The numerical solution of elliptic partial differential equations by novel block iterative methods

This item was submitted to Loughborough University's Institutional Repository by the/an author.

Additional Information:

- A Doctoral Thesis. Submitted in partial fulfillment of the requirements for the award of Doctor of Philosophy of Loughborough University.

Metadata Record: https://dspace.lboro.ac.uk/2134/10471

Publisher: © Reza Sojoodi-Haghighi

Please cite the published version.
THE NUMERICAL SOLUTION OF ELLIPTIC PARTIAL DIFFERENTIAL EQUATIONS BY NOVEL BLOCK ITERATIVE METHODS

BY

REZA SOJOODI-HAGHIGHI, B.Sc., M.Sc.

A Doctoral Thesis submitted in partial fulfilment of the requirements for the award of Doctor of Philosophy of Loughborough University of Technology.

November, 1981.

Supervisor: Professor D.J. Evans, Ph.D., D.Sc.
Department of Computer Studies.

© by Reza Sojoodi-Haghigi, 1981.
DECLARATION

I declare that the following thesis is a record of research work carried out by me, and that the thesis is of my own composition. I also certify that neither this thesis nor the original work contained therein has been submitted to this or any other institution for a degree.

R. Sojoodi-Haghighi.
To my wife

TAHMINAH

without whose support and encouragement this work would not have been possible, and

to my sons,

OMID and EMAN.
ACKNOWLEDGEMENTS

I wish to acknowledge my sincere gratitude to my supervisor, Professor D.J. Evans. His keen enthusiasm for work and constant good humoured spirits have been a source of encouragement to me. His comments and criticisms have been appreciated as much as his learned guidance and advice during the past three years of study and preparation of this thesis.

My thanks also go to Dr. J. Shanehchi for his useful and valuable comments in private communications, and to Mrs. B. Wright who typed this thesis so professionally.

At last but by no means least I am deeply indebted to my mother-in-law Mrs. T. Motiee, for her moral and financial support throughout my studies abroad.
CONTENTS

CHAPTER 1: INTRODUCTION.

1.1 Introduction.

1.2 Remarks on the classification of partial differential equations.

1.3 Type of boundary conditions and boundary value problems.

1.4 Problems involving elliptic partial differential equations.

CHAPTER 2: BASIC LINEAR ALGEBRAIC THEORY AND APPROXIMATION METHODS FOR SOLVING P.D. Eqs. OF ELLIPTIC TYPE.

2.1 Notations and definitions.

2.2 Eigenvalues and eigenvectors.

2.3 Abstract mathematical structures.
   2.3.1 Linear space.
   2.3.2 Metric space.
   2.3.3 Normed linear space.
   2.3.4 Inner product space.
   2.3.5 Hilbert space.

2.4 Numerical methods for solving partial differential equations.
   2.4.1 Semi-analytical methods.
   2.4.2 Finite difference methods.
   2.4.3 The finite element method.

CHAPTER 3: METHODS FOR SOLVING LINEAR SYSTEMS OF EQUATIONS.

3.1 Introduction.

3.2 Methods for solving a set of linear equations.

3.3 Direct methods.
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3.1 Gaussian-elimination.</td>
<td>87</td>
</tr>
<tr>
<td>3.3.2 LU decomposition.</td>
<td>89</td>
</tr>
<tr>
<td>3.3.3 The Crout form of factorisation for symmetric matrices.</td>
<td>96</td>
</tr>
<tr>
<td>3.3.4 Choleski decomposition (square root method)</td>
<td>97</td>
</tr>
<tr>
<td>3.4 Iterative methods.</td>
<td>100</td>
</tr>
<tr>
<td>3.4.1 Basic iterative methods.</td>
<td>101</td>
</tr>
<tr>
<td>3.4.2 Convergence of point iterative methods.</td>
<td>104</td>
</tr>
<tr>
<td>3.4.3 Rate of convergence.</td>
<td>107</td>
</tr>
<tr>
<td>3.4.4 The optimum acceleration parameter for the S.O.R. Method.</td>
<td>115</td>
</tr>
</tbody>
</table>

CHAPTER 4: NEW PARALLEL STRATEGIES FOR SOLVING LINEAR SYSTEM OF EQUATIONS.

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1 Introduction.</td>
<td>123</td>
</tr>
<tr>
<td>4.2 Direct methods.</td>
<td>124</td>
</tr>
<tr>
<td>4.2.1 The factorisation process of the Q.I.F. method.</td>
<td>124</td>
</tr>
<tr>
<td>4.2.2 The computational cost analysis of the Q.I.F. method.</td>
<td>130</td>
</tr>
<tr>
<td>4.2.3 A modification of the Q.I.F. method.</td>
<td>138</td>
</tr>
<tr>
<td>4.2.4 The W.D.Z. factorisation.</td>
<td>144</td>
</tr>
<tr>
<td>4.2.5 The Choleski form of the Q.I.F. method.</td>
<td>150</td>
</tr>
<tr>
<td>4.3 Iterative methods.</td>
<td>156</td>
</tr>
<tr>
<td>4.3.1 Introduction.</td>
<td>156</td>
</tr>
<tr>
<td>4.3.2 The Quadrant Interlocking splitting technique.</td>
<td>157</td>
</tr>
<tr>
<td>4.3.3 Basic Quadrant Interlocking Iterative (Q.I.I.) methods.</td>
<td>161</td>
</tr>
<tr>
<td>4.3.4 Convergence of the basic Quadrant Interlocking Iterative methods.</td>
<td>165</td>
</tr>
<tr>
<td>4.3.5 Practical implementation of the parallel schemes.</td>
<td>197</td>
</tr>
</tbody>
</table>
CHAPTER 1

INTRODUCTION
1.1 **INTRODUCTION.**

Partial differential equations occur in a variety of forms in many different branches of Mathematical Physics. These equations can be classified according to various criteria, which may include such formal aspects as the number of dependent or independent variables, the order of the derivatives involved and the degree of non-linearity of the equations. These equations can also be categorised according to the methods which are employed for solving the partial differential equations, or according to particular properties which their solutions may possess.

In pure analysis, to solve partial differential equations we may use methods such as transformation or separation of variables. However, these methods can only be applied to very special classes of problems [WEINBERGER, 1965].

In practice, we employ numerical methods for the solution of such systems and although the types of methods used in numerical analysis of differential equations do not generally correspond with those used in mathematical analysis, both depend upon particular properties of the solution.

This thesis is concerned with the numerical solution of certain types of partial differential equations and therefore, with practical problems which can be treated by certain numerical methods. In practice, the numerical methods for solving the differential problems depend upon the nature of other auxiliary conditions, such as boundary or initial conditions. Certain types of auxiliary conditions are suitable only for certain corresponding types of differential equations, and in general, physical problems suggest auxiliary conditions which
are suitable for the differential equations involved in the problem. If, the auxiliary conditions are specified in such a way that there exists one and only one solution (uniqueness) for the differential problem, and in addition, a small change in these given auxiliary conditions result in a small change in the solution (stability), then the problem is said to be well-posed. Since numerical methods are by nature approximate processes, however, these methods rarely produce exact solutions for a given problem. However, it can be shown [see STEPHENSON, 1970], that if the differential problem is well-posed, then the solution of this problem is expected to be accurate.

To illustrate the properties of well-posed problems and to show the well posedness conditions we study the following problems.

Consider the Laplace's equation in two-dimensions:

$$U_{xx} + U_{yy} = 0$$  \hspace{1cm} (1.1.1)

with the given boundary conditions

$$U(x,0) = \frac{\sin nx}{n}$$

$$U_y (y=0) = 0$$

$$U_{yy}(y=0) = 0$$  \hspace{1cm} (1.1.2)

where $n$ is a parameter. It is well known that the analytical solution to the above system is obtained by separation of variables and can be shown to be:

$$U(x,y) = \frac{1}{n} \sin nx \cosh ny$$  \hspace{1cm} (1.1.3)

As $n$ tends to infinity, the boundary conditions converge to
\[ U(x,0) = 0 \] and \[ \frac{\partial U}{\partial y} = 0, \] which together with (1.1.1) implies, by Taylor's series, \[ U(x,y) = 0. \]

However, as \( n \) tends to infinity, \( U(x,y) \) given in (1.1.3) becomes infinitely large, so the problem defined by (1.1.1) - (1.1.2) is not well-posed.

Now, consider a general linear equation of second order, in two space variables,

\[
\mathcal{L}U = a_{xx}U + 2b_{xy}U + c_{yy}U + p_{x}U + q_{y}U + su = f \quad \text{in } \Omega \quad (1.1.4)
\]

\[
U = g(x,y) \quad \text{on } \partial \Omega \quad (1.1.5)
\]

where \( \Omega \) is a closed region with boundary \( \partial \Omega \).

Suppose the coefficients of the operator \( \mathcal{L} \) of equation (1.1.4) satisfy the conditions

\[
b^2 - ac < 0, \quad a > 0 \quad \text{and} \quad s \leq 0. \quad (1.1.6)
\]

Then it can be shown that,

a) if \( \mathcal{L}U \geq 0 \) in \( \Omega \) and \( U \) is not a constant, then \( U \) cannot attain a non-negative maximum in \( \Omega \),

b) the equation (1.1.4) with boundary condition (1.1.5) has a unique solution.

Let the new solution to the perturbed boundary data, i.e. \( g + \varepsilon \) be \( U + v \), then we have;

\[
\begin{align*}
\mathcal{L}v &= 0 \quad \text{in } \Omega \\
v &= \varepsilon \quad \text{on } \partial \Omega
\end{align*}
\]  \quad (1.1.7)
and it follows from (a) that the magnitude of the perturbation \(|v|\) in \(\Omega\) is bounded by \(|c|\). Therefore, the differential equation (1.1.4) with the boundary condition (1.1.5) is well-posed.

1.2 REMARKS ON THE CLASSIFICATION OF PARTIAL DIFFERENTIAL EQUATIONS.

Classification of partial differential equations can be achieved according to the nature of subsidiary conditions which must be imposed in order to obtain a well-posed problem. In the case of linear differential equations of the second order in two independent variables, this classification is easy to describe. The most general differential equation of this type is

\[
\frac{\partial^2 U}{\partial x^2} + B \frac{\partial^2 U}{\partial x \partial y} + C \frac{\partial^2 U}{\partial y^2} + D \frac{\partial U}{\partial x} + E \frac{\partial U}{\partial y} + FU + G = 0
\]

(1.2.1)

where \(A, B, C, D, E, F\) and \(G\) are constants or functions of the \(x, y\). Therefore, we can classify our problem according to the sign of \(B^2 - 4AC\).

\[
B^2 - 4AC
\]

(1.2.2)

For this we consider the following quadratic equation

\[
At^2 + Bt + C = 0
\]

(1.2.3)

If (1.2.2) is positive, then equation (1.2.3) has real roots \(t_1, t_2\) (say), then by a change of variable the equation (1.2.1) becomes

\[
\frac{\partial^2 U}{\partial \xi \partial \eta} = g_1 \left( \xi, \eta, \frac{\partial U}{\partial \xi}, \frac{\partial U}{\partial \eta}, U \right)
\]

(1.2.4)

which is the standard form for a Hyperbolic equation.
Where (1.2.2) is equal to zero, then (1.2.3) has two equal roots, and with a transformation of the variables (1.2.1) is transformed into

$$\frac{\partial^2 u}{\partial \eta^2} = g_2\left(\xi, \eta, \frac{\partial u}{\partial \xi}, \frac{\partial u}{\partial \eta}, u\right)$$

(1.2.5)

and it is said to be of parabolic type. Finally, if (1.2.2) is negative, the equation (1.2.3) has complex roots and by a suitable change of variables the equation (1.2.1) can be rewritten as:

$$p \frac{\partial^2 u}{\partial \xi^2} + q \frac{\partial^2 u}{\partial \eta^2} = g_3\left(\xi, \eta, \frac{\partial u}{\partial \xi}, \frac{\partial u}{\partial \eta}, u\right)$$

(1.2.6)

where p and q are positive. This is the standard form for an Elliptic differential equation.

The above classification, when the coefficients A, B and C are a function of the independent variables x and y, generally depends on the region in which the differential equation (1.2.1) is defined. For example, the differential equation

$$x \frac{\partial^2 u}{\partial x^2} + y \frac{\partial^2 u}{\partial x \partial y} + \frac{\partial^2 u}{\partial y^2} = 0,$$

is hyperbolic when $y^2 > 4x$. At all points on the parabola $y^2 = 4x$ the equation is parabolic and it is elliptic when $y^2 < 4x$.

For hyperbolic and parabolic equations to be well-posed, it is generally required that the auxiliary condition be specified on a region which is open in the direction of one of the variables. However, in elliptic equations it is a characteristic of well-posed problems that conditions are specified at all points on the boundary of a closed region.
The classical examples of these equations according to the above classification scheme are:

- **Parabolic**\[ \frac{\partial^2 T}{\partial x^2} = \frac{1}{\alpha^2} \frac{\partial T}{\partial t} \] Heat Conduction
- **Hyperbolic**\[ \frac{\partial^2 \phi}{\partial x^2} = \frac{1}{c^2} \frac{\partial^2 \phi}{\partial t^2} \] Wave Equation
- **Elliptic**\[ \frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} = 0 \] Laplace's equation.

### 1.3 TYPES OF BOUNDARY CONDITIONS AND BOUNDARY VALUE PROBLEMS

As mentioned earlier, in order to obtain well-posed differential equations, suitable boundary or initial conditions must be selected for different types of equations. The elliptic equations are, in general, associated with some boundary conditions whereas hyperbolic equations are generally associated with initial conditions. However, parabolic equations are intermediate in nature between the elliptic and hyperbolic equations.

Boundary conditions can be broadly divided into two major categories as described below.

(i) **Specific boundary conditions:** This type of boundary condition states that the dependent variable or variables take on specified values at specified points or regions in space and time. It can be subdivided into two types, homogeneous and nonhomogeneous. A homogeneous boundary condition is one such that, if \( U = U_1 \) is a function satisfying these boundary conditions, \( aU = aU_1 \), also
satisfies them where α is any parameter. For example boundary condition of type \( U = 0 \) or \( \frac{\partial U}{\partial n} = 0 \) will certainly meet these requirements.

(ii) General boundary conditions: A general boundary condition can be typified by a situation in which we have a field extending from one medium to another. Here we cannot specify in advance the exact values of the dependent variables on the interface, but nevertheless some relation between the values on the two sides of the interface can be specified.

According to the different types of boundary conditions we also have different types of boundary value problems.

Suppose that \( \Omega \) is a bounded region in the \((x,y)\)-plane with boundary \( \partial \Omega \). There are four main types of boundary value problems which can be defined on the region:

(i) Dirichlet problem, where \( U \) is specified at each point of the boundary, i.e. \( U \) given on \( \partial \Omega \).

(ii) Neumann problem, where values of normal derivative of the function \( U \) are given on the boundary, i.e. \( \frac{\partial U}{\partial n} \) given on \( \partial \Omega \).

(iii) Robbins' problem, where a combination of \( U \) and its derivatives is given on the boundary, i.e. \( \alpha(x,y)U + \beta(x,y)\frac{\partial U}{\partial n} \) given on \( \partial \Omega \).

(iv) \( U \) is given on part of \( \partial \Omega \) and \( \frac{\partial U}{\partial n} \) is given on the remaining part.

In this thesis we are mainly concerned with numerical methods for solving elliptic partial differential equations of the type
\[ \nabla^2 U = f \quad \text{on } \Omega \]

which can be shown to be well-posed if any of the above conditions is satisfied on the boundary \( \partial \Omega \) of the region \( \Omega \).

1.4 PROBLEMS INVOLVING ELLIPTIC PARTIAL DIFFERENTIAL EQUATIONS.

The solution of partial differential equations has been of much interest to the engineers and applied scientists since some physical problems can be interpreted in terms of such equations. These type of problems can be conveniently represented mathematically by means of which certain processes of logical reasoning can be carried out with relative ease.

