lennartsson (1972). "A network analogue method for computing the

T. Lloyd and H. McCallion (1968). "Bounds for the optimum over-relaxation

M. S. Lynn (1964). "On the equivalence of SOR, SSOR and USSOR as applied

L. A. Lyusternik (1947). "Notes on the numerical solution of boundary
value problems (Laplace's equation) by the mesh method", Trudy
mat. Inst. V.A. Steklova, vol.20, pp. 49-64.


IEEE Trans. Microwave Theory Tech. (Corresp.), vol. MTT-17, pp. 51-2, January.


W.


Y.


Z.


### APPENDIX 1

**SOME SECOND ORDER PDE WITH TWO INDEPENDENT VARIABLES**

<table>
<thead>
<tr>
<th>Name</th>
<th>Equation</th>
<th>Application</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laplace</td>
<td>$u_{xx} + u_{yy} = 0$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Poisson</td>
<td>$u_{xx} + u_{yy} = f(x,y)$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Helmholtz</td>
<td>$u_{xx} + u_{yy} + k^2 u = f(x,y)$</td>
<td>Physics, Engineering, etc.</td>
<td>Elliptic</td>
</tr>
<tr>
<td>Schrödinger¹</td>
<td>$u_{xx} + u_{yy}$</td>
<td></td>
<td>Elliptic</td>
</tr>
<tr>
<td></td>
<td>$+ 2m n^{-2} [E - V(x,y)] u = 0$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Diffusion²</td>
<td>$u_{xx} = h^{-2} u_t$</td>
<td>Heat conduction</td>
<td>Parabolic</td>
</tr>
<tr>
<td>Diffusion³</td>
<td>$u_{xx} = h^{-2} u_t - k^{-1} p$</td>
<td>Heat conduction with sources</td>
<td></td>
</tr>
<tr>
<td>Wave</td>
<td>$u_{xx} = c^{-2} u_{tt}$</td>
<td>Vibration of string</td>
<td>Hyperbolic</td>
</tr>
<tr>
<td>Wave</td>
<td>$u_{xx} = c^{-2} u_{tt} + a u_t$</td>
<td>Vibration with friction</td>
<td></td>
</tr>
<tr>
<td>Wave⁴</td>
<td>$u_{xx} = c^{-2} u_{tt} + a u_t + b u$</td>
<td>Transmission line</td>
<td></td>
</tr>
<tr>
<td>Tricomi⁵</td>
<td>$u_{xx} + y^{-1} u_{yy} = 0$</td>
<td>Transonic flow</td>
<td></td>
</tr>
</tbody>
</table>

¹ Morse-Feshbach (1953) p.243  
² Moon-Spencer (1961) p.412  
³ Ibid. p.427  
⁴ Ibid. p.439  
⁵ Garabedian (1964) p.422
### APPENDIX 2

**FINITE DIFFERENCE APPROXIMATIONS ON VARIOUS MESHES**

<table>
<thead>
<tr>
<th>No.</th>
<th>Differential equation and mesh</th>
<th>Difference operator</th>
<th>Difference equation</th>
<th>Local truncation error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Laplace ( u_{xx} + u_{yy} = 0 ) 5-point rectangular</td>
<td><img src="image" alt="Difference operator diagram" /></td>
<td>[ u_0 = \frac{1}{2(h_1^2+h_2^2)} \left[ h_2^2(u_1+u_2) + h_1^2(u_3+u_4) \right] ]</td>
<td>[ u_0 = \frac{1}{20} \left( u_5^{(n-1)} + u_6^{(n-1)} + u_7^{(n)} + u_8^{(n)} \right) ]</td>
</tr>
<tr>
<td>2</td>
<td>Laplace ( u_{xx} + u_{yy} = 0 ) 9-point rectangular</td>
<td><img src="image" alt="Difference operator diagram" /></td>
<td>[ u_0 = \frac{1}{20} \left( u_5^{(n-1)} + u_6^{(n-1)} + u_7^{(n)} + u_8^{(n)} \right) ]</td>
<td>[ u_0 = \frac{1}{20} \left( u_5^{(n-1)} + u_6^{(n-1)} + u_7^{(n)} + u_8^{(n)} \right) ]</td>
</tr>
<tr>
<td>3</td>
<td>Poisson ( u_{xx} + u_{yy} = f(x,y) ) 5-point square</td>
<td><img src="image" alt="Difference operator diagram" /></td>
<td>[ \sum_{i=1}^{4} u_i - 4u_0 = h^2f_0 ]</td>
<td>[ \frac{4h^2}{6} M_4 ]</td>
</tr>
<tr>
<td>No.</td>
<td>Differential equation and mesh</td>
<td>Difference operator</td>
<td>Difference equation</td>
<td>Local truncation error</td>
</tr>
<tr>
<td>-----</td>
<td>---------------------------------</td>
<td>---------------------</td>
<td>---------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>4</td>
<td>5-point square</td>
<td><img src="image" alt="5-point grid" /></td>
<td>$\sum_{i=1}^{4} u_i - 4u_0 = 2h^2 f_0$</td>
<td>$\frac{4h^2}{3} M_4$</td>
</tr>
<tr>
<td>5</td>
<td>9-point square</td>
<td><img src="image" alt="9-point grid" /></td>
<td>$\frac{1}{6} [4 \sum_{i=1}^{4} u_i + \sum_{i=5}^{8} u_i - 20u_0] = h^2 f_0 + \frac{h^4}{12} (\nabla^2 f)_0$</td>
<td>$\frac{520h^6}{3 \cdot 8!} M_8$</td>
</tr>
<tr>
<td>6</td>
<td>7-point triangular (equilateral)</td>
<td><img src="image" alt="7-point grid" /></td>
<td>$4 \sum_{i=1}^{4} u_i + 8u_0 = \frac{h^2}{2} \left( \sum_{i=1}^{4} f_i + 8f_0 \right)$</td>
<td>$0[h^2]$</td>
</tr>
<tr>
<td>7</td>
<td>13-point triangular</td>
<td><img src="image" alt="13-point grid" /></td>
<td>$\sum_{i=1}^{6} u_i - 6u_0 = \frac{3h^2}{2} f_0 + \frac{3h^4}{32} (\nabla^2 f)_0$</td>
<td>$0[h^4]$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>$\sum_{i=1}^{6} u_i - 6u_0 = \frac{3h^2}{2} f_0$</td>
<td>$0[h^2]$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>$\sum_{i=1}^{12} u_i - 48u_0 = 9h^2 f_0$</td>
<td>$0[h^4]$</td>
</tr>
<tr>
<td>No.</td>
<td>Differential equation and mesh</td>
<td>Difference operator</td>
<td>Difference equation</td>
<td>Local truncation error</td>
</tr>
<tr>
<td>-----</td>
<td>--------------------------------</td>
<td>---------------------</td>
<td>---------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>8</td>
<td>4-point regular hexagons</td>
<td><img src="image" alt="Hexagon Diagram" /></td>
<td>[ \frac{3}{4} \sum_{i=1}^{4} u_i - 3u_0 = \frac{3h^2}{4} f_0 ]</td>
<td>(0[h])</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>No.</th>
<th>Differential equation and mesh</th>
<th>Difference operator</th>
<th>Difference equation</th>
<th>Local truncation error</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>Fourth order</td>
<td><img src="image" alt="Square Diagram" /></td>
<td>(-8 \sum_{i=1}^{4} u_i + 8 \sum_{i=5}^{8} u_i + 12 u_9 + 20u_0 = h^4 f_0)</td>
<td>(\frac{10h^2}{9} M^6)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>No.</th>
<th>Differential equation and mesh</th>
<th>Difference operator</th>
<th>Difference equation</th>
<th>Local truncation error</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Helmholtz (v^2 u + k^2 u = 0)</td>
<td><img src="image" alt="Square Diagram" /></td>
<td>(-4 \sum_{i=1}^{4} u_i + (4-\lambda)u_0 = 0) where (\lambda = (hk)^2)</td>
<td>(0[h^2])</td>
</tr>
<tr>
<td>No.</td>
<td>Differential equation and mask</td>
<td>Difference operator</td>
<td>Difference equation</td>
<td>Local truncation error</td>
</tr>
<tr>
<td>-----</td>
<td>--------------------------------</td>
<td>---------------------</td>
<td>---------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>11</td>
<td>13-point(^3) square</td>
<td>See No. 10</td>
<td>[ \begin{align*} \frac{4}{5} u_i + 2 \sum_{i=1}^{5} u_i + \sum_{i=9}^{12} u_i + tu_0 &amp;= 0 \ s &amp;= 2m - 8 \ t &amp;= m^2 - 8m + 20 \ \pi &amp;= (hk_{n-1})^2 \end{align*} ]</td>
<td>(0[h^2])</td>
</tr>
<tr>
<td>12</td>
<td>17-point(^9) square</td>
<td></td>
<td>[ \begin{align*} \frac{1}{108} \left[ \sum_{i=1}^{4} u_i + 11 \sum_{i=5}^{8} u_i + 16 \sum_{i=9}^{16} u_i + tu_0 \right] &amp;= 0 \ s &amp;= 216m - 177 \ t &amp;= 108m^2 - 864m + 640 \ \pi &amp;= (hk_{n-1})^2 \end{align*} ]</td>
<td>(0[h^2])</td>
</tr>
</tbody>
</table>