In this section we turn our attention to those problems which can be described in terms of partial differential equations of Elliptic type. For example, it can be shown that the Laplace's equation, \( \nabla^2 U = 0 \) arises in theories associated with physical problems such as steady flow of heat or electricity in homogeneous conductors, the irrotational flow of incompressible fluid or potential problems in electricity, magnetism and gravitating matter of points devoid of these entities.

Poisson's equation \( \nabla^2 U = f \), on the other hand, arises in St. Venant's theory of torsion, the slow motion of incompressible viscous fluid, and the inverse square law theories of electricity, magnetism and gravitating matter at points where the charge density, pole strength or mass density respectively, are non-zero. [SMITH, 1965].

As mentioned before, in order to single out a particular physical situation, it is necessary to introduce boundary conditions. The
form of these conditions usually dictates the choice of coordinate systems. Having written the differential equation in a suitable coordinate system, it must be solved under the appropriate boundary conditions.

In order to analyse the above discussion further, we shall list some physical problems yielding Laplace's equation.

(i) **Heat Conduction.**

Suppose that $V$ is an arbitrary volume throughout which the following physical laws exist;

1) the quantity of heat in $V$ is proportional to its mass and to its absolute temperature,

2) heat flows from a higher to a lower temperature,

3) the rate of flow of heat across an area of $V$ is proportional to that area and to the space rate of change of temperature perpendicular to the area.

Furthermore, let $S$ be the surface surrounding this volume and $dS$ be an element of $S$. According to the laws 2 and 3 described above, it can be seen that $VTdS$ ($V$ is a notation for the gradient) gives the rate of change of temperature normal to the surface element $dS$ times the area of that element. Therefore, the outflow of heat through the element $dS$ of surface $S$ is given by

$$-kVTdS \quad . \tag{1.4.1}$$

$dS$ is defined as having the direction of the outward normal to the surface element, and by law 2 the heat flow in the direction of the negative temperature gradient accounting for the negative sign of
Hence, the total outflow of heat from \( V \) can be obtained by integrating the above equation over the entire surrounding surface \( S \), i.e.

\[
\text{outflow} = - \oiint_S k \nabla T \, dS
\]  

(1.4.2)

Law 1, however, indicates that the total loss, per unit time, of heat contained in \( V \) can be expressed as:

\[
\text{Heat loss} = - \iiint_V \frac{\partial}{\partial t} (\rho c \nabla T) \, dV
\]  

(1.4.3)

where \( \rho \) and \( c \) are density and specific heat of the medium respectively and are assumed to be constants.

We now assume that there is no creation or removal of thermal energy in the volume \( V \). Therefore, equating the heat loss from the volume with the outflow of heat through the surface \( S \) we have,

\[
\oiint_S k \nabla T \, dS = \iiint_V \frac{\partial}{\partial t} (\rho c \nabla T) \, dV
\]  

(1.4.4)

Applying the divergence theorem \cite{SCHELKUNOFF, 1957} to the left hand side of equation (1.4.4) we can deduce that:

\[
\oiint_S k \nabla T \, dS = \iiint_V \nabla \cdot (k \nabla T) \, dV
\]  

(1.4.5)

and hence, (1.4.4) can be rewritten as;

\[
\iiint_V \nabla \cdot (k \nabla T) \, dV = \iiint_V \frac{\partial}{\partial t} (\rho c \nabla T) \, dV
\]  

(1.4.6)
Since both terms of equation (1.4.6) are integrated over the same
volume, we get;

\[ \iiint \left[ \nabla \cdot (k \nabla T) - \frac{3}{\partial t} (c_p T) \right] dV = 0 \quad (1.4.7) \]

From (1.4.7) and the fact that \( V \) is an arbitrary volume
possessing the above mentioned physical laws, it can be so realised
that the integrand in (1.4.7) must itself be equal to zero throughout
any such region yielding the following equation as the partial
differential equation of heat conduction;

\[ \nabla \cdot (k \nabla T) = \frac{3}{\partial t} (c_p T) \quad (1.4.8) \]

Assuming that the temperature distribution is independent of
time then the quantity \( \frac{3}{\partial t} (c_p T) \) in (1.4.8) vanishes. Furthermore,
if the thermal conductivity \( k \) is constant throughout the medium
then (1.4.8) can be rewritten as;

\[ \nabla^2 T = 0 \quad (1.4.9) \]

which represent the Laplace equation.

(ii) Electrostatics.

The basic equation of electrostatics in a vacuum can be derived
from a generalization of certain empirical information as follows:

1) Electric charges are made up of discrete elements, the carriers
of which are sufficiently small to be treated as points.
2) The force between two point charges acts along the line joining the two charges, is inversely proportional to the square of the distance between them and proportional to the product of the magnitude of the two charges.

3) The force acting on a point charge by a group of point charges is given by the vector sum of the forces defined in (2).

4) Two types of charge occur. Like charges repel and unlike charges attract. See [WAYLAND, 1959].

Having these empirical laws we are now going to translate them into mathematical notation. Consider a distribution of point charges, \( q_1, q_2, \ldots, q_{n-1}, q_n \) (Fig. 1.1). The force on a test charge \( q_0 \) at \( p \) will be;

\[
\mathbf{F} = - \sum_{i=1}^{n} \frac{k q_0 q_i}{r_i^2} \frac{\mathbf{r}_i}{r_i} \tag{1.4.10}
\]

where \( \mathbf{r}_i \) is the radius vector from \( p \) to charge \( q_i \), \( r_i \) its absolute value, and \( k \) is a constant of proportionality, depending on the units employed.
Since two types of charge are postulated under law (4) we shall call one + and the other − (the ordinary electron carrying a negative charge). This sign convention, plus the repulsive nature of the force between like charges, accounts for the negative sign in equation (1.4.10).

The test charge \( q_o \) appears as a constant in each term of the summation in equation (1.4.10), hence it can be factored out to give;

\[
\vec{F} = q_o \left[ - \sum_{i=1}^{n} k \frac{q_i}{r_i^2} \hat{r}_i \right].
\]  

(1.4.11)

The bracketed term is defined as the electric intensity at \( p \) due to this distribution of the \( n \) charge \( q_1, \ldots, q_n \).

\[
\vec{E}(p) = - \sum_{i=1}^{n} k \frac{q_i}{r_i^3} \hat{r}_i.
\]  

(1.4.12)

It follows from (1.4.11) and (1.4.12) that the force on any test charge \( q_o \) placed at \( p \), due to this charge distribution, will be;

\[
\vec{F} = q_o \vec{E}(p).
\]  

(1.4.13)

(It should be noted that this assumes that the presence of the test charge does not disturb the distribution of the other charges.)

From known properties of the operator \( \nabla \), and the expression for the gradient and divergence in spherical coordinates, we obtain
\[ \nabla \cdot \frac{\hat{r}}{r^3} = \left[ \nabla \frac{1}{r^2} \right] \cdot \hat{r} + \frac{1}{r^3} \nabla \cdot \hat{r} \]

\[ = - \frac{3}{r^4} \hat{e}_r \cdot \hat{r} + \frac{1}{r^3} \left( \frac{1}{r^2} \frac{\partial r^3}{\partial r} \right) \]

\[ = - \frac{3}{r^3} + \frac{3}{r^3} = 0 \quad (1.4.14) \]

Consequently, for all points not occupied by one of the charges (for such points \( \vec{E} \) becomes infinite), we obtain

\[ \nabla \cdot \vec{E} = 0 \quad (1.4.15) \]

Another consequence of laws 1 to 4 is that the electric intensity in charge-free space can be written as the gradient of a scalar.

Consider a force field due to a single force centre \( O \), such that the force at any point \( p \) is a function only of the distance \( r \) between \( O \) and \( p \), and is directed along the line joining \( O \) and \( p \). Let \( \hat{e}_r \) be a unit vector in the \( Op \) direction. Then, for a central force system

\[ \vec{E} = f(r) \hat{e}_r \quad (1.4.16) \]

and the line integral of \( \vec{E} \) taken around any closed curve \( c \) (Fig. 1.2), will be

\[ \int_c \vec{E} \cdot d\vec{l} = \int_c f(r) \hat{e}_r \cdot d\vec{l} \quad (1.4.17) \]
The expression $\mathbf{e}_r \cdot d\mathbf{r}$ represents the component of $d\mathbf{r}$ in the $\hat{r}$ direction, or the scalar change of length of $r$ as the point $p$ moves around the curve. Consequently, if $f(r)$ is single valued;

$$\int_C \mathbf{E} \cdot d\mathbf{r} = \int_C f(r) dr = 0 . \quad (1.4.18)$$

The vanishing of the line integral (1.4.18) for any central force field assures the vanishing of such integrals for the inverse square field of electrostatics. Furthermore, its vanishing for the field of a typical charge means that it will vanish for the field due to a distribution of such charges under the assumption that there is no interaction between the fields.

The vanishing of the line integral of the electric intensity about any closed path shows that the line integral of $\mathbf{E}$ between any two charge-free points is independent of the path. Consequently we can represent $\mathbf{E}$ as the gradient of a scalar potential function

$$\mathbf{E} = - \nabla \phi . \quad (1.4.19)$$

If we combine equation (1.4.15) with (1.4.19) we find

$$\nabla \cdot \mathbf{E} = - \nabla \cdot \nabla \phi$$

$$= - \nabla^2 \phi = 0 . \quad (1.4.20)$$

Hence we see that the electrostatic potential obeys Laplace's equation.
CHAPTER 2

Basic Linear Algebraic Theory and Approximation Methods for Solving P.D. Eqs. of Elliptic Type.
2.1 NOTATIONS AND DEFINITIONS.

In this thesis, we shall use capital letters for matrices, and unless otherwise specified, a matrix indicated by a capital letter is assumed to be square. The elements of a matrix \( A \) (say) will be represented by lower case letters with two subscripts such as \( a_{i,j} \), where \( i \) denotes its position in the \( i^{th} \) row and \( j \) its position in the \( j^{th} \) column. The identity matrix, usually denoted by \( I \), which is a matrix with elements \( I_{i,j} \) defined as follows:

\[
\begin{align*}
I_{i,i} &= 1 \quad \text{for all } 1 \leq i \leq n \\
I_{i,j} &= 0 \quad \text{for all } 1 \leq i \neq j \leq n 
\end{align*}
\]  

(2.1.1)

The inverse of a square matrix \( A \) (if it exists) is a matrix usually denoted by \( A^{-1} \), if and only if

\[
A \cdot A^{-1} = A^{-1} \cdot A = I .
\]  

(2.1.2)

The matrix \( B \) is said to be transpose of a matrix \( A \) if

\[
b_{i,j} = a_{j,i} \quad \text{for all } 1 \leq i, j \leq n .
\]  

(2.1.3)

The matrix \( B \) is usually denoted by \( A^T \). If the matrix \( A \) is symmetric (i.e. \( a_{i,j} = a_{j,i} \) for all \( i,j \)) then,

\[
A^T = A .
\]  

(2.1.4)

The matrix \( A \) is said to be orthogonal if

\[
A^T \cdot A = I .
\]  

(2.1.5)
The set of elements \( a_{i,i} \), \( i = 1,2,\ldots,n \) of matrix \( A \) is a principal (main) diagonal of \( A \). If \( a_{i,j} = 0 \) for \( |i-j| > 0 \), i.e. all elements are zero except for the principal diagonal, the matrix \( A \) is said to be a diagonal matrix.

The matrix \( A = (a_{i,j}) \) is diagonally dominant if

\[
|a_{i,i}| > \sum_{j=1, j\neq i}^{n} |a_{i,j}| \quad \text{for all } 1 \leq i \leq n .
\]

\( A \) is said to be strictly diagonally dominant if strict inequality holds for all \( 1 \leq i \leq n \) in (2.1.6).

The matrix \( A = (a_{i,j}) \) is a lower triangular (strictly lower triangular) matrix if \( a_{i,j} = 0 \) for \( i < j \) (\( i \leq j \)).

The matrix \( A = (a_{i,j}) \) is a upper triangular (strictly upper triangular) matrix if \( a_{i,j} = 0 \) for \( i > j \) (\( i \geq j \)).

A matrix \( A = (a_{i,j}) \) is said to be a band matrix if \( a_{i,j} = 0 \) for \( |i-j| > m \) ("bandwidth" \( 2m+1 \) since this is the number of non-zero diagonals in the band), for example, if \( a_{i,j} = 0 \) for \( |i-j| > 1 \), i.e. all elements are zero except for the principal diagonal and sub and super principal diagonals, then the matrix \( A \) is tridiagonal. More generally, if there are \( m_1 \) non-zero diagonals immediately below and \( m_2 \) non-zero diagonals immediately above the principal diagonal, then \( a_{i,j} = 0 \) for \( i > j + m_1 \) and \( j > i + m_2 \) and matrix \( A \) is a band matrix of "bandwidth" \( p = m_1 + m_2 + 1 \).

If most of the elements \( a_{i,j} \) of the matrix \( A \) are zero then \( A \) is said to be a sparse matrix.

If most of the elements \( a_{i,j} \) of the matrix \( A \) are non-zero, then the matrix \( A \) is a full (Dense) matrix.
The determinant of a matrix $A$ will be denoted either by $\det(A)$ or $|A|$. A matrix $A$ is non-singular if and only if $\det(A) \neq 0$.

The conjugate of matrix $A = (a_{i,j})$ is denoted by

$$\overline{A} = (\overline{a}_{i,j}). \quad (2.1.7)$$

The Hermitian transpose (or conjugate transpose) of a matrix $A$ will be denoted by $A^H$, note that

$$A^H = \overline{A}^T = \overline{A}^T = (\overline{a}_{j,i}). \quad (2.1.8)$$

The matrix $A = (a_{i,j})$ is said to be a Hermitian matrix if

$$A^H = A. \quad (2.1.9)$$

The Trace of a matrix $A = (a_{i,j})$ is given by

$$\text{Trace } A = \sum_{i=1}^{n} a_{i,i}. \quad (2.1.10)$$

A permutation matrix $P = (p_{i,j})$ is a matrix with exactly one non-zero element, namely unity, in each row and each column. Thus for example,

$$P = \begin{bmatrix}
0 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0
\end{bmatrix} \quad (2.1.11)$$

is a permutation matrix of order 4. For any permutation matrix $P$ we have

$$P*P^T = P^T*P = I \quad (2.1.12)$$

$$P^T = P^{-1}. \quad (2.1.13)$$
Vectors are represented as underlined lower case letters with their elements denoted by the same letter, but with a lower suffix giving the position of that element in the vector. For example, the column vector $\mathbf{v}$ with its elements denoted by $v_i$ is written as:

$$
\mathbf{v} = \begin{bmatrix}
v_1 \\
v_2 \\
\vdots \\
v_n
\end{bmatrix}
$$

(2.1.14)

To each column vector as shown above there exists a corresponding row vector which is usually denoted by the same name but with upper suffix $T$. The corresponding row vector of the column vector $\mathbf{v}$ in (2.1.14) is

$$
\mathbf{v}^T = [v_1, v_2, \ldots, v_n]
$$

(2.1.15)

In fact, the column vector is a $N \times 1$ column matrix.

Zero (or Null) matrix $(0)$ is the matrix with all its elements zero.

A matrix $A = (a_{i,j})$ of order $n$ is irreducible, if given any two disjoint subsets $S$ and $T$ of $W$, the set of the first $n$ positive integers, such that $S + T = W$, there exist $i \in S$ and $j \in T$ such that $a_{i,j} \neq 0$.

Theorem: The matrix $A$ is irreducible if and only if there does not exist a permutation matrix $P$ such that $P^{-1}AP$ has the form:

$$
P^{-1}AP = \begin{pmatrix}
F & 0 \\
G & H
\end{pmatrix}
$$

(2.1.16)

where $F$ and $H$ are square matrices and $O$ is a null matrix, [YOUNG, 1971].
2.2 EIGENVALUES AND EIGENVECTORS.

The eigen problem for a given matrix $A$ of order $n$ is to find the eigenvalues $\lambda$ and the eigenvectors $v$ ($v \neq 0$) such that

$$Av = \lambda v \quad (2.2.1)$$

The eigen problem may be written as:

$$(A - \lambda I)v = 0 \quad (2.2.2)$$

which is a system of $n$ homogeneous linear equations. This system has a non-trivial solution, $v \neq 0$, if and only if the matrix of the system is singular, i.e.

$$\det(A - \lambda I) = 0 \quad (2.2.3)$$

(2.2.3) is called the characteristic equation of the matrix $A$.

The characteristic equation (2.2.3) has $n$ roots $\lambda_i$, $i = 1, 2, \ldots, n$ say (counting multiplicities), i.e. an eigenvalue problem has $n$ solutions $\lambda_i$, $i = 1, \ldots, n$. Associated with each $\lambda_k$ there is at least one eigenvector solution $v_k$ of

$$(A - \lambda_k I)v_k = 0 \quad (2.2.4)$$

(note that, any arbitrary multiple of $v_k$ is also a solution).

Theoretically,

$$\det(A - \lambda I) = \prod_{i=1}^{n} (\lambda_i - \lambda)$$

thus,

$$\det(A) = \prod_{i=1}^{n} \lambda_i \quad (2.2.5)$$

and

$$\text{Trace } (A) = \sum_{i=1}^{n} \lambda_i \quad .$$
The spectral radius of matrix $A$ is defined as:

$$\rho(A) = \max_{1 \leq i \leq n} |\lambda_i|.$$ \hspace{1cm} (2.2.6)

Let

$$g(t) = \frac{p(t)}{q(t)}$$

be a rational function, then

$$g(A) = p(A)[q(A)]^{-1}$$

is defined provided $q(A)$ is non-singular.

If $\lambda$ is an eigenvalue of $A$, $g(\lambda)$ is an eigenvalue of $g(A)$, for example, $\lambda^2$ is an eigenvalue of $A^2$, $\lambda - h$ is an eigenvalue of $A - hI$ and $\lambda^{-1}$ is an eigenvalue of $A^{-1}$.

The similarity transformation of the matrix $A$ is defined by $S^{-1}AS$, provided $S$ is a non-singular matrix.

Now, suppose

$$A v_i = \lambda_i v_i$$ \hspace{1cm} (2.2.7)

then

$$S^{-1}A S^{-1} v_i = \lambda_i S^{-1} v_i.$$ \hspace{1cm} (2.2.8)

Therefore,

$$S^{-1}AS x_i = \lambda_i x_i$$ \hspace{1cm} (2.2.9)

where $x_i = S^{-1} v_i$, thus the eigenvalues of $S^{-1}AS$ are the same as the eigenvalues of $A$. The eigenvectors are related by

$$v_i = S x_i$$ \hspace{1cm} (2.2.10)

(note that, the trace of a matrix is invariant under a similarity transformation.)
Given a vector $\mathbf{x}$ and a Hermitian matrix $\mathbf{A}$ then the Hermitian form is:

$$\mathbf{x}^H \mathbf{A} \mathbf{x} = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} \bar{x}_i x_j .$$  \hspace{1cm} (2.2.11)

This is a real number, since $(\mathbf{x}^H \mathbf{A} \mathbf{x})^H = \mathbf{x}^H \mathbf{A}^H \mathbf{x} = \mathbf{x}^H \mathbf{A} \mathbf{x}$.

Given a real vector $\mathbf{x}$ and a real symmetric matrix $\mathbf{A}$ then the quadratic form is:

$$\mathbf{x}^T \mathbf{A} \mathbf{x} = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} x_i x_j .$$ \hspace{1cm} (2.2.12)

**Definition (2.2.1):** A Hermitian matrix is positive definite if its Hermitian form is positive for all $\mathbf{x} \neq 0$, i.e.

$$\mathbf{x}^H \mathbf{A} \mathbf{x} > 0 \quad \forall \mathbf{x} \neq 0 .$$ \hspace{1cm} (2.2.13)

**Definition (2.2.2):** A real symmetric matrix is positive definite if its quadratic form is positive for all $\mathbf{x} \neq 0$, i.e.

$$\mathbf{x}^T \mathbf{A} \mathbf{x} > 0 \quad \forall \mathbf{x} \neq 0 .$$ \hspace{1cm} (2.2.14)

See [JENNINGS, 1977].

**Theorem 2.2.1** Necessary and sufficient conditions for a Hermitian (or a real symmetric) matrix $\mathbf{A}$ to be positive definite is that, the eigenvalues of $\mathbf{A}$ are all positive. [YOUNG, 1971].
GERSCHGORIN'S THEOREM.

If \( \lambda \) is any eigenvalue of \( A \) then

\[
|a_{i,i} - \lambda| \leq \sum_{j=1}^{n} |a_{i,j}|
\]

for at least one \( i \), (i.e. every eigenvalue of \( A \) lies in at least one of the discs with centres \( a_{i,i} \) and radii \( \sum_{j \neq i} |a_{i,j}| \)).

Proof.

Let \( A \mathbf{x} = \lambda \mathbf{x} \) where \( \mathbf{x}^T = [x_1, \ldots, x_n] \)

then \( (A - \lambda I)\mathbf{x} = 0 \)

i.e. \( (a_{i,i} - \lambda)x_i + \sum_{j \neq i} a_{i,j} x_j = 0 \quad \forall \ i \)

thus \( |a_{i,i} - \lambda| \leq \sum_{j \neq i} |a_{i,j}| \frac{|x_j|}{|x_i|} \quad \forall \ i \).

Choosing \( i \) such that \( x_i \) is the element of the largest modulus gives the result that

\[
|a_{i,i} - \lambda| \leq \sum_{j=1}^{n} |a_{i,j}|
\]

for at least one \( i \).

COROLLARY:

A strictly diagonally dominant matrix is \( \text{nilpotent} \) definite.
Special matrices: Let \( A = (a_{i,j}) \) be a matrix of order \( n \), the matrix \( A \) is said to be an L-matrix if

\[
a_{i,i} > 0, \quad 1 \leq i \leq n \quad (2.2.16)
\]

\[
a_{i,j} \leq 0, \quad i \neq j \quad \text{and} \quad i,j = 1,2,\ldots,n \quad (2.2.17)
\]

Definition (2.2.3): A real matrix \( A \) is a Steiltjes matrix if \( A \) is a positive definite L-matrix.

Definition (2.2.4): A real and non-singular matrix is said to be a M-matrix if (2.2.17) holds and \( A^{-1} \succ 0 \).

(Note: By \( A \succeq 0 \) we mean that all elements of matrix \( A \) are real and non-negative, i.e. non-negative matrix).

Theorem (2.2.2): If \( A \) is a positive definite matrix, then for any non-singular matrix \( L \) the matrix \( M \) given by

\[
M = L A L^H
\]

is a positive definite. [YOUNG, 1971].

2.3 ABSTRACT MATHEMATICAL STRUCTURE.

2.3.1 Linear Space:

Definition (2.3.1): A linear vector space (or a linear space) \( X \) is a set of elements (or vectors) such that if \( x,y \in X \) then \( \alpha x + \beta y \in X \), where \( \alpha \) and \( \beta \) are scalars. More precisely the operations (called
addition and scalar multiplication) must satisfy the following axioms:

(i) \[ x + y = y + x \] (commutative) (2.3.1)

(ii) \[ x + (y+z) = (x+y) + z \] (associative) (2.3.2)

(iii) There exists a unique element \( 0 \in X \) such that

\[ x + 0 = x \quad \text{for all} \quad x \in X \] (zero element) (2.3.3)

(iv) For each \( x \in X \) there exists a unique element \( -x \) such that

\[ x + (-x) = 0 \] (inverse) (2.3.4)

(v) \[ a(bx) = (ab)x \] (2.3.5)

(vi) \[ a(x+y) = ax + ay \] (2.3.6)

(vii) \[ (a+b)x = ax + bx \] (2.3.7)

(viii) \[ 1(x) = x \] (2.3.8)

The scalars which have appeared in the above axioms (i.e. \( a, b, \ldots \)) can be real (or complex).

Definition (2.3.2): The elements \( x_i \in X, \quad i = 1, 2, \ldots, n \) are linearly independent if:

\[ \alpha_1 x_1 + \alpha_2 x_2 + \ldots + \alpha_n x_n = 0 \implies \alpha_i = 0, \quad i = 1, 2, \ldots, n \] (2.3.9)

otherwise the elements are linearly dependent.
Definition (2.3.3): $X$ has finite dimension $n$ (a positive integer) if there exist $n$ elements $x_i \in X$, $i = 1, 2, \ldots, n$ which are linearly independent, whilst any $n+1$ elements are linearly dependent. If no such value of $n$ exists then $X$ is called an infinite dimensional space.

Definition (2.3.4): A set of elements $x_1, x_2, \ldots$, is a basis for $X$ if the elements $x_i$ are linearly independent and if every $x \in X$ can be expressed uniquely as a linear combination of the $x_i$.

Linear Spaces of Functions: A function, considered as a whole, can frequently be thought of as constituting an element of a linear space. For example:

(i) $C[a, b]$ represents the space of continuous functions real valued on the real interval $a \leq t \leq b$.

(ii) $C^n[a, b]$ denotes the linear space of $n$ times continuously differentiable functions on $[a, b]$.

(iii) $L^p[a, b]$ is a linear space of measurable functions such that

\[ \int_a^b |f(t)|^p \, dt < \infty \quad \text{for all } f \in L^p[a, b] . \]  

(2.3.10)

Definition (2.3.5): A function $f$ is bounded on $\Omega$ if there exists a constant $M$ such that

\[ |f(P)| \leq M \quad \text{for all } P \in \Omega . \]  

(2.3.11)
\( B(\Omega) \) denotes a linear space of bounded functions on \( \Omega \). An operator is frequently used as a function whose domain is itself a function space, e.g.

\[
\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}
\]

is such that

\[
\Delta : c^2(\Omega) + c(\Omega).
\]

The operator \( L : X \rightarrow Y \) is said to be linear if

\[
L(\alpha x_1 + \beta x_2) = \alpha L(x_1) + \beta L(x_2),
\]

where \( x_1 \) and \( x_2 \in X \) which is a linear space.

2.3.2 Metric Spaces:

A metric space \( X \) (not necessarily a linear space) is a set of elements, such that for any pair \( x, y \in X \) a real non-negative "distance function" \( d(x,y) \) is defined, satisfying the axioms:

(i) \( d(x,y) \geq 0 \) \hspace{1cm} (2.3.13)

(ii) \( d(x,y) = 0 \) if and only if \( x = y \) \hspace{1cm} (2.3.14)

(iii) \( d(x,y) = d(y,x) \) (symmetric) \hspace{1cm} (2.3.15)

(iv) \( d(x,y) \leq d(x,z) + d(z,y) \). \hspace{1cm} (2.3.16)

Definition (2.3.6): An element \( x \in X \) (metric space) is a limit of the infinite sequence \( x_1, x_2, \ldots, x_n, \ldots \) of elements in \( X \) if
\[ d(x_n, x) \to 0 \quad \text{as } n \to \infty \quad (2.3.17) \]

i.e. given any \( \varepsilon > 0 \) there exists \( N(\varepsilon) \) such that \( d(x_n, x) < \varepsilon \) for all \( n > N(\varepsilon) \). We say that \( \{x_n\} \) converges to \( x \), where \( \{x_n\} \) denotes the infinite sequence. It can be proved that if \( \{x_n\} \) has a limit \( x \), then this limit is unique. See [RUDIN, 1976].

**Definition (2.3.7):** A sequence \( \{x_n\} \) of elements in a metric space \( X \) is a Cauchy sequence (fundamental sequence) if for every \( \varepsilon > 0 \) there exists \( N(\varepsilon) \) such that

\[ d(x_m, x_n) < \varepsilon \quad \text{for all } m, n \geq N(\varepsilon) . \]

It can be seen that if \( \{x_n\} \) converges to \( x \in X \) then the sequence is a Cauchy sequence, but the converse of this is not necessarily true. e.g. \( X \) set of all rational numbers. Then \( x_n = (1 + \frac{1}{n})^n \) defines a Cauchy sequence \( \{x_n\} \) in \( X \), but \( \lim_{n \to \infty} x_n = e \) is an irrational number.

**Definition (2.3.8):** If every Cauchy sequence in a metric space \( X \) converges to an element in \( X \), then the space is said to be complete.

### 2.3.3 Normed Linear Space:

A linear space \( X \) is a normed linear space if for any \( x \in X \), there is defined a real number \( \|x\| \) satisfying the following axioms:

(i) \[ \|x\| \geq 0 \quad (2.3.18) \]

(ii) \[ \|x\| = 0 , \quad \text{if and only if } x = 0 \quad (2.3.19) \]
Note that, a normed linear space is a metric space with \( d \) defined as:
\[
d(x, y) = \|x - y\| \tag{2.3.22}
\]
It can be easily verified that the above definition satisfies the metric space axioms.

If \( X \) is a complete normed linear space, and if for any \( \varepsilon > 0 \) we can find \( N(\varepsilon) \) such that
\[
\|x_n - x_m\| < \varepsilon, \quad m, n > N(\varepsilon)
\]
then there is an \( x \in X \) such that
\[
\lim_{n \to \infty} \|x - x_n\| = 0 \tag{2.3.24}
\]
A complete normed linear space is often called a Banach space.

In the following, we mention the vector and matrix norm and the relation between them.

(i) **Vector Norms:**

Let \( x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \) denote a column vector. A norm of the vector \( x \) is a real number \( \|x\| \) satisfying the axioms:

\[
\text{(1)} \quad \|x\| > 0 \tag{2.3.25}
\]
(2) \( \|x\| = 0 \), if and only if \( x = 0 \) \hspace{1cm} (2.3.26)

(3) \( \|\alpha x\| = |\alpha| \|x\| \), for any scalar \( \alpha \) \hspace{1cm} (2.3.27)

(4) \( \|x + y\| \leq \|x\| + \|y\| \). \hspace{1cm} (2.3.28)

The most frequently used vector norms are

\[ \|x\|_1 = \sum_{i=1}^{n} |x_i| \] \hspace{1cm} (1 norm) \hspace{1cm} (2.3.29)

\[ \|x\|_2 = \left( \sum_{i=1}^{n} |x_i|^2 \right)^{1/2} \] \hspace{1cm} (Euclidean vector norm) \hspace{1cm} (2.3.30)

\[ \|x\|_\infty = \max_{1 \leq i \leq n} |x_i| \] \hspace{1cm} (\( \infty \) norm) \hspace{1cm} (2.3.31)

(2.3.29) - (2.3.31) are particular cases of

\[ \|x\|_p = \left( \sum_{i=1}^{n} |x_i|^p \right)^{1/p} \] \hspace{1cm} (p norm) \hspace{1cm} p \geq 1 \hspace{1cm} (2.3.32)

(ii) **Matrix Norms:**

A norm of \( n \times n \) matrix \( A \) is a real number \( \|A\| \) satisfying the axioms:

(1) \( \|A\| \geq 0 \) \hspace{1cm} (2.3.33)

(2) \( \|A\| = 0 \), if and only if \( A = (0) \) \hspace{1cm} (2.3.34)
(3) \[ \| \alpha A \| = |\alpha| \| A \| \] , for any scalar \( \alpha \) \hspace{1cm} (2.3.35)

(4) \[ \| A + B \| \leq \| A \| + \| B \| \] . \hspace{1cm} (2.3.36)

In applications we also require the additional axiom:

(5) \[ \| AB \| \leq \| A \| \| B \| \] . \hspace{1cm} (2.3.37)

Compatible (Consistent Norm):

A vector and matrix norm are compatible if

\[ \| Ax \| \leq \| A \| \| x \| \] \hspace{1cm} (2.3.38)

Subordinate Matrix Norms:

Given a vector norm, then the matrix norm which is subordinate to this vector norm is defined by

\[ \| A \| = \max_{x \neq 0} \frac{\| Ax \|}{\| x \|} = \max_{x \neq 0} \frac{\| A \|}{\| x \|} \hspace{1cm} (2.3.39) \]