2. Ibid. p.442.
3. Ibid. p.444.
5. Ibid. p.448.
6. Ibid. p.450.
APPENDIX 3

MOLECULES FOR THE OPERATORS $v^2$, $v^4$, AND $(v^2 + k^2)$

<table>
<thead>
<tr>
<th>No.</th>
<th>Differential operator</th>
<th>Difference operator</th>
<th>Discretization error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$h^2 v^2 u = \frac{2}{s_2} a_{2h} + \frac{2}{s_3} a_{3h} + \frac{2}{s_4} a_{4h}$</td>
<td>$s_1 = a_1 (a_1 + a_3)$, $s_2 = a_2 (a_2 + a_4)$, $s_3 = a_3 (a_3 + a_1)$, $s_4 = a_4 (a_4 + a_2)$</td>
<td>$O[h]$</td>
</tr>
<tr>
<td>2</td>
<td>$h^2 v^2 u =$</td>
<td></td>
<td>$O[h^2]$</td>
</tr>
<tr>
<td>3</td>
<td>$2h^2 v^2 u =$</td>
<td></td>
<td>$O[h^2]$</td>
</tr>
</tbody>
</table>

Prop. A

Prop. A
<table>
<thead>
<tr>
<th>No.</th>
<th>Differential operator</th>
<th>Difference operator</th>
<th>Discretization error</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>$12h^2v^2u = $</td>
<td>![Diagram 1]</td>
<td>$O(h^4)$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>![Diagram 2]</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>$72h^2v^2u = $</td>
<td>![Diagram 3]</td>
<td>$O(h^4)$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>![Diagram 4]</td>
<td>Prop. A</td>
</tr>
<tr>
<td>No.</td>
<td>Differential operator</td>
<td>Difference operator</td>
<td>Discretization error</td>
</tr>
<tr>
<td>-----</td>
<td>----------------------</td>
<td>---------------------</td>
<td>---------------------</td>
</tr>
<tr>
<td>6</td>
<td>$6h^2v^2u = $</td>
<td><img src="#" alt="Diagram 1" /></td>
<td>$O[h^6]$</td>
</tr>
<tr>
<td>7</td>
<td>$\frac{3}{2}h^2v^2u = $</td>
<td><img src="#" alt="Diagram 2" /></td>
<td>$O[h^6]$</td>
</tr>
<tr>
<td>8</td>
<td>$9h^2v^2u = $</td>
<td><img src="#" alt="Diagram 3" /></td>
<td>$O[h^6]$</td>
</tr>
<tr>
<td>9</td>
<td>$\frac{3}{4}h^2v^2u$</td>
<td><img src="#" alt="Diagram 4" /></td>
<td>$O[h]$</td>
</tr>
</tbody>
</table>

Prop. A
<table>
<thead>
<tr>
<th>No.</th>
<th>Differential operator</th>
<th>Difference operator</th>
<th>Discretization error</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>$h^4 \nabla^4 u =$</td>
<td><img src="image1.png" alt="Diagram 10" /></td>
<td>$0[h^2]$</td>
</tr>
<tr>
<td>11</td>
<td>$6h^4 \nabla^4 u =$</td>
<td><img src="image2.png" alt="Diagram 11" /></td>
<td>$0[h^4]$</td>
</tr>
</tbody>
</table>
### Appendix 3 (continued)

<table>
<thead>
<tr>
<th>No.</th>
<th>Differential operator</th>
<th>Difference operator</th>
<th>Discretization error</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>$12h^4v^4 u = \ldots$</td>
<td><img src="image1" alt="Diagram" /></td>
<td>$O(h^2)$ Prop. A</td>
</tr>
<tr>
<td>13</td>
<td>$108h^4v^4 u = \ldots$</td>
<td><img src="image2" alt="Diagram" /></td>
<td>$O(h^2)$ Prop. A</td>
</tr>
</tbody>
</table>
Appendix 3 (continued)