It can be shown that

(1) \[ \| A \|_1 = \max_{1 \leq j \leq n} \left\{ \sum_{i=1}^{n} |a_{i,j}| \right\} \] (maximum column sum of absolute values) \hspace{1cm} (2.3.40)

(2) \[ \| A \|_\infty = \max_{1 \leq i \leq n} \left\{ \sum_{j=1}^{n} |a_{i,j}| \right\} \] (maximum row sum of absolute values) \hspace{1cm} (2.3.41)
(3) \[ \|A\|_2 = \left( \rho(A^H A) \right)^{1/2} \] (called spectral norm)
\[ = \left\{ \text{maximum eigenvalue of } A^H A \right\}^{1/2} \] \hspace{1cm} (2.3.42)

(4) \[ \|A\|_E = \left\{ \sum_{i=1}^{n} \sum_{j=1}^{n} |a_{i,j}|^2 \right\}^{1/2} \] (Euclidian Matrix norm) \hspace{1cm} (2.3.43)

**Bound for Spectral Radius:**

If
\[ A x_i = \lambda_i x_i \]
then
\[ |\lambda_i| \|x_i\| = \|Ax_i\| \]
\[ \leq \|A\| \|x_i\| \text{, for any compatible norm.} \]

Thus
\[ |\lambda_i| \leq \|A\| \]
and hence
\[ \rho(A) = \max_{1 \leq i \leq n} |\lambda_i| \leq \|A\| \text{, for any compatible norm.} \] \hspace{1cm} (2.3.44)

2.3.4 **Inner Product Space.**

A linear space \( X \) is an inner product space if for every pair \( x, y \in X \) there is defined a quantity \( \langle x, y \rangle \) such that

(i) \[ \langle x+y, z \rangle = \langle x, z \rangle + \langle y, z \rangle \] \hspace{1cm} (2.3.45)

(ii) \[ \langle x, y \rangle = \overline{\langle y, x \rangle} \] (complex conjugate) \hspace{1cm} (2.3.46)

(iii) \[ \langle \alpha x, y \rangle = \overline{\alpha} \langle x, y \rangle \] \( \alpha \) is any scalar, Note; \( \langle x, ay \rangle = \overline{\alpha} \langle x, y \rangle \) \hspace{1cm} (2.3.47)
(iv) \((x,x) \geq 0\), with equality if and only if \(x = 0\). (2.3.48)

(In a real inner product space \((x,y)\) is real and axiom (ii) becomes \((x,y) = (y,x)\), and \(a\) is a real number in axiom (iii)).

An inner product space is a normed linear space with norm

\[ \|x\| = (x,x)^{1/2} \]  

(2.3.49)

**Definition (2.3.9).** A set of elements \(S\) in an inner product space \(X\) is complete if

\[ (y,x) = 0 \quad \text{for all } x \in S \]  

implies \(y = 0\).

2.3.5 **Hilbert Space.**

A complete inner product space will be called a Hilbert space, \(H\), if the following additional requirements are fulfilled.

(i) \(H\) is infinite dimensional; that is, given any integer \(n\), we can find \(n\) independent elements.

(ii) There is a complete sequence of elements in \(H\).

There are many known examples of Hilbert spaces such as:

(1) The set of all infinite sequence \(\{a_i\}\) for which

\[ \sum_{i=1}^{\infty} |a_i|^2 < \infty \]

augmented by the usual definitions for addition and scalar products and by
\[(a,b) = \sum_{i=1}^{\infty} a_i \bar{b}_i, \quad a = \{a_i\}, \quad b = \{b_i\}\]

as the definition of an inner product, constitutes a Hilbert space. It is called the sequential Hilbert space and is designated by $\ell^2$.

(2) $L^2[a,b]$ is a Hilbert space consisting of all functions defined on $[a,b]$ which are measurable and for which $|f(x)|^2$ is integrable. The inner product is defined by

\[ (f,g) = \int_a^b f(x) \overline{g(x)} \, dx \]

and its norm by

\[ ||f||^2 = (f,f) = \int_a^b |f(x)|^2 \, dx. \]

(3) Let $B$ be a region. The set of function $f(z)$ which are single valued and analytic in $B$ and for which

\[ \iint_B |f(z)|^2 \, dxdy < \infty \]

will be designated by $L^2(B)$. It can be shown that $L^2(B)$ contains a complete sequence of functions, see for example [DAVIS, 1975].

If $B$ is bounded, or can be mapped 1-1 conformally onto a bounded region, then $L^2(B)$ is a Hilbert space.

2.4 NUMERICAL METHODS FOR SOLVING PARTIAL DIFFERENTIAL EQUATIONS.

There are many numerical methods for solving boundary value problems of elliptic type. The main numerical approaches for
solving such problems fall, in general, in one of the following categories:

(i) **Semi-analytical methods**, which are used to obtain a function which approximates the solution of the boundary value problem at any point of the region in which the solution is sought.

(ii) **Finite difference method**, which is the most widely used method for solving partial differential equations. In this method, the differential equation is replaced by an approximate difference formula. This difference formula is then used to seek an approximated solution of the problem at discrete points of the region.

(iii) **Finite element method**. This method provides a global approximation based on very simple local representations. Thus the domain of a problem is partitioned into a large number of small subdomains which are the "elements". The local representation is then the form of the approximate solution in an individual element.

We shall now consider some different methods of the above categories in more detail.

2.4.1 **Semi-Analytical Methods.**

In semi-analytical methods we define an approximation to the solution of the boundary value problem in question, of the form

\[ u^* = \sum_{i=1}^{n} a_i \psi_i(x,y) \]  \hspace{1cm} (2.4.1)

where \( \{\psi_i(x,y)\} \) is a set of pre-defined functions. We aim
to evaluate the unknown coefficients $a_i$ in (2.4.1) such that $u^*$ is the best approximation to the solution of the problem. This can be done for example, by the least squares method, collocation methods, Sigillito's method, etc. The selection of the functions $\psi_i(x,y)$ in (2.4.1) is closely related to the approximation methods employed to evaluate the unknown $a_i$. In the following we shall describe in detail some of these semi-analytical methods.

Sigillito's Method.

The method of Sigillito is based on an a priori inequality in 2-dimensional space such as:

$$
\iint_B u^2 \, dx \, dy \leq \alpha_1 \iint_B (\nabla u)^2 \, dx \, dy + \alpha_2 \int_{\partial B} u^2 \, ds,
$$

(2.4.2)

for computing an approximate solution, with error bounds, for Dirichlet problems. [SIGILLITO, 1977]. It is important to mention here that the trial functions which are used in this method need not satisfy either the differential equation or the boundary conditions.

It should be noted that the a priori inequality gives a practical method for determining a posteriori error estimation for any sufficiently smooth approximate solution. It should also be noted, however, that to apply the method, all constants which appear in the inequality (i.e. $\alpha_1$, $\alpha_2$) must be known explicitly or be computable.
Theorem (2.4.1): Let $u$ be a function with piecewise continuous second order derivatives in $\Omega$, but otherwise arbitrary, then it can be seen that the inequality (2.4.2) is true where $\alpha_1$ and $\alpha_2$ are explicitly determined constants. [BRAMBLE & PAYNE, 1963].

Having the explicit inequality (2.4.2) we wish to approximate the solution of the Dirichlet problem:

$$\nabla^2 u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f(x,y) \quad \text{in } \Omega$$

where

$$u = g(x,y) \quad \text{on } \partial \Omega \quad (2.4.3)$$

Let an approximate solution of $u$ be

$$u^* = \sum_{k=1}^{n} a_k \psi_k \quad (2.4.4)$$

where, because of the a'priori nature of (2.4.2) the trial functions $\psi_k$ need not satisfy either the boundary condition or the differential equation, but need only be $C^2(\Omega)$ (i.e. twice differentiable in $\Omega$). Now assume the exact solution of (2.4.3) to be $u$. Then $u-u^*$ can also be seen to be twice differentiable in $\Omega$ (i.e. $u-u^* \in C^2(\Omega)$) and we can, therefore now substitute this function in (2.4.2) to obtain the following inequality:

$$\int\int_{\Omega} (u^*-u^*)^2 \, dx \, dy \leq \alpha_1 \int\int_{\Omega} (\nabla^2 u - \nabla^2 u^*)^2 \, dx \, dy + \alpha_2 \int_{\partial \Omega} (u^*-u^*) \, ds$$

$$= \alpha_1 \int\int_{\Omega} (f - \nabla \cdot \sum_{k=1}^{n} a_k \psi_k)^2 \, dx \, dy + \alpha_2 \int_{\partial \Omega} (g - \sum_{k=1}^{n} a_k \psi_k)^2 \, ds$$

$$\quad (2.4.5)$$
which gives an $L_2$ norm bound on the error of the approximation.

Since $f$, $g$ and $\psi_k$, $k=1,\ldots,n$ are known functions, the right hand side of (2.4.5) is a function of the coefficients $a_1,\ldots,a_n$, $E(a_1,\ldots,a_n)$ say. Then we seek $a_k$ such that (2.4.4) is a good approximation to the solution $u$ of boundary value problem of (2.4.3). For this we determine the unknowns $a_k$, $k=1,\ldots,n$, such that the bound $E(a_1,\ldots,a_n)$ is as small as possible. To find its minimum value we set the partial derivatives of $E$ with respect to all $a_k$ equal to zero. We thus obtain the following system of $n$ linear equations in $n$ unknowns:

\[
\frac{\partial E}{\partial a_i} = 2\alpha_1 \int_B \left( f - \nabla^2 \sum_{k=1}^n a_k \psi_k \right) \nabla^2 \psi_i \, dxdy + \\
+ 2\alpha_2 \int_{\partial B} \left( g - \sum_{k=1}^n a_k \psi_k \right) \psi_i \, dS = 0, \quad i=1,\ldots,n.
\]

Also from (2.4.5) we have the computable error bound

\[
\int_B (u - u^*)^2 \, dxdy \leq \alpha_1 \int_B f^2 \, dxdy + \alpha_2 \int_{\partial B} g^2 \, dS \\
- 2 \sum_{k=1}^n a_k \left\{ \alpha_1 \int_B f \nabla^2 \psi_k \, dxdy + \alpha_2 \int_{\partial B} g \psi_k \, dS \right\} \\
+ \sum_{k=1}^n \sum_{i=1}^n a_k a_i \left\{ \alpha_1 \int_B (\nabla^2 \psi_k) (\nabla^2 \psi_i) \, dxdy + \alpha_2 \int_{\partial B} \psi_k \psi_i \, dS \right\}
\]

(2.4.7)
where terms like
\[
\int_B (\nabla^2 \psi_k) (\nabla^2 \psi_i) \, dx\, dy, \quad \int_{\partial B} \psi_k \psi_i \, dS, \quad \int_B f \nabla^2 \psi_k \, dx \, dy
\]
and \( \int_{\partial B} g \psi_k \, dS \) in (2.4.7) would have already been computed for the system (2.4.6). Then, if we assume that \( a_1^*, \ldots, a_n^* \) is the solution of the system (2.4.6) we can therefore obtain the approximate solution
\[
u^* = \sum_{k=1}^n a_k^* \psi_k
\]
which is the best approximation to \( u \) in the sense that the error bound \( E(a_1^*, \ldots, a_n^*) \) on
\[
\iint_B (u-u^*)^2 \, dx \, dy
\]
is a minimum.

This method requires the user to supply a set of trial functions from which the approximating function is formed. This presents the opportunity to incorporate known information. If a good choice has been made for the trial functions a very accurate approximation is possible. On the other hand, a bad choice of trial functions will lead to a poor approximation. Thus a discussion of the methods for selecting trial functions is a necessary part of any presentation of this method. However, some general comments on trial function selections are possible. For instance, one could use trial functions suggested by the boundary conditions or the differential equation or by the solution of special cases of the problem, which can be
solved exactly. See [FINLAYSON, 1972] for further discussions along these lines. Typical trial functions will be subsets of the following complete sets: productions of sines and cosines, products of powers of $x$ and $y$, polynomials which satisfy the differential equation etc.

**Least Squares Method for Solving Boundary Value Problems of Elliptic Type.**

Let $\mathcal{L}$ be a linear operator defined for some linear set $D_\mathcal{L}$ which is dense in a given Hilbert space $H$. Consider the linear equation

$$\mathcal{L}u = f,$$  \hspace{1cm} (2.4.8)

where $f$ is a given element from $H$. The objective here is to solve the above system by the least squares method. For this, we choose a sequence of linearly independent elements $\{\phi_i | \phi_i \in D_\mathcal{L}\}$, and the approximation solution is represented in the form

$$u^* = \sum_{i=1}^{n} a_i \phi_i,$$  \hspace{1cm} (2.4.9)

where the unknown constants $a_i$, $i=1,2,\ldots,n$ are determined such that

$$\| \mathcal{L}u^* - f \|^2$$  \hspace{1cm} (2.4.10)

is a minimum. It can be shown, see [MIKLIN, 1964], that the above minimization gives rise to a system of linear equations of the form

$$\sum_{i=1}^{n} a_i (\mathcal{L}\phi_i, \mathcal{L}\phi_k) = (f, \mathcal{L}\phi_k) \hspace{1cm} k=1,2,\ldots,n$$  \hspace{1cm} (2.4.11)

with unknowns $a_1, a_2, \ldots, a_n$ where $(\cdot, \cdot)$ defines the inner product.
The numerical success of the method of least squares depends upon selecting the approximate solution from a complete system of functions.

As an application of the least squares method for solving boundary value problems we consider the Laplace equation:

\[ \nabla^2 u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \quad \text{in } \mathbb{R} \quad (2.4.12) \]

with

\[ u = g(x,y) \quad \text{on } \partial \mathbb{R} \quad (2.4.13) \]

where \( \mathbb{R} \) is a connected two dimensional region with boundary \( \partial \mathbb{R} \).

To obtain the approximate solution to the above problem, we select a sequence of linearly independent and harmonic functions \( \psi_k \), i.e. \( \nabla^2 \psi_k = 0 \) and define a linear combination of the form:

\[ u^* = \sum_{k=1}^{n} a_k \psi_k \quad (2.4.14) \]

to be an approximate solution of (2.4.12). Then the values of \( a_1, a_2, \ldots, a_n \) are determined such that

\[ I = \int_{\partial \mathbb{R}} \left[ u^* - g \right]^2 \, dS \]

\[ = \int_{\partial \mathbb{R}} \left[ \sum_{k=1}^{n} a_k \psi_k - g \right]^2 \, dS \quad (2.4.15) \]

is a minimum. The minimum of equation (2.4.15) is given when

\[ \frac{\partial I}{\partial a_k} = 0 , \quad k=1,2,\ldots,n \quad (2.4.16) \]
which leads to a system of $n$ linear equations in the unknowns $a_k$.

It should be noted here that the function $\psi_k$, $k=1,2,\ldots,n$ are chosen such that they do not satisfy the boundary condition (i.e. $I \neq 0$). For the selection of the functions $\psi_k$ we use the fact that the real and imaginary parts of an analytic function are harmonic (from the Cauchy-Riemann equation). Thus, for Laplace's equation on $R$, the harmonic polynomials

$$\begin{cases}
  \text{Re}(z^n) & n = 0,1,\ldots \\
  \text{Im}(z^n) & n = 1,2,\ldots
\end{cases} \quad (2.4.17)$$

where $z = x + iy$, form a complete system. See [DAVIS & RABINOWITZ, 1961]

The approximate solution $u^*$ is now defined as

$$u^*(x,y) = \sum_{k=1}^{2N+1} a_k \psi_k(x,y) \quad (2.4.18)$$

where;

$$\psi_k(x,y) = \begin{cases}
  \text{Re}(z^{k/2}) & \text{for } k \text{ even} \\
  \text{Im}(z^{(k-1)/2}) & \text{for } k \text{ odd} \\
  1 & \text{for } k=1
\end{cases} \quad (2.4.19)$$

As mentioned before, one advantage of using semi-analytical methods for solving elliptic differential equations is that simple error estimates became available once the computation has been performed. As an example, we consider the error bounds for Laplace's equation when solved by the method of least squares. For this we have the following maximum principle theorem.
Theorem (2.4.2): A non-constant harmonic function attains its maximum and minimum on the boundary of the region. For proof of this theorem and a full discussion about maximum principles for a wide class of elliptic equations see [PROTTER & WEINBERGER, 1967].

Now let $u$ be an analytic solution of (2.4.12) with boundary (2.4.13) and $u^*$ be a combination of harmonic functions which are to approximate $u$. Then $u - u^*$ will also be harmonic and therefore, the maximum principle allows us to determine the error $u - u^*$ throughout the interior of domain, since the maximum error is attained on the boundary.

2.4.2 Finite Difference Methods:

Finite difference methods provide a powerful tool for the solution of many problems involving differential equations. The domain of the independent variables are replaced by a finite set of points, usually referred to as mesh points, and at each mesh point we find an approximation to the differential problem in terms of the function values at other points. The values at the mesh points are required to satisfy difference equations obtained by replacing partial derivatives by partial difference quotients or by certain other more sophisticated techniques. We restrict our attention here to the elliptic differential problems and describe two most common ways to discretize such problems. For this we consider elliptic problems of the type in which the function $u(x,y)$ satisfies:
\[ \mathcal{L}[u(x,y)] = f(x,y) \quad (x,y) \in \Omega \] (2.4.20)

and

\[ \mathcal{M}_h[u(x,y)] = g(x,y) \quad (x,y) \in \partial \Omega \] (2.4.21)

where \( \Omega \subseteq \mathbb{R}^2 \) is a simply connected open bounded domain with boundary \( \partial \Omega \). We now define \( \overline{\Omega} = \Omega \cup \partial \Omega \).

Suppose that the region \( \overline{\Omega} \) is subdivided by two groups of parallel lines into a square net. That is, we replace the point set \( \overline{\Omega} \) by the point set \( \overline{\Omega}_h = \Omega_h \cup \partial \Omega_h \) by taking any point \( (\overline{x}, \overline{y}) \in \Omega \) and constructing the lines \( x = \overline{x} + ih \) and \( y = \overline{y} + jh \), \( i, j = 0, \pm 1, \pm 2, \ldots \), where \( h \) is any positive number, which we call the mesh size. The intersection of the lines are mesh points with co-ordinates \( (\overline{x} + ih, \overline{y} + jh) \).

(Fig. 2.4.1)

The sets of mesh points are:

\[ \overline{\Omega}_h \equiv \text{set of all mesh points in } \overline{\Omega}. \] This does not include points on the boundary unless they are boundary mesh points.

\[ \Omega_h \equiv \text{set of all mesh point in } \Omega \text{ such that the four adjacent mesh points are in } \overline{\Omega} \text{ and the intervening segments are in } \Omega. \] Points
of $\Omega_h$ are indicated by $\mathbf{X}$ in (Fig. 2.4.1).

\[ \Omega_h \equiv \overline{\Omega}_h - \Omega_h \] = boundary mesh points plus some interior mesh points.

(1)

\[ \Omega_h^0 \equiv \text{set of "ACCESSIBLE" points of } \Omega_h; \ i.e. \ the \ set \ of \ points \ of \ \Omega_h \ which \ are \ adjacent \ to \ a \ point \ in \ \Omega_h, \ \Omega_h^{(1)} \subseteq \Omega_h. \]

The points in (Fig. 2.4.1) which are indicated by $\bullet$ belong to $\Omega_h^{(1)}$.

\[ \Omega_h^{(0)} \equiv \Omega_h - \Omega_h^{(1)} \equiv \text{set of "INACCESSIBLE" points, which are indicated by } 0 \text{ in (Fig. 2.4.1).} \]

Now, we sketch briefly methods for discretizing the elliptic problem (2.4.20).

Taylor's Series Method.

The simplest and most direct method for discretizing (2.4.20) is to approximate the various derivatives $\frac{\partial u}{\partial x}$, $\frac{\partial u}{\partial y}$, $\frac{\partial^2 u}{\partial x^2}$, etc., in terms of differences between the function values $u(x_i, y_j) = u_{ij}$ at adjacent points of the mesh. Then, by substitution of approximated derivatives into the differential equation, we obtain an expression for $u_{ij}$ at each point of the $\overline{\Omega}_h$. To find the difference formulae for the derivatives of $u$, we use the finite Taylor's series expansion in $u$. The validity of the approximations for $u_{ij}$ depends, therefore, upon the existence of partial derivatives of $u$ of sufficiently high order.

To simplify matters we consider the linear second order elliptic problem.
\[ A \frac{\partial^2 u}{\partial x^2} + C \frac{\partial^2 u}{\partial y^2} + D \frac{\partial u}{\partial x} + E \frac{\partial u}{\partial y} + F u = G, \quad \text{in } \Omega \quad (2.4.22) \]

\[ u(x,y) = g(x,y) \quad \text{on } \partial \Omega, \quad (2.4.23) \]

where \( A = A(x,y), \ C = C(x,y), \) etc., and \( A > 0, \ C > 0 \) and \( F \leq 0. \)

Assume that \( u(x,y) \) has partial derivatives of the fourth order in a sufficiently large neighbourhood about a point \( P \equiv (x,y) \in \Omega, \)

taking four adjacent mesh points of \( P \) as shown in (Fig. 2.4.2).

\[ \begin{align*}
\text{NN} & \\
\text{N} & \text{P} \quad \text{E} \\
\text{WW} & \\
\text{W} & \text{P} \quad \text{E} \\
\text{S} & \\
\text{SS} & \\
\end{align*} \]

(Fig. 2.4.2)

where \( E \equiv (x+h,y), \ W \equiv (x-h,y), \ N \equiv (x,y+h) \) and \( S \equiv (x,y-h), \)
then by Taylor's theorem we have:

\[ u(E) = u(x+h,y) = u(P) + h \frac{u}{u} + \frac{h^2}{2!} u_{xx} + \frac{h^3}{3!} u_{xxx} + \frac{h^4}{4!} u_{xxxx} (x, y+h, y) \quad (2.4.24) \]

where

\[ u_x \equiv \left. \frac{\partial u}{\partial x} \right|_P, \quad u_{xx} \equiv \left. \frac{\partial^2 u}{\partial x^2} \right|_P, \text{etc.}, \text{ and } 0 < \xi_1 < 1. \]

Similarly for \( 0 < \xi_2 < 1 \) we have

\[ u(W) = u(x-h,y) = u(P) - h \frac{u}{u} + \frac{h^2}{2!} u_{xx} - \frac{h^3}{3!} u_{xxx} + \frac{h^4}{4!} u_{xxxx} (x-h, y, y) \quad (2.4.25) \]
Combining (2.4.24) and (2.4.25), we have
\[ \frac{\partial^2 u}{\partial x^2} \bigg|_p = u_{xx} = \frac{u(E) - 2u(P) + u(W)}{h^2} - \frac{h^2}{4!} \left[ u_{xxxx} (x + \xi_1 h, y) + u_{xxxx} (x - \xi_2 h, y) \right] . \]  
(2.4.26)

Similarly using Taylor's expansion, we obtain
\[ u(N) = u(x, y+h) = u(P) + hu_y + \frac{h^2}{2!} u_{yy} + \frac{h^3}{3!} u_{yyy} + \frac{h^4}{4!} u_{yyyy} (x, y+\eta_1 h) \]
(2.4.27)

where \[ 0 < \eta_1 < 1 \]
and
\[ u(S) = u(x, y-h) = u(P) - hu_y + \frac{h^2}{2!} u_{yy} - \frac{h^3}{3!} u_{yyy} + \frac{h^4}{4!} u_{yyyy} (x, y-\eta_2 h) \]
(2.4.28)

where \[ 0 < \eta_2 < 1 \] .

Then, we have
\[ \frac{\partial^2 u}{\partial y^2} \bigg|_p = u_{yy} = \frac{u(N) - 2u(P) + u(S)}{h^2} - \frac{h^2}{4!} \left[ u_{yyyy} (x, y+\eta_1 h) + u_{yyyy} (x, y-\eta_2 h) \right] . \]  
(2.4.29)

Using the same method, we can also obtain the following:
\[ \frac{\partial u}{\partial x} \bigg|_p = u_x = \frac{u(E) - u(W)}{2h} - \frac{h^2}{12} \left[ u_{xxx} (x + \xi_3 h, y) + u_{xxx} (x - \xi_4 h, y) \right] \]  
(2.4.30)

and
\[ \frac{\partial u}{\partial y} \bigg|_p = u_y = \frac{u(N) - u(S)}{2h} - \frac{h^2}{12} \left[ u_{yyy} (x, y+\eta_3 h) + u_{yyy} (x, y-\eta_4 h) \right] \]  
(2.4.31)

where \[ 0 < \xi_3, \xi_4, \eta_3, \eta_4 < 1 \] .
Then on substitution of the above approximations (2.4.26), (2.4.29), (2.4.30) and (2.4.31), neglecting terms of order $h^2$ and higher power, into (2.4.22) the following difference equation corresponding to (2.4.22) at the point $P = (x,y)$ in terms of $u(x,y)$ and functional values of the four neighbourhood points is obtained

$$(A + \frac{1}{2} Dh)u(x+h,y) + (A - \frac{1}{2} Dh)u(x-h,y) + (C + \frac{1}{2} Eh)u(x,y+h)$$

$$+ (C - \frac{1}{2} Eh)u(x,y-h) - 2(A + C - \frac{1}{2} Fh^2)u(x,y) = G h^2$$

(2.4.32)

where of course $A$, $C$, $D$, $E$, $F$ and $G$ are all evaluated at the point $(x,y)$.

Rewriting (2.4.32) in the form

$$c_1 u(x+h,y) + c_3 u(x-h,y) + c_2 u(x,y+h) + c_4 u(x,y-h)$$

$$- c_0 u(x,y) = t(x,y)$$

(2.4.33)

where $c_i$, $i = 0,1,2,3,4$ are functions of $x$ and $y$, it can be seen that

$$c_0 > c_1 + c_2 + c_3 + c_4 ,$$

(2.4.34)

because $A > 0$, $C > 0$, and $F < 0$.

In problems where the boundary $\partial \Omega$ is a curve, we have a set of mesh points $\partial \Omega_h^{(1)}$ which are not exactly distance $h$ away from the adjacent points, and therefore, difference equations derived from a non-square mesh will be used to treat such mesh points. In the following we shall derive difference equations for problem (2.4.22) - (2.4.23), taking the neighbouring points of $P = (x,y)$ as follows:
which are depicted in (Fig. 2.4.3).

\[ E \equiv (x+S_1 h, y), \quad W \equiv (x-S_3 h, y), \quad N \equiv (x,y+S_2 h) \text{ and } S \equiv (x,y-S_4 h) \]

where \( 0 < S_i < 1, \ i = 1,2,3,4 \). However, Taylor series expansions similar to (2.4.24) - (2.4.28) can still be used and the following finite difference approximations are obtained,

\[
\frac{\partial u}{\partial x} \bigg|_p \equiv u_x = \frac{1}{h} \left[ \frac{S_2}{S_1(S_1+S_3)} \ u(E) - \frac{S_1}{S_3(S_1+S_3)} \ u(W) - \frac{S_3-S_1}{S_1S_3} \ u(P) \right] + O(h^2)
\]

(2.4.35)

and

\[
\frac{\partial^2 u}{\partial x^2} \bigg|_p \equiv u_{xx} = \frac{2}{h^2} \left[ \frac{1}{S_1(S_1+S_3)} \ u(E) + \frac{1}{S_3(S_1+S_3)} \ u(W) - \frac{1}{S_1S_3} \ u(P) \right] + O(h)
\]

(2.4.36)

with similar expressions for \( \frac{\partial u}{\partial y} \) and \( \frac{\partial^2 u}{\partial y^2} \).

Substitution of (2.4.35) and (2.4.36) into the differential
equation (2.4.22) result in the equation

\[ \alpha_1 u(E) + \alpha_3 u(W) + \alpha_2 u(N) + \alpha_4 u(S) - \alpha_0 u(P) = t(x,y) \]  

\[ (2.4.37) \]

where now

\[ \alpha_1 = \frac{2A+hS_3D}{S_1(S_1+S_3)} \], \[ \alpha_3 = \frac{2A-hS_3D}{S_3(S_1+S_3)} \], \[ \alpha_2 = \frac{2C+hS_4E}{S_2(S_2+S_4)} \], \[ \alpha_4 = \frac{2C-hS_2E}{S_4(S_2+S_4)} \]

\[ \alpha_0 = \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 - Fh^2 \text{, and } t = G h^2 \]. \[ (2.4.38) \]

Every \( \alpha_i \) appearing in (2.4.38) and in the equation (2.4.33), (i.e. square mesh) will be positive provided \( h \) is chosen small enough to satisfy

\[ h < \min \left\{ \frac{2A}{|D|}, \frac{2C}{|E|} \right\}. \] \[ (2.4.39) \]

The minimum has taken over all points of \( \Omega \cup \partial \Omega \). Since \( A > 0 \), \( C > 0 \), and \( A, C, D, E \) are continuous in the closed region \( \Omega \cup \partial \Omega \), therefore they are bounded. Consequently a positive minimum exists. See [YOUNG, 1962].

As an example, we consider the potential problem (\( V^2 \) is denoted by \( \Delta \))

\[ \Delta[u(x,y)] = 0 \quad , \quad (x,y) \in \Omega \]  

\[ (2.4.40) \]

and

\[ u(x,y) = g(x,y) \quad (x,y) \in \partial \Omega \]  

\[ (2.4.41) \]

and at points of \( \Omega_h \) we can replace \( \Delta[u(x,y)] \) by \( \Delta_h[u(x,y)] \) which by using (2.4.33) we have

\[ \Delta_h[u(x,y)] = \frac{1}{h^2} \left[ u(N) + u(S) + u(E) + u(W) - 4u(P) \right] \]  

\[ (2.4.42) \]
with an error $O(h^2)$, as $h \to 0$. Using (2.4.42) we set up the difference equations

$$\Delta_h[u(P)] = 0, \quad \forall \, P \in \Omega_h \quad (2.4.43)$$

and if we apply (2.4.43) for all points $P \in \partial \Omega_h \cap \Omega_h$, then we obtain a linear system,

$$A \, u_h = b \quad (2.4.44)$$

we clearly need a technique for dealing with the points $\partial \Omega_h$. A possibility is to use irregular nets which are dealt with later.