<table>
<thead>
<tr>
<th>No.</th>
<th>Differential operator</th>
<th>Difference operator</th>
<th>Discretization error</th>
</tr>
</thead>
</table>
| 14  | \( \frac{9}{16} h^4 v^4 u = \) | \[ \begin{array}{c}
 1 \\
 3 \\
 1 \\
 1 \\
 1 \\
 1 \\
 1 \\
 1 \\
 1 \\
 1 \\
 \end{array} \]
  + 0[h^6] | 0[h^2] |
| 15  | \((v^2 + k^2)\) | \[ \begin{array}{c}
 1 \\
 t \\
 1 \\
 1 \\
 1 \\
 1 \\
 1 \\
 1 \\
 1 \\
 1 \\
 \end{array} \]
  \( t = 4 - m \) |
  \( m = (hk)^2 \) |
  Prop. A |
| 16  | \((v^2 + k^2)\) | \[ \begin{array}{c}
 1 \\
 2 \\
 1 \\
 2 \\
 1 \\
 1 \\
 \end{array} \]
  \( s = 2m - 8 \) |
  \( t = m^2 - 8m + 20 \) |
  \( m = (hk_{n-1})^2 \) |
  0[h^2] |
<table>
<thead>
<tr>
<th>No.</th>
<th>Differential operator</th>
<th>Difference operator</th>
<th>Discretization error</th>
</tr>
</thead>
<tbody>
<tr>
<td>17</td>
<td>$(v^2 + k^2)$</td>
<td>![Diagram 1]</td>
<td>$0[h^4]$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$r = -2m + 28$</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>$s = 32m - 154$</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>$t = 12m^2 - 120m + 368$</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>$m = (hk_{n-1})^2$</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>$(v^2 + k^2)$</td>
<td>![Diagram 2]</td>
<td>$0[h^2]$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$s = 24m - 39$</td>
<td>Prop. A</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$t = 12m^2 - 36m + 128$</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>$m = (hk_{n-1})^2$</td>
<td></td>
</tr>
<tr>
<td>No.</td>
<td>Differential operator</td>
<td>Difference operator</td>
<td>Discretization error</td>
</tr>
<tr>
<td>-----</td>
<td>-----------------------</td>
<td>---------------------</td>
<td>---------------------</td>
</tr>
<tr>
<td>19</td>
<td>((\sigma^2 + k^2))</td>
<td><img src="image" alt="Diagram" /></td>
<td>(O[h^2]) Prop. A</td>
</tr>
</tbody>
</table>

\[
s = 216m - 177 \\
t = 108m^2 - 864m + 640 \\
m = (hk_{n-1})^2
\]
### APPENDIX 4

**COMPARISON OF THE NUMBER OF ARITHMETIC OPERATIONS IN VARIOUS METHODS**

<table>
<thead>
<tr>
<th>Method</th>
<th>$Ax = b$</th>
<th>$A^{-1}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\div$</td>
<td>$\times$</td>
</tr>
<tr>
<td>Direct Gaussian Elimination</td>
<td>$\frac{1}{3} n^3 + \frac{2}{3} n - \frac{1}{3} n$</td>
<td>$\frac{1}{3} n^3 + \frac{2}{3} n - \frac{5}{6} n$</td>
</tr>
<tr>
<td>Jordan</td>
<td>$\frac{1}{2} n^3 + \frac{1}{2} n$</td>
<td>$\frac{1}{2} n^3 - \frac{1}{2} n$</td>
</tr>
<tr>
<td>Choleski (symmetric)</td>
<td>$\frac{1}{6} n^3 + \frac{3}{2} n^2 + \frac{1}{3} n$</td>
<td>$\frac{1}{6} n^3 + \frac{2}{3} n - \frac{7}{6} n$</td>
</tr>
<tr>
<td>Tridiagonal</td>
<td>$2n$</td>
<td>$4(n-1)$</td>
</tr>
<tr>
<td></td>
<td>$n^2$</td>
<td>$n^2 - n$</td>
</tr>
</tbody>
</table>

1 The number of arithmetic operations for iterative methods relates to one iteration.
APPENDIX 5

COMPARISON OF CONVERGENCE RATES OF ITERATIVE METHODS

Laplace's equation with Dirichlet boundary conditions is solved on a square region. The region is divided into \( n^2 \) squares with \( nh = n \), giving \((n-1)^2\) interior points. A consistent ordering is applied.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Convergence rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Point-iterative</td>
<td></td>
</tr>
<tr>
<td>Jacobi</td>
<td>( \frac{1}{2} h^2 )</td>
</tr>
<tr>
<td>JOR</td>
<td>( h^2 )</td>
</tr>
<tr>
<td>Gauss-Seidel</td>
<td>( h^2 )</td>
</tr>
<tr>
<td>Optimum SOR (( \omega = \omega_{opt} ) throughout)</td>
<td>2 ( h )</td>
</tr>
<tr>
<td>General: ( \omega_1^{(n)} \neq \omega_2^{(n)} )</td>
<td></td>
</tr>
<tr>
<td>Sheldon: ( \omega_1^{(1)} = \omega_2^{(1)} = 1, \omega_1^{(2)} = \omega_2^{(2)} = \omega_{opt}^{(3)} = \ldots = \omega_{opt}^{(3)} = \ldots = \omega_{opt} )</td>
<td>2 ( h )</td>
</tr>
<tr>
<td>MSOR</td>
<td>( \sim 2 \frac{1}{2} )</td>
</tr>
<tr>
<td>Modified Sheldon: ( \omega_1^{(1)} = 1, \omega_2^{(1)} = \omega_1^{(2)} = \omega_2^{(2)} = \omega_{opt} )</td>
<td>( \sim h^{1/2} )</td>
</tr>
<tr>
<td>SSOR</td>
<td>( \sim 2 h )</td>
</tr>
<tr>
<td>USSOR</td>
<td>not better than SOR</td>
</tr>
<tr>
<td>SMSOR</td>
<td>not better than SOR</td>
</tr>
<tr>
<td>USMSOR</td>
<td></td>
</tr>
<tr>
<td>Richardson (( m ) parameters)</td>
<td></td>
</tr>
<tr>
<td>( m \sim \infty )</td>
<td>( h )</td>
</tr>
<tr>
<td>( m \sim 2n )</td>
<td>( .89 h )</td>
</tr>
<tr>
<td>( m \sim n )</td>
<td>( .78 h )</td>
</tr>
<tr>
<td>( m \sim \frac{1}{2} n )</td>
<td>( .59 h )</td>
</tr>
</tbody>
</table>
# Appendix 5 (continued)

<table>
<thead>
<tr>
<th>Methods</th>
<th>Convergence rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>J-SI</td>
<td>$\sim h$</td>
</tr>
<tr>
<td>JOR-SI</td>
<td>$\sim h$</td>
</tr>
<tr>
<td>GS-SI</td>
<td>$\sim 2h$</td>
</tr>
<tr>
<td>Opt. SOR-SI</td>
<td>$\sim 2h$</td>
</tr>
<tr>
<td>MSOR-SI</td>
<td>$\sim 2h$</td>
</tr>
<tr>
<td>SSOR-SI</td>
<td>$\sim h^{1/2}$ (SOR)</td>
</tr>
<tr>
<td>Cyclic Chebyshev (CC-SI)</td>
<td>2h</td>
</tr>
</tbody>
</table>