We shall now attempt to evaluate an error bound for the solution, using Gershgorin's technique to prove that:

$$|u(P) - u_h(P)| < k \, h^\alpha, \quad \forall \, P \in \Omega \cap \overline{\Omega}_h. \quad (2.4.45)$$

**Lemma (2.4.1).**

If $u(x,y)$ has bounded derivatives of all orders up to and including the 4th in $\overline{\Omega}$, then

$$|\Delta_h[u(x,y)] - \Delta[u(x,y)]| \leq \frac{h^2}{6} M_4, \quad \forall \, (x,y) \in \Omega_h \quad (2.4.46)$$

where $u(x,y)$ is the solution of problem (2.4.40) - (2.4.41) and

$$M_k = \max \left\{ \max_{(x,y) \in \Omega} \left| \frac{\partial^k u(x,y)}{\partial x^k} \right|, \max_{(x,y) \in \overline{\Omega}} \left| \frac{\partial^k u(x,y)}{\partial y^k} \right| \right\}$$

**Proof.**

We again use Taylor's series expansions (2.4.26) and (2.4.29) so that we have
\[ u(E) - 2u(P) + u(W) = h^2 u_{xx} + \frac{h^4}{24} \left[ u_{xxxx}(x+\xi_1 h, y) + u_{xxxx}(x-\xi_2 h, y) \right] \]

\[ = h^2 u_{xx} + \frac{h^4}{12} \left[ u_{xxxx}(x+\theta h, y) \right] \quad (2.4.47) \]

where

\[-1 < \theta < 1\]

and

\[ u(N) - 2u(P) + u(S) = h^2 u_{yy} + \frac{h^4}{24} \left[ u_{yyyy}(x,y+\eta_1 h) + u_{yyyy}(x,y-\eta_2 h) \right] \]

\[ = h^2 u_{yy} + \frac{h^4}{12} \left[ u_{yyyy}(x,y+\phi h) \right] \quad (2.4.48) \]

where

\[-1 < \phi < 1 \]

Combining (2.4.47) and (2.4.48) and by (2.4.42) we obtain

\[ h^2 \Delta_h \left[ u(x,y) \right] = h^2 (u_{xx} + u_{yy}) + \frac{h^4}{12} \left[ u_{xxxx}(x+\theta h, y) + u_{yyyy}(x,y+\phi h) \right] \]

\[ = h^2 (u_{xx} + u_{yy}) + \frac{h^4}{12} \left[ u_{xxxx}(x+\theta h, y) + u_{yyyy}(x,y+\phi h) \right] \quad (2.4.49) \]

Now, assume that

\[ \max_{(x,y) \in \Omega} u_{xxxx}(x+\theta h, y) = M_{14} \quad \max_{(x,y) \in \Omega} u_{yyyy}(x,y+\phi h) = M_{24} \]

and \[ \max(M_{14}, M_{24}) = M_4 \]

and the fact that \( (u_{xx} + u_{yy})_p = \Delta u_p(x,y) \) the result (2.4.46) follows.

**Lemma (2.4.2).**

If \( \Delta_h[u(x,y)] \leq 0 \) in \( \Omega_h \) and \( u \geq 0 \) on \( \partial \Omega_h^{(1)} \),

then \( u \geq 0 \) on \( \Omega_h \cup \partial \Omega_h^{(1)} \).
Suppose \( u(Q) < 0 \) for a point \( Q \in \Omega_h \), then at some point \( P \in \Omega_h \) the function \( u \) has a minimum value, say \( -M < 0 \). If the neighbouring points \( N, W, S \) and \( E \) of \( P \) are as usual, which are shown in (Fig. 2.4.2) then since \( \Delta_h u \leq 0 \), we have

\[
(2.4.50) \quad u(N) + u(S) + u(W) + u(E) - 4u(P) \leq 0
\]

thus

\[
(2.4.51) \quad u(P) = -M \geq \frac{1}{4} (u(N) + u(S) + u(W) + u(E)).
\]

But the minimum value of \( u \) on \( \Omega_h \cup \partial \Omega_h(1) \) occurs at \( P \), so that \( u(i) \geq u(P), \ i = N, S, W, E \), thus unless \( u(i) = u(P) \) the right hand side (2.4.51) is greater than \( -M \) and we have a contradiction. Therefore \( u(i) = -M \) for all points adjacent to \( P \). In the same way we can show that \( u(Q) = -M \) for all \( Q \in \Omega_h \cup \partial \Omega_h(1) \). This implies that \( u = -M < 0 \) at points of \( \partial \Omega_h(1) \), which contradicts the original hypothesis that \( u(Q) \geq 0 \) where \( Q \in \partial \Omega_h(1) \). Thus we have that \( u(P) \geq 0 \) for all \( P \in \Omega_h \cup \partial \Omega_h(1) \).

Lemma (2.4.3).

If \( |u| \leq v \) on \( \partial \Omega_h(1) \) and \( |\Delta_h u| \leq -\Delta_h v \) on \( \Omega_h \), then \( |u| \leq v \) on \( \Omega_h \cup \partial \Omega_h(1) \), where \( \Delta_h v \leq 0 \) on \( \Omega_h \) and \( v \geq 0 \) on \( \partial \Omega_h(1) \).

Proof.

By Lemma (2.4.2) we have \( v \geq 0 \) on \( \Omega_h \cup \partial \Omega_h(1) \), we must show that \( v \geq |u| \) on \( \Omega_h \cup \partial \Omega_h(1) \). First let \( w = v - u \), since \( v \geq |u| \) on \( \partial \Omega_h(1) \) and \( v \geq 0 \) on \( \partial \Omega_h(1) \), then \( v - u \geq v - |u| \geq 0 \) on \( \partial \Omega_h(1) \) implies
\( w \gg 0 \) on \( \Omega_h^{(1)} \). Also

\[
\Delta_h \left[ w \right] = \Delta_h \left[ v-u \right] = \Delta_h \left[ v \right] - \Delta_h \left[ u \right]
\]

\( \leq \Delta_h \left[ v \right] + \left| \Delta_h \left[ u \right] \right| \leq 0 \quad (2.4.52) \)

By Lemma (2.4.2), we have \( w \gg 0 \) on \( \Omega_h \cup \Omega_h^{(1)} \), i.e. \( v \gg u \) on \( \Omega_h \cup \Omega_h^{(1)} \). To prove that \( v \gg -u \), let \( u_2 = -u \), \( w_2 = v-u_2 \). Since \( v \gg |u| \) on \( \Omega_h^{(1)} \), we have that \( v \gg u_2 \) on \( \Omega_h^{(1)} \), also

\[
\Delta_h \left[ w_2 \right] = \Delta_h \left[ v-u_2 \right] = \Delta_h \left[ v \right] - \Delta_h \left[ u_2 \right]
\]

\( \leq \Delta_h \left[ v \right] + \left| \Delta_h \left[ u_2 \right] \right| = \Delta_h \left[ v \right] + \left| \Delta_h \left[ u \right] \right| \leq 0 \quad (2.4.53) \)

Thus \( w_2 \gg 0 \) on \( \Omega_h^{(1)} \) and \( \Delta_h \left[ w_2 \right] \leq 0 \) on \( \Omega_h \) so that by Lemma (2.4.2) \( w_2 \gg 0 \) on \( \Omega_h \cup \Omega_h^{(1)} \), i.e. \( v-u_2 \gg 0 \) or \( v+u \gg 0 \) on \( \Omega_h \cup \Omega_h^{(1)} \).

**Theorem (2.4.3).** If the exact solution \( u(x,y) \) of the Dirichlet problem (2.4.40) with boundary condition (2.4.41) has bounded derivatives of all orders up to and including the 4th on \( \bar{\Omega} \), and if \( r \) is the radius of a circle containing \( \bar{\Omega} \), then for all \( (x,y) \in \Omega_h \cup \Omega_h^{(1)} \)

\[
|u(x,y) - u_h(x,y)| \leq \frac{h^2 r^2 M_4}{24} + \max_{(x,y) \in \Omega_h^{(1)}} |u(x,y) - u_h(x,y)|
\]

(2.4.54)

where \( u_h(x,y) \) is any (mesh) function defined on \( \Omega_h \cup \Omega_h^{(1)} \) satisfying
the difference equation $\Delta_h \left[ u_h \right] = 0$.

**Proof.**

Let $\Omega$ be contained in the circle $(x-x_0)^2 + (y-y_0)^2 = r^2$ and we define the function $v$ by

$$v(x,y) = A \left\{ 1 - \frac{(x-x_0)^2 + (y-y_0)^2}{r^2} \right\} + \beta \quad (2.4.55)$$

where $A > 0$. Inside the circle we have

$$\frac{(x-x_0)^2 + (y-y_0)^2}{r^2} < 1,$$

so that

$$v(x,y) \geq \beta, \quad (x,y) \in \Omega_h \quad (2.4.56)$$

the equality occurs if and only if the circle passes through a point of $\Omega$ which is also a point of $\Omega_h$.

Now, for $(x,y) \in \Omega_h$ we have

$$\Delta_h \left[ v(x,y) \right] = \frac{1}{h^2} \left[ v(x+h,y) + v(x-h,y) + v(x,y+h) + v(x,y-h) - 4v(x,y) \right]$$

$$= - \frac{A}{h^2} \frac{1}{r^2} \left[ (x-x_0+h)^2 + (y-y_0)^2 + (x-x_0-h)^2 + (y-y_0)^2 + (x-x_0)^2 + (y-y_0+h)^2 + (x-x_0)^2 + (y-y_0-h)^2 - 4(x-x_0)^2 \right]$$

$$- 4(y-y_0)^2 \right]$$
thus
\[ \Delta_h [v(x,y)] = -\frac{4A}{r^2}, \quad \forall (x,y) \in \Omega_h. \quad (2.4.57) \]

If we take \( A = \frac{h^2 \tau M_4}{24} \), then from (2.4.57) we obtain
\[ -\Delta_h [v(x,y)] = \frac{h^2 M_4}{6}, \quad \forall (x,y) \in \Omega_h. \quad (2.4.58) \]

Now, from Lemma (2.4.1) we have that
\[ |\Delta_h [u(x,y)] - \Delta [u(x,y)]| \leq 64 \frac{h^2 M_4}{6} = -\Delta_h [v(x,y)] , \]
and so
\[ |\Delta_h [u] - \Delta_h [u_h] + \Delta_h [u_h] - \Delta [u]| \leq -\Delta_h [v] \text{ on } \Omega_h \]
thus
\[ |\Delta_h [u] - \Delta_h [u_h]| \leq |\Delta_h [u_h] - \Delta [u]| - \Delta_h [v] \text{ on } \Omega_h. \quad (2.4.59) \]

But we have that \( \Delta [u] = 0 \) and \( \Delta_h [u_h] = 0 \) therefore we have
\[ |\Delta_h [u] - \Delta_h [u_h]| \leq -\Delta_h [v] \text{ on } \Omega_h. \quad (2.4.60) \]

Let
\[ \beta = \max_{(x,y) \in \partial \Omega_h} |u(x,y) - u_h(x,y)|, \quad (2.4.61) \]
then by Lemma (2.4.3) since
\[ |\Delta_h [u-u_h]| \leq -\Delta_h [v] \text{ on } \Omega_h \]
and
\[ v \geq \beta = \max |u-u_h| \quad \text{on } \partial \Omega_h^{(1)} \]

we have that
\[ |u-u_h| \leq \max_{(x,y)\in \Omega_h \cup \partial \Omega_h^{(1)}} (v(x,y)) = A + \beta. \]

Thus from (2.4.61) and the assumption for the value of A we have
\[ |u(x,y) - u_h(x,y)| \leq \frac{h^2}{24} r^2 M_4 + \max_{(x,y)\in \partial \Omega_h^{(1)}} |u(x,y) - u_h(x,y)| \]
\[ (2.4.62) \]
for all \((x,y) \in \Omega_h \cup \partial \Omega_h^{(1)}\).

From a purely practical standpoint we can approximate \(M_4\) in the following way:

We have
\[ u_{xx} + u_{yy} = 0 \]
so that
\[ u_{xxyy} + u_{yyyy} = 0 \]
then
\[ u_{xx} = -u_{yyyy} = -u_{xxxx}. \]

Thus, the approximation of \(u_{xxyy}\) will be used in approximating \(M_4\).

It can easily be shown that
\[ u_{xxyy} = \frac{1}{h^4} \left[ 4u(P) - 2(u(N) + u(S) + u(W) + u(S)) + u(NE) + u(NW) + u(SE) + u(SW) \right] \]
\[ (2.4.63) \]
with an $O(h^2)$ truncation error. Therefore, with (2.4.63) we obtain an $O(h^2)$ approximation to both $u_{xxxx}$ and $u_{yyyy}$ involving the nine given points. This, however, demands boundedness of the sixth derivatives of $u$ on $\Omega$ which are even more stringent conditions than those on the $(M_4)$ fourth derivatives in the original difference replacement.

The inequality (2.4.62) provides a bound on $|u(x,y) - u_h(x,y)|$ for $(x,y) \in \Omega_h \cup \Omega_h^{(1)}$ if we know the value of, or can bound

$$\max_{(x,y) \in \Omega_h^{(1)}} |u(x,y) - u_h(x,y)|.$$ 

However, this is not usually the case, and we need a bound and a technique for the set $\Omega_h$. Three possible methods for treating the boundary come immediately to mind;

i) The value of $u_h$ at all points of $\Omega_h \cap \Omega$ is taken equal to the value of $u$ at a point in the vicinity of $\Omega$.

ii) The value of $u_h(P)$, $P \in \Omega \cap \Omega_h$, is determined by the use of linear interpolation involving at least one value of $u_h = u$ at a nearby point of $\Omega$.

iii) Difference equations derived from a non-square mesh are used at all points $P \in \Omega \cap \Omega_h$. We shall now describe two of these procedures in detail.

i) For each $P \in \Omega \cap \Omega_h$ we let $u_h(P) = u(Q) = g(Q)$,
where \( Q \in \partial \Omega \) is on the same vertical line through point \( P \) such that \( PQ \) is less than \( h \). To find a bound for \( |u(P) - u_h(P)| \) we note that

\[
|u(P) - u_h(P)| = |u(P) - u_h(Q)| = |u(P) - u(Q)|. \quad (2.4.64)
\]

But by Taylor's series we have

\[
u(Q) = u(P) + \overline{PQ} \left( \frac{\partial u}{\partial y} \right)_P + \left( \frac{\overline{PQ}^2}{2!} \frac{\partial^2 u}{\partial y^2} \right)_P + \ldots
\]

\[
\leq u(P) + \overline{PQ} \left( \frac{\partial u}{\partial y} \right)_{\bar{y}}
\]

where \( \bar{y} \) is a point on the line \( PQ \), therefore

\[
|u(Q) - u(P)| \leq \overline{PQ} \left( \frac{\partial u}{\partial y} \right)_{\bar{y}} \quad \text{for} \quad V P \in \Omega \cap \partial \Omega^{(1)} \quad (2.4.66)
\]

Since \( \overline{PQ} < h \) and \( \left| \frac{\partial u}{\partial y} \right|_{\bar{y}} < M_1 \) thus we obtain

\[
|u(Q) - u(P)| \leq h M_1, \quad (2.4.67)
\]
so that in (2.4.62)

\[ \max_{(x,y) \in \Omega_h} |u(P) - u_h(P)| \leq h M_1 \]  

(2.4.68)

and we have

\[ |u(x,y) - u_h(x,y)| \leq \frac{h^2 r^\alpha M_4}{24} + h M_1 \quad \forall (x,y) \in \Omega_h. \]  

(2.4.69)

Hence, the order of the error has dropped from \( h^2 \) to \( h \), which illustrates the fact that we would obtain less accuracy in the solution. Note that the procedure reduces the number of points at which \( u_h \) is unknown. The same discussion can be followed for the point \( Q' \) on the horizontal line which passes through point \( P \).

ii) (This method is due to Collatz). For any point \( P \in \Omega \cap \partial \Omega_h \) let \( Q \) be the point of \( \Omega \) nearest to \( P \) which is on a horizontal or vertical line through \( P \). Let \( T \) be the point on the line \( PQ \) but on the opposite side of \( P \) to \( Q \) defined as follows:

If a point of \( \partial \Omega \) lies between \( P \) and the adjacent mesh point \( Z \) in the direction \( \overrightarrow{QP} \), then we let \( T \) be this point (Fig. 2.4.5a), otherwise we choose \( T \) to be \( Z \) (Fig. 2.4.5b) where \( Z \in \partial \Omega_h \) or \( \partial \Omega \) if \( P \in \partial \Omega_h^{(1)} \), and \( Z \) must be in \( \partial \Omega_h \) if \( P \in \partial \Omega_h - \partial \Omega_h^{(1)} \).