**Block-iterative**

- **Line Jacobi (1 line)**: $h^2$
- **Line Gauss-Seidel (1 line)**: $2h^2$
- **Optimum SLQR**: (1 line) $\sqrt{2}h$; (2 lines) $4h$ (SSOR)
- **Block SSOR**: $= 2^{1/4}$ (SSOR-SI)

**ADI (m parameters):**

- **Peaceman-Rachford:**
  - fixed $m$
  - variable $m$

- **Wachspress:**
  - fixed $m$
  - variable $m$

| SER                           | not worse than any background method used (see Chapter 3) |
### APPENDIX 6

**STORAGE REQUIREMENT FOR SOLVING** $A\vec{x} = \vec{b}$ **WITH** $n \times n$ **MATRIX**

<table>
<thead>
<tr>
<th>Method</th>
<th>Storage locations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jacobi</td>
<td>$n^2 + 3n$</td>
</tr>
<tr>
<td>JOR</td>
<td>$n^2 + 3n$</td>
</tr>
<tr>
<td>Gauss-Seidel</td>
<td>$n^2 + 2n$</td>
</tr>
<tr>
<td>SOR</td>
<td>$n^2 + 2n$</td>
</tr>
<tr>
<td>SSOR</td>
<td>$n^2 + 2n$</td>
</tr>
<tr>
<td>Richardson</td>
<td>$n^2 + 3n + 2m$, $m$ - chosen number of parameters</td>
</tr>
<tr>
<td>Peaceman-Rachford</td>
<td>$n^2 + 3n$ (or $15n - 8$ if $C_L$ and $C_U$ are tridiagonal)</td>
</tr>
<tr>
<td>Steepest descent</td>
<td>$n^2 + 4n + 2$</td>
</tr>
</tbody>
</table>
APPENDIX 7
THE CARRÉ ALGORITHM

The procedure for computing the optimum relaxation factor is as follows [Carré (1961)]

a) Perform the first iteration with \( \omega_0^{(0)} = 1 \).
b) Set \( \omega_0^{(m+1)} = 1.375 \).
c) Perform a cycle of 12 iterations using \( \omega_0^{(m)} = \omega_0^{(m+1)} \).
d) Compute two norms, \( n^{(n)} \) and \( n^{(n-1)} \), of the displacement vector

\[
\delta^{(n)} = \delta^{(n)} - \delta^{(n-1)}
\]

where \( \delta^{(n)} \) is generated by the SOR iteration, using

\[
n^{(n)} = \| \delta^{(n)} \|_1 = \sum_{i \in N} |\delta_i^{(n)}|,
\]

or

\[
n^{(n)} = \| \delta^{(n)} \|_2 = \left( \sum_{i \in N} |\delta_i^{(n)}|^2 \right)^{1/2}
\]

at the end of each \( m \)-th cycle of 12 iterations.

e) Compute an estimate \( \eta_e \) to \( \eta_{\text{max}} \) from

\[
\eta_e^{(n)} = n^{(n)} / n^{(n-1)}
\]

or, at the end of the first cycle, by applying Aitken's transform

\[
\eta_e^{(n)} = p^{(n-2)} - \frac{(p^{(n-1)} - p^{(n-2)})^2}{p^{(n-2)} - 2p^{(n-1)} + p^{(n-2)}}
\]

where

\[
p^{(n)} = n^{(n)} / n^{(n-1)}
\]

Note: \( p^{(n)} \) has to converge monotonically in order to be applicable in (A7.5).
f) Compute an estimate \( \omega_o \) to \( \omega \) from

\[
\omega_o^{(m+1)} = 2 \left[ 1 + \left( \frac{\eta_e(m) + \omega_o(m) - 1}{\eta_e(m) (\omega_o(m))^2} \right)^{1/2} \right]^{-1}, \quad m \geq 1
\]  

(A7.7)

with \( \omega_o^{(1)} = 1.375 \).

g) Decrease the value of \( \omega_o^{(m+1)} \) obtained from (A7.7) according to

\[
\omega_e^{(m+1)} = \omega_o^{(m+1)} - \frac{(2 - \omega_o^{(m+1)})}{4}
\]  

(A7.8)

h) Make the following substitution

\[
\omega_o^{(m+1)} = \omega_e^{(m+1)}
\]  

(A7.9)

i) An estimate \( \omega_o^{(m+1)} \) of \( \omega \) is deemed satisfactory if

\[
|\omega_o^{(m+1)} - \omega_o^{(m)}| < \frac{2 - \omega_o^{(m+1)}}{20}
\]  

(A7.10)

or

\[
\frac{|\omega_o^{(m+1)} - \omega_o^{(m)}|}{2 - \omega_o^{(m+1)}} < 0.05
\]  

(A7.11)

j) Check the termination criterion for the maximum error of the solution, \( \| \xi^{(n)} \|_\omega \), which satisfies the following inequality.

\[
\| e^{(n)} \|_\omega \leq \frac{\eta_e \| \xi^{(n)} \|_\omega}{(1 - \eta_e)} \leq \epsilon_{tol}
\]  

(A7.12)

where

\[
\| \xi^{(n)} \|_\omega = \max_{i \in N} |\xi^{(n)}|,
\]  

(A7.13)

\[
\eta_o = \begin{cases} 
\eta_c(m) & \text{if } (A7.10) \text{ is not satisfied} \\
\omega_0^{(m+1)} - 1 & \text{if } (A7.10) \text{ is satisfied}
\end{cases}
\]  

(A7.14)

and \( \epsilon_{tol} \) is a given tolerance.

1) If (A7.10) and (A7.12) are not satisfied, repeat the process from
step (c).

m) If (A7.10) is satisfied and (A7.12) is not satisfied, continue the SOR iteration process from step (k)', computing \( \| \hat{v}^{(n)} \|_\infty \) only.

n) If (A7.12) is satisfied (regardless of (A7.10)), then stop.
APPENDIX 8

SPIRAL AND CONTOUR SCANNINGS

An experimental analysis indicates that the spectral radius of an iteration matrix, \( M \), satisfies the following relation

\[
\rho_c(M) < \rho_s(M) < \rho_r(M) = \rho_d(M) = \rho_{wb}(M)
\]  
(A8.1)

where the subscripts refer to the contour, spiral, raster, diagonal, and "white-black" scanning procedures. The inequalities are increased by complicated or multiply-connected boundary conditions.

The spiral scanning procedure proceeds from scanning the perimeter of the largest rectangle of elements in the array and then the enclosed nested set of rectangles. The contour scan performs essentially the same scanning operation but for more general boundary geometries. The problem of possible discontinuous orderings is remedied by using a self-consistent optimization process to choose the starting point that produces a minimum number of discontinuities, starting from a point to the boundary of Dirichlet type with the highest value. If there is, however, an ordering with a smaller number of discontinuities, that ordering is selected. A simple physical argument suggests that the propagation of the effect of Dirichlet and Neumann boundary conditions throughout the array should be then faster than for the other scannings.
THE THOMAS ALGORITHM FOR A TRIDIAGONAL MATRIX

The following constitutes the basis of the algorithm

\[ a_i u_{i-1} + b_i u_i + c_i u_{i+1} = d_i \quad \text{for } 1 \leq i \leq N \]  \hspace{1cm} (A9.1)

with \( a_1 = c_N = 0 \).

The algorithm is as follows:

First compute

\[ \beta_i = b_i - \frac{a_i c_{i-1}}{\beta_{i-1}} \quad \text{with } \beta_1 = b_1 \]  \hspace{1cm} (A9.2 a)

and

\[ \gamma_i = \frac{d_i - a_i \gamma_{i-1}}{\beta_i} \quad \text{with } \gamma_1 = \frac{d_1}{b_1} \]  \hspace{1cm} (A9.2b)

for \( 1 \leq i \leq N \).

Then the values of the dependent variable \( u \) are given recursively by

\[ u_N = \gamma_N \]

and

\[ u_i = \gamma_i - \frac{c_i u_{i+1}}{\beta_i} \quad \text{for } 1 \leq i \leq N \]  \hspace{1cm} (A9.3)

The procedure (A9.2) is equivalent to Gaussian elimination, and (A9.3) is the backward substitution.
APPENDIX 10

THRESHOLD JACOBI DIAGONALIZATION ALGORITHM

FOR EIGENVALUES AND EIGENVECTORS

A symmetric matrix $A$ of order $N$ is to be transformed to a similar diagonal matrix $A'$, with the eigenvalues in its diagonal elements. The eigenvectors are to be developed in an extra matrix $R = r_{ij}$. At the beginning of the process, the matrix $R$ is assumed to be $R = I$. The initial off-diagonal norm is obtained from

$$v_1^{(n)} = \left[ \sum_{i<k} 2a_{ik}^2 \right]^{1/2} \quad (A10.1)$$

and divided by $N$ at each stage to produce the threshold. The final norm is obtained from

$$v_F^{(m)} = \frac{v_1^{(m-1)} \times 10^{-6}}{N}, \quad n > n \quad (A10.2)$$

This value is set sufficiently small so that any off-diagonal $a_{pq}$ shall be smaller than $|v_F^{(m)}|$. Each pivotal element is annihilated in the following sequence:

$$\alpha = -a_{pq}^{(n)}$$

$$\beta = \frac{1}{2} \left( a_{pp}^{(n)} - a_{qq}^{(n)} \right)$$

$$\gamma = \text{sgn}(\beta) \frac{\alpha}{\sqrt{\alpha^2 + \beta^2}} 1/2$$

$$\sin \theta = \frac{\gamma}{\sqrt{2[1 + (1-\gamma^2)]}^{1/2}} $$

$$\cos \theta = (1 - \sin^2 \theta)^{1/2}$$

$$\{ \text{A10.3} \}$$
\[ t^{(n+1)} = a_{ip}^{(n)} \cos \theta - a_{iq}^{(n)} \sin \theta \]
\[ a_{iq}^{(n+1)} = a_{ip}^{(n)} \sin \theta - a_{iq}^{(n)} \cos \theta \]
\[ a_{ip}^{(n+1)} = t^{(n+1)} \]  \hspace{1cm} (A10.4)

\[ t^{(n+1)} = r_{ip}^{(n)} \cos \theta - r_{iq}^{(n)} \sin \theta \]
\[ r_{iq}^{(n+1)} = r_{ip}^{(n)} \sin \theta + r_{iq}^{(n)} \cos \theta \]
\[ r_{ip}^{(n+1)} = t^{(n+1)} \]  \hspace{1cm} (A10.5)

\[ a_{pp}^{(n+1)} = a_{pp}^{(n)} \cos^2 \theta - 2a_{pq}^{(n)} \sin \theta \cos \theta + a_{qq}^{(n)} \sin^2 \theta \]
\[ a_{pq}^{(n+1)} = a_{pq}^{(n)} \sin^2 \theta + 2a_{pq}^{(n)} \sin \theta \cos \theta + a_{qq}^{(n)} \cos^2 \theta \]
\[ a_{pq}^{(n+1)} = (a_{pp}^{(n)} - a_{qq}^{(n)}) \sin \theta \cos \theta + a_{pq}^{(n)} (\cos^2 \theta - \sin^2 \theta) \]  \hspace{1cm} (A10.6)

This iteration is repeated until all of the pivotal elements are less than the threshold.
APPENDIX 11

THE AITKEN AND HIGHER ORDER TRANSFORMS

The Aitken transformation can be derived in several ways. Some of them are presented below.

1. Consider a sequence \( \{x^{(i)}\} \) such that \( \lim_{i \to \infty} \{x^{(i)}\} = x \). Let the error be defined as

\[
e(n) = x^{(n)} - x
\]  
(A11.1)

Assume that the ratio of any successive errors is constant

\[
\frac{e(n)}{e(n-1)} = \frac{e(n-1)}{e(n-2)} = \ldots = \frac{e(1)}{e(0)} = \alpha
\]  
(A11.2)

If \(|\alpha| < 1\) then \( \{x^{(i)}\} \) has geometric convergence. Thus successive errors can be expressed as

\[
e(1) = e(0) \alpha
\]
\[
e(2) = e(1) \alpha = e(0) \alpha^2
\]
\[
\vdots
\]
\[
e(n) = e(n-1) \alpha = e(0) \alpha^n
\]  
(A11.3)

or

\[
x^{(n)} = x - (x - x^{(0)}) \alpha^n
\]  
(A11.4)

where \(x\) is the limit (or base) of the sequence. We want to eliminate the base explicitly. Thus

\[
x^{(1)} = x - (x - x^{(0)}) \alpha
\]  
(A11.5)

\[
x^{(2)} = x - (x - x^{(0)}) \alpha^2
\]  
(A11.6)
From (11.5)
\[ a = \frac{x - x^{(1)}}{x - x^{(0)}} \]  \hspace{1cm} (A11.7)

Substituting (A11.7) into (A11.6) yields
\[ \frac{(x - x^{(1)})^2}{x - x^{(0)}} = x - x^{(2)} \]
\[ x = \frac{[x^{(1)}]^2 - x^{(2)}x^{(0)}}{2x^{(1)} - x^{(2)} - x^{(0)}} = \frac{x^{(0)}x^{(2)} - [x^{(1)}]^2}{x^{(0)} - 2x^{(1)} + x^{(2)}} \]  \hspace{1cm} (A11.8)

For an \( n \)'th element of \( \{x^{(i)}\} \), (A11.8) becomes
\[ x = \frac{[x^{(n)}]^2 - x^{(n+1)}x^{(n-1)}}{2x^{(n)} - x^{(n+1)} - x^{(n-1)}} \]  \hspace{1cm} (A11.9a)
\[ x = \frac{[x^{(n-1)}]^2 - x^{(n)}x^{(n-2)}}{2x^{(n-1)} - x^{(n)} - x^{(n-2)}} \]  \hspace{1cm} (A11.9b)

or
\[ x = \frac{[x^{(n+1)}]^2 - x^{(n+2)}x^{(n)}}{2x^{(n+1)} - x^{(n+2)} - x^{(n)}} \]  \hspace{1cm} (A11.9c)

This property results from the assumption (A11.2). In other words, (A11.9a) involves the central differences, (A11.9b) - backward, and (A11.9c) - forward differences. It is seen that (A11.9b) is most suitable for the analysis of the SER method.