(Fig. 2.4.5a)  

(Fig. 2.4.5b)
The value of $u_h(P)$ is determined using linear interpolation defined by

$$u_h(P) = \frac{h_1}{h_1 + h_2} u_h(T) + \frac{h_2}{h_1 + h_2} u_h(Q),$$

(2.4.69)

where $h_1 = \overline{PQ}$ and $h_2 = \overline{PT}$. If $T \in \Omega$ then $u_h(P)$ is immediately determined. Otherwise, (2.4.69) is added to the existing system of difference equations. In order to bound the error in (2.4.69) we consider the Taylor formula

$$u(P) = \frac{h_1}{h_1 + h_2} u(T) + \frac{h_2}{h_1 + h_2} u(Q) - \frac{h_1 h_2}{2!} \left( \frac{\partial^2 u}{\partial x^2} \right)_{P'},$$

(2.4.70)

where $P' \in \Omega$ (the set of interior points of $\Omega$). The above equation is derived from (2.4.36) by $h_1 = S_1 h$ and $h_2 = S_3 h$. From (2.4.69) and (2.4.70) we obtain

$$u(P) - u_h(P) = \frac{h_1}{h_1 + h_2} (u(T) - u_h(T)) + \frac{h_2}{h_1 + h_2} (u(Q) - u_h(Q))$$

$$- \frac{h_1 h_2}{2!} \left( \frac{\partial^2 u}{\partial x^2} \right)_{P'},$$

(2.4.71)

But $(h_1 - h_2)^2 > 0$ implies that $\frac{h_1 h_2}{4} < \frac{h_1^2 + h_2^2}{8}$ which illustrates that:

$$\frac{h_1 h_2}{2} < \frac{(h_1 + h_2)^2}{8},$$

(2.4.72)

and therefore for all points $P \in \Omega_h^{(1)}$
\[ |u(P) - u_h(P)| \leq \frac{h_1}{h_1 + h_2} |u(T) - u_h(T)| + \frac{h_2}{h_1 + h_2} |u(Q) - u_h(Q)| + \frac{h^2}{2} M_2. \]

(2.4.73)

Now, two situations are possible for a point \( P \) according to the \( x \) and \( y \) directions which we characterise by \( x \).

(1) \( Q, T \) both belong to \( \Omega \) (Fig. 2.4.5a). Thus in (2.4.73)

\[ h_1 < h, \quad h_2 < h, \quad u(T) = u_h(T) \text{ and } u(Q) = u_h(Q). \]

Therefore,

\[ |u(P) - u_h(P)| \leq \frac{h^2}{2} M_2, \quad P \in \Omega_h. \]

(2.4.74)

(2) \( Q \in \Omega, \; T \equiv Z \in (\Omega_h \cup \Omega_h) \cap \Omega \) (Fig. 2.4.5b). In this case

\[ h_1 < h, \quad h_2 = h \text{ then } h_1 < h_2 \text{ which implies } \frac{h_1}{h_1 + h_2} < \frac{1}{2}, \]

and

\[ u(Q) - u_h(Q) = 0. \]

Thus, from (2.4.73) we obtain

\[ |u(P) - u_h(P)| \leq \frac{1}{2} |u(T) - u_h(T)| + \frac{h^2}{2} M_2, \quad P \in \Omega_h. \]

(2.4.75)

Combining (2.4.74) and (2.4.75) we have that for both situations

\[ |u(P) - u_h(P)| \leq \frac{1}{2} \max_{\Omega_h \cup \Omega_h} |u(T) - u_h(T)| + \frac{h^2}{2} M_2, \quad P \in \Omega_h. \]

(2.4.76)
Let
\[ \alpha = \max_{P \in \Omega_h \cup \partial \Omega_h} |u(P) - u_h(P)| \] (2.4.77)
and
\[ \beta = \max_{P \in \Omega_h} |u(P) - u_h(P)| . \] (2.4.78)

Now (2.4.76) holds for all \( P \in \partial \Omega_h \) and so
\[ \max_{P \in \partial \Omega_h} |u(P) - u_h(P)| \leq \frac{1}{2} \max_{T \in \Omega_h \cup \partial \Omega_h} |u(T) - u_h(T)| + \frac{h^2 M_2}{2} \] (2.4.79)
or
\[ \beta \leq \frac{1}{2} \max(\alpha, \beta) + \frac{h^2 M_2}{2} . \] (2.4.80)

Substituting the above bound in (2.4.62) we obtain:
\[ |u(x,y) - u_h(x,y)| \leq \frac{h^2}{24} + \frac{r^2 M_4}{24} + \max_{Q \in \partial \Omega_h} (1) |u(Q) - u_h(Q)| , \]
\[ P \in \Omega_h \cup \partial \Omega_h \] (2.4.81)
or
\[ \alpha \leq \frac{h^2}{24} + \frac{r^2 M_4}{24} + \beta . \] (2.4.82)

Now there exists two possibilities: (a) \( \alpha \geq \beta \) or (b) \( \beta \geq \alpha \)

(a) \( \alpha \geq \beta : \) then from (2.4.80) we have
\[ \beta \leq \frac{1}{2} \alpha + \frac{h^2 M_2}{2} \] (2.4.83)

and from (2.4.82) it can be seen that
\[ \alpha \leq \frac{h^2 r^2 M_4}{24} + \frac{1}{2} \alpha + \frac{h^2 M_2}{2} \]  
\[ \text{(2.4.84)} \]

or

\[ \alpha \leq \frac{h^2 r^2 M_4}{12} + h^2 M_2 . \]  
\[ \text{(2.4.85)} \]

(b) \( \beta \gg \alpha \): then from (2.4.80) we have

\[ \beta \leq h^2 M_2 \]  
\[ \text{(2.4.86)} \]

and from (2.4.82), we obtain

\[ \alpha \leq \frac{h^2 r^2 M_4}{12} + h^2 M_2 . \]  
\[ \text{(2.4.87)} \]

Thus, we have

\[ \max_{\bar{P} \in \Omega_h, \Omega_{h, h}} |u(P) - u_h(P)| \leq \frac{h^2 r^2 M_4}{12} + h^2 M_2 , \]  
\[ \text{(2.4.88)} \]

which implies that the method has an accuracy of order \( O(h^2) \).

iii) By using Difference equations derived from a non-square mesh, at all points \( P \in \Omega \cap \Omega_h \), and it can be seen that the bound is obtained from

\[ \max_{\bar{P} \in \Omega_h, \Omega_{h, h}} |u(P) - u_h(P)| \leq \frac{h^2 r^2 M_4}{6} + 2h^2 M_2 \]  
\[ \text{(2.4.89)} \]

which is twice the bound given in (2.4.88) \[ \text{[GREENSPAN, 1965]} \].

The shortcomings of the above results.

(1) They all demand that \( \frac{\partial^4 u}{\partial x^4} \) and \( \frac{\partial^4 u}{\partial y^4} \) be bounded on \( \bar{\Omega} \). A
sufficient condition for this is that \( u \in C^4(\Omega) \), which only happens in practice when \( \Omega \) is smooth, i.e. no difficult corners, see [WHITEMAN & WEBB, 1970].

(2) Difficulty of extending the analysis to the usual nine point difference operator when the region has curved boundaries.

(3) The strategy cannot be extended to problems involving Neumann boundary conditions.

We shall now derive finite difference approximations of second-order self-adjoint elliptic partial differential equations.

Considering the equation (2.4.22) if

\[
D = \frac{\partial A}{\partial x} \quad \text{and} \quad E = \frac{\partial C}{\partial y},
\]

then the equation (2.4.22) can be rewritten in the form:

\[
\frac{\partial}{\partial x} \left( A \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left( C \frac{\partial u}{\partial y} \right) + Fu = G
\]

which is a second-order self-adjoint elliptic partial differential equation.

In order to derive the finite difference approximation of (2.4.91) for a non-square mesh, finite difference expression of the following form are used.

\[
\frac{\partial}{\partial x} \left( A \frac{\partial u}{\partial x} \right) = \left\{ A(x + \frac{1}{2} S_1 h, y) \left[ \frac{u(x+S_1 h, y) - u(x, y)}{S_1 h} \right] - A(x - \frac{1}{2} S_3 h, y) \left[ \frac{u(x, y) - u(x-S_3 h, y)}{S_3 h} \right] \right\} / \frac{h(S_1+S_3)}{2}
\]
and
\[
\frac{\partial^2 u}{\partial y^2} = \left\{ C(x,y + \frac{1}{2} S_2 h) \left[ \frac{u(x,y+S_2 h) - u(x,y)}{S_2 h} \right] ight. \\
\left. - C(x,y - \frac{1}{2} S_4 h) \left[ \frac{u(x,y) - u(x,y-S_4 h)}{S_4 h} \right] \right\} \frac{h}{2} (S_2 + S_4). 
\]  
(2.4.92)

The resulting finite difference equation then has the usual form:
\[
\alpha_1 u(x+S_1 h,y) + \alpha_3 u(x-S_3 h,y) + \alpha_2 u(x,y+S_2 h) + \alpha_4 u(x,y-S_4 h) - \alpha_0 u(x,y) = \tau(x,y),
\]
where now
\[
\alpha_1 = \frac{A(x + \frac{1}{2} S_1 h,y)}{S_1 (S_1 + S_3)} , \quad \alpha_3 = \frac{A(x - \frac{1}{2} S_3 h,y)}{S_3 (S_1 + S_3)} ,
\]
\[
\alpha_2 = \frac{C(x,y + \frac{1}{2} S_2 h)}{S_2 (S_2 + S_4)} , \quad \alpha_4 = \frac{C(x,y - \frac{1}{2} S_4 h)}{S_4 (S_2 + S_4)} ,
\]
\[
\alpha_0 = \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 - \frac{1}{2} h^2 \tau .
\]  
(2.4.93)

and
\[
\tau = \frac{1}{2} h^2 G.
\]

Consequently, a difference equation can be obtained for which the local accuracy is as good as $O(h^2)$ for both regular and irregular mesh points. Moreover, if there are no irregular points in $\Omega_h$, the set of all interior mesh points, then the difference equation is symmetric.
Varga's Integral Method.

This method suggested by [VARGA, 1962] and further analysed by [GRIFFIN & VARGA, 1963] is to approximate the integral of the differential problem. With each mesh point we associate a "mesh region" $\Omega_{ij}$, such that the solution $u$ of the problem satisfies the double-integral of the differential equation (2.4.20) over the region $\Omega_{ij}$. The double-integral is then approximated in terms of the function values at nearby mesh points to give an expression for $u_{ij}$.

The boundary $\partial \Omega$ of the region $\Omega$ is approximated by a polygonal boundary $\partial \Omega'$, and the mesh is chosen so that it coincides with $\partial \Omega'$ only at mesh points. The mesh region is defined so that the boundary of $\Omega_{ij}$ lies half-way between the mesh lines (see Fig. 2.4.6).

As an example, we consider approximating the second-order self-adjoint elliptic partial differential equation

$$- \frac{\partial}{\partial x} \left( p(x,y) \frac{\partial u}{\partial x} \right) - \frac{\partial}{\partial y} \left( p(x,y) \frac{\partial u}{\partial y} \right) + \sigma(x,y) u = f(x,y) \quad \text{in } \Omega$$

(2.4.94)
\[ \alpha(x,y) u + \beta(x,y) \frac{\partial u}{\partial n} = \gamma(x,y) \quad \text{on } \partial \Omega. \quad (2.4.95) \]

It is assumed that \( P, \sigma, f \) are continuous in \( \overline{\Omega} = \Omega \cup \partial \Omega \), with

\[ P(x,y) > 0, \quad \sigma(x,y) > 0, \quad (x,y) \in \overline{\Omega} \quad (2.4.96) \]

and that \( \alpha, \beta, \gamma \) defined on the boundary of \( \Omega \), are piecewise continuous and satisfy

\[ \alpha(x,y) > 0, \quad \beta(x,y) > 0, \quad \alpha + \beta > 0, \quad (x,y) \in \partial \Omega. \quad (2.4.97) \]

The mesh spacings defined by

\[ h_i = x_{i+1} - x_i, \quad k_j = y_{j+1} - y_j \quad (2.4.98) \]

need not be uniform in each co-ordinate direction.

Let \( \Omega_h \) be the set of all interior mesh points and \( \partial \Omega_h \) be the set of all boundary mesh points, then the "mesh region" \( \Omega_{ij} \) associated with the mesh point \((x_i, y_j)\) lying in \( \Omega_h \) is defined by the lines

\[ x = x_i - \frac{h_i - 1}{2}, \quad x = x_i + \frac{h_i}{2} \quad (2.4.99) \]

\[ y = y_j - \frac{k_j - 1}{2}, \quad y = y_j + \frac{k_j}{2} \]

(See Fig. 2.4.7)
For each mesh point \((x_i, y_j)\) for which \(u(x_i, y_j) = u_{i,j}\) is unknown, equation (2.4.94) is integrated over the corresponding "mesh region" \(\Omega_{ij}\) so,

\[
- \int_{\Omega_{ij}} \left[ \frac{\partial}{\partial x} \left( p \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left( p \frac{\partial u}{\partial y} \right) \right] \, dx \, dy + \int_{\Omega_{ij}} \, \nabla u \, dx \, dy = \int_{\Omega_{ij}} \, f \, dx \, dy \quad .
\]

(2.4.100)

Now, it is known that if \(\phi(x,y)\) and \(\psi(x,y)\) are any two differentiable functions defined in \(\Omega_{ij}\), then by Green's theorem we have

\[
\int_{\Omega_{ij}} \left( \frac{\partial \phi}{\partial x} - \frac{\partial \psi}{\partial y} \right) \, dx \, dy = \int_{\partial \Omega_{ij}} (\psi \, dx + \phi \, dy) \quad ,
\]

(2.4.101)

where \(\partial \Omega_{ij}\) is the boundary of \(\Omega_{ij}\).

Assuming that,

\[
\phi = - p \frac{\partial u}{\partial x} \quad \text{and} \quad \psi = p \frac{\partial u}{\partial y}
\]

(2.4.102)

then (2.4.100) becomes

\[
\int_{\partial \Omega_{ij}} \left[ p \frac{\partial u}{\partial y} \, dx - p \frac{\partial u}{\partial x} \, dy \right] + \int_{\Omega_{ij}} \, \nabla u \, dx \, dy = \int_{\Omega_{ij}} \, f \, dx \, dy \quad .
\]

(2.4.103)

If \(g(x_i, y_j) = g_{i,j}\), then as an approximation of the double integrals above we have:

\[
\int_{\Omega_{ij}} \, g \, dx \, dy = g_{i,j} A_{i,j}
\]

(2.4.104)

where
\[ A_{i,j} = \text{area } (\Omega_{ij}) \]
\[ = \frac{(h_{i-1} + h_i)(k_{j-1} + k_j)}{4}. \quad (2.4.105) \]

Also, the line integral of (2.4.103) taken over the four sides of \( \Omega_{ij} \) can be approximated using central differences by

\[
\int_{\partial \Omega_{ij}} \left[ p \frac{\partial u}{\partial y} \, dx - p \frac{\partial u}{\partial x} \, dy \right] = \frac{h_{i-1} + h_i}{2} \left[ p_{i,j+\frac{1}{2}} \frac{u_{i,j} - u_{i,j+1}}{k_j} \right] + \frac{k_{j-1} + k_j}{2} \left[ p_{i+\frac{1}{2},j} \frac{u_{i,j} - u_{i+1,j}}{h_i} \right]
\]
\[ + p_{i-\frac{1}{2},j} \frac{u_{i+1,j} - u_{i,j}}{h_{i-1}} \]  
\[ + p_{i,j-\frac{1}{2}} \frac{u_{i,j-1} - u_{i+1,j}}{k_{j-1}} \]  
\[ = \left[ p_{i+\frac{1}{2},j} \frac{u_{i+1,j} - u_{i,j}}{h_{i+1} + h_i} \right] \left[ p_{i,j+\frac{1}{2}} \frac{u_{i,j+1} - u_{i,j}}{k_{j+1} + k_j} \right]. \quad (2.4.106) \]

where

\[ p_{i+1,j} = p(x_{i+\frac{1}{2}}, y_j), \text{ etc.} \quad (2.4.107) \]

Thus, for each mesh point \((x_i,y_j)\) where \(u(x_i,y_j)\) is unknown, a finite difference expression of the following form has been derived.

\[
C_{i,j} u_{i,j} - W_{i,j} u_{i-1,j} - E_{i,j} u_{i,j+1} - S_{i,j} u_{i,j-1} - N_{i,j} u_{i,j+1} = \frac{h_{i-1} + h_i}{2} \left[ \frac{k_{j-1} + k_j}{2} \right] \tau_{i,j}. \quad (2.4.108)
\]
where, for mesh point \((x_i, y_j)\) of \(\Omega_h\) we have,

\[
W_{i,j} = p_{i-1,j} \frac{k_{j-1} + k_j}{2h_{i-1}}
\]

\[
E_{i,j} = p_{i+1,j} \frac{k_{j-1} + k_j}{2h_i}
\]

\[
N_{i,j} = p_{i,j+1} \frac{h_{i-1} + h_i}{2k_j}
\]

\[
S_{i,j} = p_{i,j-1} \frac{h_{i-1} + h_i}{2k_{j-1}}
\]

\[
C_{i,j} = N_{i,j} + S_{i,j} + W_{i,j} + E_{i,j} + c_{i,j} \left( \frac{h_{i-1} + h_i}{2} \right) \left( \frac{k_{j-1} + k_j}{2} \right)
\]

and

\[
F_{i,j} = f_{i,j} \left( \frac{h_{i-1} + h_i}{2} \right) \left( \frac{k_{j-1} + k_j}{2} \right)
\]

The equation (2.4.108) is called a 5-point formula, as each mesh point \((x_i, y_j)\) is coupled to at most four other adjacent mesh points. The system of linear equations corresponding to the element \(u_{i,j}\) can be written as

\[
A \hat{u} = k + \tau(u)
\]

(2.4.110)

neglecting the term \(\tau(u)\). We solve the system

\[
A \hat{u} = k
\]

(2.4.111)

as an approximation to \(u\).
It can be seen that the matrix $A$ has the following properties:

1. $A$ is real and symmetric,
2. it is diagonally dominant,
3. $A$ is an irreducible matrix.