II. The second derivation uses determinant of matrices composed out of the elements of \( \{x^{(i)}\} \) and the displacements (or differences) between successive element values. It will be seen that the use of the central or, particularly, the backward- or forward differences produce formulae as in (A11.9). Assume the backward difference
\[
\delta(n) = x(n) - x(n-1)
\]

Then
\[
x = \frac{x(n-1) x(n-2) \begin{vmatrix} \delta(n) & x(n-2) \\ 1 & \delta(n-1) \end{vmatrix} - 1 \begin{vmatrix} \delta(n) & \delta(n-1) \\ 1 & 1 \end{vmatrix}}{\delta(n-1) - \delta(n)}
\]

\[
x = \frac{[x(n-1) \delta(n-1) - x(n-2) \delta(n)]}{\delta(n-1) - \delta(n)}
\]

\[
x = \frac{[x(n-1) x(n-2) x(n) - x(n-1)]}{2x(n-1) - x(n) - x(n-2)}
\]  

(A11.10)

The same formula is obtained if one uses
\[
x = \frac{x(n-1) x(n) \begin{vmatrix} \delta(n) & \delta(n+1) \\ 1 & \delta(n) \end{vmatrix} - \delta(n+1) \begin{vmatrix} \delta(n) & 1 \\ 1 & \delta(n) \end{vmatrix}}{\delta(n-1) - \delta(n)}
\]

\[
x = \frac{x(n-1) x(n+1) - [x(n)]^2}{x(n-1) - 2x(n) + x(n+1)}
\]  

(A11.11)

If the forward difference,
\[
\delta(n) = x(n+1) - x(n)
\]
is employed, then one must use
\[
x = \frac{x(n-1) x(n) \begin{vmatrix} \delta(n-1) & x(n) \\ 1 & \delta(n) \end{vmatrix} - \delta(n) \begin{vmatrix} \delta(n-1) & \delta(n) \\ 1 & 1 \end{vmatrix}}{\delta(n-1) - \delta(n)}
\]

\[
x = \frac{x(n-1) x(n+1) - [x(n)]^2}{x(n-1) - 2x(n) + x(n+1)}
\]  

(A11.12)

This method can be generalized to the k'th order nonlinear transform.

Eq. 3.130 of Sec. 3.8 is an extension of the form (A11.11). This form
has the following mnemonic advantage over the other formulae: the
superscript \( n \) is associated with the second diagonal of the numerator,
the superscripts of the first row go from \((n-k)\) to \((n)\), and the last
column has the superscripts from \((n)\) to \((n+k)\).

III. This method of derivation of the Aitken and higher order trans-
forms is probably the most important for it considers both the vector
(rather than scalar) sequences generated by some iterative methods (see
Eq. 3.47)

\[
\xi^{(n)} = M \xi^{(n-1)} + \xi
\]

and also the eigenvalues of the iteration matrix \( M \). A by-product of
this derivation relates to the convergence of the sequence \( \{x^{(i)}\} \); we
do not assume here that the sequence converges geometrically but we
show that the geometric convergence must be achieved for large number
of iterations.

Let the error vector be (see Sec. 2.6.2)

\[
\xi^{(n)} = \xi^{(n)} - \bar{\xi}
\]

\[
= \sum_{\forall m} \xi^n_m a_m \chi_m \quad , \quad m \in N
\]

Then each \( i \)th component of the vector can be written as

\[
[\xi^{(n)} \cdot \lambda_i] \lambda_i = \left[ \sum_{\forall m} \xi^n_m a_m (\chi_m \cdot \lambda_i) \right] \lambda_i \quad , \quad \text{(A11.15a)}
\]
or

\[
x^{(n)}_i - x_i = \sum_{\forall m} \xi^n_m b_m \quad , \quad \text{(A11.15b)}
\]

where

\[
b_m = a_m (\chi_m \cdot \lambda_i) \quad , \quad \text{(A11.16)}
\]

If the eigenvalues \( \xi_m \) of \( M \) are real,
\[ |\xi_1| > |\xi_n|, \quad n > 2, \ldots, N, \quad (A11.17) \]

and
\[ |\xi_1| < 1, \quad (A11.18) \]

then, for large \( n \),
\[ x_i^{(n)} - x_i \sim b_1 \xi_1^n, \quad (A11.19a) \]
\[ x_i^{(n+1)} - x_i \sim b_1 \xi_1^{n+1}. \quad (A11.19b) \]

Thus (as in 3.100),
\[ \frac{e_i^{(n+1)}}{e_i^{(n)}} \sim \xi_1. \quad (A11.20) \]

If each component of the error vector converges nearly geometrically,
then the vector itself converges nearly geometrically. Assume that the
equality in Eq. (A11.20) holds, and that the backward differences are
\[ \delta_i^{(n)} = x_i^{(n)} - x_i^{(n-1)}, \quad (A11.21a) \]
\[ \delta_i^{(n+1)} = \delta_i^{(n)} - \delta_i^{(n-1)} = x_i^{(n)} - 2x_i^{(n-1)} + x_i^{(n-2)}, \quad (A11.21b) \]

Substituting (A11.19) into (A11.21) yields
\[ \delta_i^{(n)} = b_1 \xi_1^n (1 - \xi_1^{-1}), \quad (A11.22a) \]
\[ \delta_i^{(n+1)} = b_1 \xi_1^{n+1} (1 - \xi_1^{-1}), \quad (A11.22b) \]
\[ \delta_i^{(n+2)} = b_1 \xi_1^{n+2} (1 - 2\xi_1^{-1} + \xi_1^{-2}) = b_1 \xi_1^n (1 - \xi_1^{-1})^2. \quad (A11.22c) \]

Squaring (A11.22a) and dividing it by (A11.22c) gives
\[ b_1 \xi_1 = \frac{[\delta_i^{(n)}]^2}{\delta_i^{(n+2)}}. \quad (A11.23) \]
Combining (A11.23) with (A11.19a) leads to Aitken's formula

\[
x_i = x_i^{(n)} - \frac{[\delta_i^{(n)}]^2}{\delta_i^{(n+1)-2}} = \frac{[x_i^{(n+1)}]^2 - x_i^{(n)}(n+2)}{2x_i^{(n+1)} - x_i^{(n+2)}}
\]  

(A11.24)

By analogy to (A11.10-12), it can be easily shown that

\[
\begin{vmatrix}
  x_i^{(n)} - x_i^{(n)} & \delta_i^{(n)} \\
  \delta_i^{(n)} & \delta_i^{(n+2)}
\end{vmatrix} = 0 ,
\]

(A11.25)

and

\[
x_i = \frac{[\delta_i^{(n)}]^2}{\delta_i^{(n+2)}} = x_i^{(n)} - \frac{[\delta_i^{(n)}]^2}{\delta_i^{(n+2)}} .
\]

(A11.26)

In order to obtain Aitken's transform we assumed that the matrix M has only one absolutely dominant eigenvalue \( \xi_1 \). When two eigenvalues absolutely dominate the spectrum of M then, by the same argument as in (A11.19),

\[
x_i^{(n)} - x_i^{(n)} \sim b_1 \xi_1^n + b_2 \xi_2^n ,
\]

(A11.27)

and one should devise a new transform for this sequence (process).

Similarly to (A11.22)

\[
\delta_i^{(n)} = b_1 \xi_1^n (1-\xi_1^{-1}) + b_2 \xi_2^n (1-\xi_2^{-1}) ,
\]

(A11.28a)

\[
\delta_i^{k(n)} = b_1 \xi_1^n (1-\xi_1^{-1})^k + b_2 \xi_2^n (1-\xi_2^{-1})^k , \quad k = 2,3,4 .
\]

(A11.28b)

In order to eliminate four unknowns \( \{b_1, b_2, \xi_1, \xi_2\} \) we need the differences up to the fourth order. This implies that this extrapolation formula requires \( (2k+1) = 9 \) points. Expanding (A11.25) leads to
\[
\begin{vmatrix}
\delta^{(n)} & \delta^2(n) & \delta^3(n) & \delta^4(n) \\
\delta^1 & \delta^1 & \delta^1 & \delta^1 \\
\delta^2 & \delta^3 & \delta^1 & \delta^1 \\
\delta^4 & \delta^3 & \delta^4 & \delta^1 \\
\end{vmatrix}
\]

Therefore,

\[
x_i = \begin{vmatrix}
\delta^1 & \delta^1 & \delta^1 & \delta^1 \\
\delta^2 & \delta^3 & \delta^4 & \delta^1 \\
\delta^2 & \delta^3 & \delta^4 & \delta^1 \\
\end{vmatrix}
\]

Similar formulae may be derived for three and more absolutely dominant eigenvalues but for vector sequences the work involved would be prohibitive.
APPENDIX 12

Various Forms of Aitken's Formula

The Aitken formula

\[ A = \frac{[x^{(n-1)}]^2 - x^{(n)}x^{(n-2)}}{2x^{(n-1)}x^{(n)} - x^{(n)} - x^{(n-2)}} \]  \hspace{1cm} (A12.1)

is very susceptible to computational errors since it has two multiplications in the numerator. One can derive three forms of (A11.1) with only one multiplication.

Form I:

\[ A = \frac{2x^{(n-1)}x^{(n-2)} - x^{(n)}x^{(n-2)}-[x^{(n-2)}]^2 + [x^{(n-1)}]^2 - 2x^{(n-1)}x^{(n-2)} + [x^{(n-2)}]^2}{2x^{(n-1)}x^{(n)} - x^{(n)} - x^{(n-2)}} \]

\[ = \frac{x^{(n-2)}[2x^{(n-1)} - x^{(n)} + x^{(n-2)}] + [x^{(n-1)} - x^{(n-2)}]^2}{2x^{(n-1)}x^{(n)} - x^{(n)} - x^{(n-2)}} \]

\[ = x^{(n-2)} + \frac{[x^{(n-1)} - x^{(n-2)}]^2}{[x^{(n-1)} - x^{(n-2)}]^2 - [x^{(n)} - x^{(n-1)}]^2} \]

\[ = x^{(n-2)} + \frac{[\delta^{(n-1)}]^2}{\delta^{(n-1)} - \delta^{(n)}} \]  \hspace{1cm} (A12.2)

Form II:

\[ A = \frac{2[x^{(n-1)}]^2 - x^{(n)}x^{(n-1)}x^{(n)} - x^{(n)}x^{(n-2)} - [x^{(n-1)}]^2 + x^{(n-1)}x^{(n)}x^{(n-2)} + x^{(n)}x^{(n-1)}x^{(n-2)}}{2x^{(n-1)}x^{(n)} - x^{(n)} - x^{(n-2)}} \]

where \[ a = x^{(n)}x^{(n-2)} \]. Thus
\[ A = x^{(n-1)} + \frac{-x^{(n-1)}[x^{(n-1)} - x^{(n-2)}] + x^{(n)}[x^{(n-1)} - x^{(n-2)}]}{2x^{(n-1)} - x^{(n)} - x^{(n-2)}} \]

\[ = x^{(n-1)} + \frac{[x^{(n)} - x^{(n-1)}][x^{(n-1)} - x^{(n-2)}]}{[x^{(n-1)} - x^{(n-2)}] - [x^{(n)} - x^{(n-1)}]} \]

\[ = x^{(n-1)} + \frac{\delta^{(n-1)} \delta^{(n)}}{\delta^{(n-1)} - \delta^{(n)}} \quad (A12.3) \]

**Form III:**

\[ A = \frac{2x^{(n)} x^{(n-1)} - [x^{(n)}]^2 - x^{(n)} x^{(n-2)} + [x^{(n)}]^2 - 2x^{(n)} x^{(n-1)} + [x^{(n-1)}]^2}{2x^{(n-1)} - x^{(n)} - x^{(n-2)}} \]

\[ = x^{(n)} + \frac{[x^{(n)} - x^{(n-1)}]^2}{[x^{(n-1)} - x^{(n-2)}] - [x^{(n)} - x^{(n-1)}]} \]

\[ = x^{(n)} + \frac{[\delta^{(n)}]^2}{\delta^{(n-1)} - \delta^{(n)}} \quad (A12.4) \]

The following formulae are useful in analysis of Aitken's transformation

(see Sec 3.5.2). Eq. (A12.4) may be rewritten into the form

\[ A = x^{(n)} + \frac{[\delta^{(n)}]^2}{\delta^{(n-1)} - \delta^{(n)}} \left[ \frac{\delta^{(n)} + \delta^{(n-1)}}{\delta^{(n)} + \delta^{(n-1)}} \right]^2 \]

\[ = x^{(n)} + \frac{\delta^{(n)}}{\delta^{(n)} + \delta^{(n-1)}} \left( \delta^{(n)} + \delta^{(n-1)} \right) \quad (A12.5) \]
Since
\[
\frac{\delta(n)}{\delta(n)+\delta(n-1)} = \frac{1}{2} \left[ 1 - \frac{\delta(n-1)-\delta(n)}{\delta(n)+\delta(n-1)} \right]
\]
\[
= \frac{1}{2} \left[ 1 - \mu(n) \right],
\]

hence
\[
A = x(n) + \frac{\frac{1}{2}(1-\mu(n))^2}{\mu(n)} (\delta(n)+\delta(n-1)).
\]

Another form of (A11.4) may also be useful
\[
A = x(n) + \frac{[\delta(n)]^2}{[\delta(n-1)]^2 - [\delta(n)]^2} (\delta(n)+\delta(n-1))
\]
\[
= x(n) + \frac{[\gamma(n)]^2}{1 - [\gamma(n)]^2} (\delta(n)+\delta(n-1))
\]
(A12.8)

where
\[
\gamma(n) = \frac{\delta(n)}{\delta(n-1)}.
\]

Still two other forms of (A11-4), containing the functions \(\mu\) and \(\gamma\), are possible
\[
A = x(n) + \frac{\delta(n)}{\frac{\delta(n-1)+\delta(n)}{\delta(n-1)-\delta(n)}} \delta(n).
\]
(A12.10)

Substituting (A11.6) into (A11.10) yields the first form
\[
A = x(n) + \frac{\frac{1}{2}(1-\mu(n))}{\mu(n)} \delta(n).
\]
(A12.11)

The other form is
\[ A = x^{(n)} + \frac{\delta^{(n)}}{\delta^{(n-1)}} \delta^{(n)} \quad (A12.12) \]

Using (A11.9) gives

\[ A = x^{(n)} + \frac{\gamma^{(n)}}{1 - \gamma^{(n)}} \delta^{(n)} \quad (A12.13) \]

Note that all the forms (A12.2-4), (A12.7-8), and (A11.12-13) contain scalars. Eqs. (A12.11) and (A12.13) are of particular interest because they can be transformed into forms containing vector quantities, viz., the functions \( \mu \) and \( \gamma \) may become measures of the vectors \( \delta \) and \( \chi \) respectively.
APPENDIX 13

**TŁ TRANSFORM (LYUSTERNIK'S APPROACH)**

This transform is efficient and, after some modification, very powerful for extremely slowly convergent sequences. Let the iteration matrix $M$ have real distinct eigenvalues with one of them, $\xi_1$, absolutely dominant but still $|\xi_1| < 1$. The displacement vector can be expanded into the eigenvectors $\chi_i$ of $M$

$$\delta^{(n)} = \chi^{(n)} - \chi^{(n+1)} = \sum_{i} \xi_i^n a_i \chi_i, \quad i \in \mathbb{N}, \quad n > 0. \quad (A13.1)$$

The error vector can be expressed as an infinite series of the displacement vectors

$$\delta^{(n)} = \chi^{(n)} - \chi = \sum_{k=0}^{\infty} \delta^{(n+k)} = \sum_{i} \frac{\xi_i^n}{1-\xi_1} a_i \chi_i, \quad n > 0 \quad (A13.2)$$

since

$$\sum_{k=0}^{\infty} \alpha \xi^k = \frac{\alpha}{1-\xi}, \quad |\xi| < 1. \quad (A13.3)$$

If $n >> 1$, then

$$\delta^{(n)} \approx \frac{\xi_1^n}{1-\xi_1} a_1 \chi_1 \quad (A13.4)$$

$$\delta^{(n)} \approx \xi_1^n a_1 \chi_1 \quad (A13.5)$$

Substituting (A13.5) into (A13.4) yields...
\[ \chi = \chi^{(n)} - \frac{1}{1-\xi_1} (\chi^{(n)} - \chi_{(n+1)}) \]  

(A13.6)

Since \( \xi_1 \) is not known we have to estimate it and expect that

\[ \chi^{(n+1)} = \chi^{(n)} - \frac{1}{1-\gamma} (\chi^{(n)} - \chi_{(n+1)}) \]

\[ = \frac{\chi^{(n+1)} - \gamma \chi^{(n)}}{1 - \gamma} \]  

(A13.6)

is closer to the exact solution, \( x \), than \( x^{(n)} \) and \( x^{(n+1)} \). Eq. (A13.6) is equivalent to (see Eq. 3.132 of Sec. 3.9)

\[ \chi^{(n)} = \frac{1}{1-\gamma} (\chi^{(n)} - \gamma \chi^{(n-1)}) \]  

(A13.7)

A practical application of this acceleration is similar to the SOR method: if \( x^{(n-1)} \) is the accepted value from the \( (n-1) \)st iteration then \( x^{(n)} \) is computed from some background iteration scheme

\[ \chi^{(n)} = M \chi^{(n-1)} + \chi \]

and the new value, \( \chi^{(n)} \), is calculated from (A13.7). Finally, a substitution \( \chi^{(n)} = \chi^{(n)} \) is required. That is a one-parameter relaxation scheme of the form

\[ \chi^{(n)} = \beta \chi^{(n)} + (1-\beta) \chi^{(n-1)} , \beta = \frac{1}{1-\gamma} \]  

(A13.8)

with \( \beta \) large.

If \( \xi_1 \approx 1 \) (very slow convergence), it is advisable to use
\[ y^{(n+p)} = x^{(n)} - \frac{1}{1-\gamma} \left( x^{(n)} - x^{(n+p)} \right) \quad (A13.9) \]

instead of (A13.6). The method is applicable when \( \xi_1 \) is a multiple eigenvalue.

**Error Estimation**

To determine the new error \( \chi^{(n)} - \xi \), let us assume that

\[ \xi_1 = \gamma + \epsilon, \quad (A13.10) \]

where \( \epsilon = 0 \left[ \left( \frac{\xi_2}{\xi_1} \right)^n \right] \), and \( |\xi_1| > |\xi_2| > \cdots > |\xi_i| \), \( i > 2, \ldots, N. \)

From (A13.7) we may introduce the matrix

\[ M_1 = \frac{1}{1-\gamma} [M - \gamma I]. \quad (A13.11) \]

Then

\[ M_1 (\chi^{(n)} - \xi) = \chi^{(n)} - \xi, \quad (A13.12) \]

and using (A13.2)

\[ M_1 (\chi^{(n)} - \xi) = \frac{1}{1-\gamma} \left[ \xi_1^n - \sum_{m=2}^{N} \frac{\xi_m^n(\xi_m - \gamma)}{1-\xi_m} a_m \xi_m \right]. \quad (A13.13) \]

Combining (A13.12) and (A13.13) leads to

\[ \chi^{(n)} - \xi = 0[\xi^n] \quad \text{for} \quad \epsilon = 0[\xi^n] \quad (A13.14) \]

This error is thus smaller than (from A13.2)

\[ \chi^{(n)} - \xi = 0[\xi_1^n] \quad (A13.15) \]

The convergence is better when the ratio \( \xi_2 / \xi_1 \) is smaller.