Therefore, $A$ is a positive definite matrix.

If we put $P = 1$ and take constant mesh sizes $h_i = h$, $k_j = k$ in (2.4.109), therefore, on division by $hk$, equation (2.4.108) becomes:

$$
\frac{(2u_{i,j} - u_{i-1,j} - u_{i+1,j})}{h^2} + \frac{(2u_{i,j} - u_{i,j+1} - u_{i,j-1})}{k^2} + \sigma_{i,j} u_{i,j} = f_{i,j} + \tau_{i,j}
$$

(2.4.112),

which is identical to the result obtained using the Taylor series.
2.4.3 FINITE ELEMENT METHOD.

Fundamental to the finite element approach for solving partial differential equations, is the concept of discretization, where the domain \( \Omega \) is divided into a large number of small subdomains which are known as "elements". As mentioned earlier in the finite difference approach, the approximate solution is defined at each isolated mesh point, however, the finite element method provides a global approximation, based on local representations which are then expressed in the form of the approximate solutions, in each individual element.

Let the region \( \Omega \) be partitioned into a mesh of triangular elements as shown in Figure (2.4.8), however other types of elements can also be used. Let the mesh points on each element be the vertices of the triangles, therefore, it can be assumed that in any element the approximate solution is of the form

\[
U_e(x, y) = \sum_{i=1}^{3} U^i \phi_i^e(x, y),
\]

where the functions \( \phi_i^e(x, y) \), known in general as "shape functions" are linear polynomials such that

\[
\phi_i^e(x_j, y_j) = \begin{cases} 
1 & i = j, \\
0 & i \neq j
\end{cases}, \quad 1 \leq i, j \leq 3.
\]

It is also possible to select more than three mesh points in each element, for example, if we choose six mesh points in each element then the shape functions \( \phi_i^e \) must be quadratic polynomials. The only restriction on the choice of mesh points in each element
is that linearly independent functions $\phi_e^i(x,y)$ must exist.

![Mesh Point and Element](Fig. 2.4.8)

It can be seen that along the interface between elements $e_1$ and $e_2$ (Fig. 2.4.8), $U_{e_1}$ is a linear equation in one variable defined uniquely by values at the points A and B. Similarly, on the same interface $U_{e_2}$ is again a linear function defined by values at the points A and B. Thus $U_{e_1} = U_{e_2}$ along AB and therefore the global solution is continuous. From the definition of the "shape function" (2.4.114) it is evident that, the shape function on different elements associated with a mesh point on the common boundary are the same form on that boundary and reduce to zero along sides opposite the corresponding mesh point. By these facts we have a global piecewise polynomial, known as a "basis function", for each mesh point. Therefore, for any mesh point $x_1 \in (x_1^i, y_1^i)$ the corresponding basis function is defined element-by-element as

$$
\phi_1^i(x) = \begin{cases} 
\phi_e^i(x) & \text{if } x_1^i \in e \\
0 & \text{if } x_1^i \notin e 
\end{cases}
$$

(2.4.115)
The union of elements in which \( \phi_i \neq 0 \) is termed the "support" and is denoted by \( S(\phi_i) \). Having established the piecewise basis functions we now can write the approximate finite element solution in the following form

\[
U(x,y) = \sum_{i=1}^{N} U_i \phi_i(x,y)
\]

where \( N \) is the number of mesh points and \( \phi_i(x,y), \ i = 1,2,\ldots,N, \) are piecewise basis functions. Using the approximate solution (2.4.116) the problem is now to describe the equations which define the unknown values \( U_i \) at the internal mesh points of the region.

It can be shown that all the common versions of the finite element method have a variational formulation.

For purposes of illustration, let us consider the following partial differential equation

\[
\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f(x,y) \quad \text{in } \Omega
\]

subject to prescribed values of \( u \) on the boundary of region. Assume that the approximate solution is of the form (2.6.116), one form that can be useful in computing the finite element approximation is "the minimization of the functional", known as "the variational principle" which is used in constructing the approximate solution by the Rayleigh-Ritz method. For this, we use a fundamental result of the calculus of variations which states that the solution of the partial differential equation (2.4.117) is equivalent to finding a function \( u \) which satisfies the same boundary conditions and minimizes the value of the integral
A more general approach for solving the finite element approximation of the form (2.4.116) with the formulation (2.4.118) also included is the "weighted residuals process". In this method, we construct \( N \) linearly independent "test functions" \( \psi_j(x,y) \) and then the approximate solution satisfies

\[
\iint_\Omega \psi_j \left[ \frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} - f \right] \, dx \, dy = 0, \quad j = 1, 2, \ldots, N \tag{2.4.119}
\]

in which \( U(x,y) \) is given by equation (2.4.116). Applying the Green's theorem (integration by parts) to the above integral, then equation (2.4.119) can be replaced by the following formula:

\[
a\left[ U, \psi_j \right] = (f, \psi_j), \quad j = 1, 2, \ldots, N \tag{2.4.120}
\]

where

\[
a\left[ U, \psi_j \right] = \iint_\Omega \left[ \left( \frac{\partial U}{\partial x} \right) \left( \frac{\partial \psi_j}{\partial x} \right) + \left( \frac{\partial U}{\partial y} \right) \left( \frac{\partial \psi_j}{\partial y} \right) \right] \, dx \, dy - \oint_{\partial \Omega} \psi_j \frac{\partial U}{\partial n} \, ds \tag{2.4.121}
\]

and

\[
(f, \psi_j) = \iint_\Omega f \psi_j \, dx \, dy \quad . \tag{2.4.122}
\]

Substituting (2.4.116) - (2.4.120) we obtain

\[
\sum_{i=1}^{N} U_i a\left[ \phi_i, \psi_j \right] = (f, \psi_j), \quad j = 1, 2, \ldots, N \tag{2.4.123}
\]
Thus, the approximate problem reduces to the solution of a system of linear algebraic equations

\[ A \hat{\mathbf{u}} = \hat{\mathbf{f}} \]  

(2.4.124)

where the matrix \( A \) is defined as

\[ A_{ij} = a[\phi_i, \psi_j] \]  

(2.4.125)

and

\[ \hat{\mathbf{u}} \equiv (u_1, u_2, \ldots, u_N)^T \quad \text{and} \quad \hat{\mathbf{f}} = ((f, \psi_1), (f, \psi_2), \ldots, (f, \psi_N))^T. \]  

(2.4.126)

It is evident that the form of the matrix \( A \) depends on the choice of the basis functions \( \phi_i \) and the test functions \( \psi_j \).

In the following we mention other methods for constructing test functions which result in a positive definite coefficient matrix \( A \).

(i) **Galerkin Scheme.** In this method we choose the test function \( \psi_j \) such that \( \psi_j = \phi_j \), then the matrix \( A \) is symmetric and positive definite. Moreover, "The number of non-zeros in the \( i \)th row corresponds to the number of nodes in \( S(\phi_i) \). If \( M_1 \) is the maximum number of mesh points per element and \( M_2 \) is the maximum number of elements that meet at a common vertex, the number of non-zeros in any row is bounded by \( M_1 M_2 - M_2 + 1 \). See [GLADWELL & WAIT, 1979]."

(ii) **Least-square Method.** Here we take the test functions in the form \( \psi_j = \Delta \phi_j \), then the equation (2.5.7) is equivalent to the determination of the parameters \( U_i \) in (2.5.4) such that, the \( L_2 \) norm of the residual \( (\Delta U - f) \) is a minimum, i.e.
minimise \[ \left\{ u_i \right\} \left| \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} - f \right|^2 \] (2.4.127)

By these forms of test functions we again have a positive definite matrix.

We now study the construction of the basis function for elements of triangular shape which probably is the most widely used finite element.

The basis functions are usually derived from local Lagrange interpolation formulae, the linear functions given earlier being just one example.

(i) The linear case. We assume the vertices of the triangle are \( x_i = (x_i, y_i), \ i = 1, 2, 3 \), then the polynomial is

\[ P_1(x, y) = a_1 + a_2 x + a_3 y \]

\[ = \sum_{j=1}^{3} U_j \phi_j^{(1)}(x, y) \]

where \( U_j, \ j = 1, 2, 3, \) are the values of \( U(x, y) \) at the vertices of the triangular element. Let

\[ D_{jk} = \det \begin{bmatrix} 1 & x & y \\ 1 & x_j & y_j \\ 1 & x_k & y_k \end{bmatrix} \]

and

\[ C_{ijk} = \det \begin{bmatrix} 1 & x_i & y_i \\ 1 & x_j & y_j \\ 1 & x_k & y_k \end{bmatrix} \]
Then

\[ \phi_i(x,y) = \frac{D_{jk}}{C_{ijk}} \]

with \((i,j,k)\) as any permutation of \((1,2,3)\).

\[ \text{(ii) The quadratic case.} \]

The polynomial is now

\[ P_2(x,y) = \alpha_1 + \alpha_2 x + \alpha_3 y + \alpha_4 x^2 + \alpha_5 xy + \alpha_6 x^2 \]

\[ = \sum_{j=1}^{6} U_j \phi_j^{(2)}(x,y) \]

where \(U_j, j = 1,2,\ldots,6\), are the values of \(U(x,y)\) at the vertices of the triangular elements together with the values at the mid-points of the sides of the triangle. The functions \(\phi_j^{(2)}(x,y), j = 1,2,\ldots,6\), are given by

\[ \phi_1^{(2)} = \phi_1^{(1)}(2\phi_1^{(1)} - 1), \text{ with } \phi_2^{(2)} \text{ and } \phi_3^{(2)} \text{ similar, and} \]

\[ \phi_4^{(2)} = 4\phi_1^{(1)} \phi_2^{(1)} \]

with \(\phi_5^{(2)}\) and \(\phi_6^{(2)}\) similar.

Further discussion along these lines and the construction of higher degree basis functions and other possible alternative interpolating schemes can be found in MITCHELL & WAIT [1977].
CHAPTER 3

METHODS FOR SOLVING LINEAR SYSTEMS OF EQUATIONS
3.1 **INTRODUCTION.**

As mentioned in Chapter Two the application of finite difference and finite element methods for solving elliptic partial differential equations always generates an associated algebraic problem. In general this algebraic problem involves the solution of large sets of linear equations of the form

\[ Au = b \]  

(3.1.1)

This chapter is concerned with various methods which are available for the solution of (3.1.1). We give particular prominence to the solution of equations arising from finite difference calculations.

In order to illustrate our discussion we consider the following model problem:

\[ \nabla^2 u = 0 \quad \text{in } \Omega \]  
\[ u = g(x,y) \quad \text{on } \partial \Omega \]  

(3.1.2) \hspace{1cm} (3.1.3)

where \( \Omega \) is the interior and \( \partial \Omega \) is the boundary of a simply connected open bounded domain in \( xy \)-plane.

For fixed \( h > 0 \) and fixed \( (\bar{x}, \bar{y}) \), the point sets \( \Omega_h, \partial \Omega_h \) and \( \overline{\Omega}_h = \Omega_h \cup \partial \Omega_h \) are constructed as described in the section (2.4.2). Suppose \( \Omega_h \) consists of \( m \) points, \( \partial \Omega_h \) consists of \( n \) points, and therefore, \( \overline{\Omega}_h \) consists of \( m+n \) points, we number the points of \( \Omega_h \) in a one-to-one fashion with the integers 1, 2, ..., \( m \) and there exists many different methods of ordering these points such as:

i) "natural" ordering, i.e. a point \( (\bar{x} + ph, \bar{y} + qh) \) occurs before \( (\bar{x} + p'h, \bar{y} + q'h) \) if \( q < q' \) or if \( q = q' \) and \( p < p' \).
ii) "red-black" ordering, i.e. all points \((x + ph, y + qh)\) with \(p + q\) even ("red" points) occur before those with \(p + q\) odd ("black" points),

iii) "ordering by diagonal", i.e. a point \((x + ph, y + qh)\) occurs before \((x + p'h, y + q'h)\) if \(p + q < p' + q'\), etc..

We then number the points on the boundary \(\partial \Omega_h\) in a similar fashion with the integers \(m+1, m+2, \ldots, m+n\), and at each point \((x, y)\) of \(\partial \Omega_h\), we set \(u(x, y) = g(x, y)\). If \((x, y)\) is numbered \(k\), then using subscript notation, this is equivalent to \(u_k = g(x, y)\).

Finally each point \((x_i, y_j)\) of \(\Omega_h\), beginning with point numbered 1 and continuing in consecutive order through the point numbered \(m\), we write down the Laplace difference analogue (2.4.42) in the form

\[
u(x_{i+1}, y_j) + u(x_{i-1}, y_j) + u(x_i, y_{j+1}) + u(x_i, y_{j-1}) - 4u(x_i, y_j) = 0 \tag{3.1.4}
\]

where the points \((x_{i+1}, y_j)\), \((x_{i-1}, y_j)\), \((x_i, y_{j+1})\) and \((x_i, y_{j-1})\) are associated with points \((x_i + h, y_j)\), \((x_i - h, y_j)\), \((x_i, y_j + h)\) and \((x_i, y_j - h)\) respectively which are the four neighbours of the point \((x_i, y_j)\). If applying (3.1.4) at a given point \((x_i, y_j)\), any of the points \((x_{i+1}, y_j)\), \((x_{i-1}, y_j)\), \((x_i, y_{j+1})\) and \((x_i, y_{j-1})\) are boundary points then in (3.1.4) we replace \(u\) at these points by the known values \(g_i\) determined previously. In practice, the equation (3.1.4) is usually written in the following form:

\[
u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{i,j} = 0 \tag{3.1.5}
\]
The above equation is generally represented by the molecule in Fig. (3.1).

\[
\begin{array}{c}
\begin{array}{c}
-1 \\
(i, j+1)
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
-1 \\
(i-1, j)
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
4 \\
(i, j)
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
-1 \\
(i+1, j)
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
-1 \\
(i, j-1)
\end{array}
\end{array}
\end{array}
\] (Fig. 3.1)

which when applied at each interior mesh point results in a set of independent, simultaneous, non-homogeneous, linear equations.

These equations can be written in the form (3.1.1) or can be expressed as:

\[
\sum_{j=1}^{n} a_{i,j} u_{j} = b_{i}, \quad i = 1, 2, \ldots, N \tag{3.1.6}
\]

where the matrix A satisfies the conditions given earlier (in Section 2.4.2), i.e. the matrix A is real, symmetric, diagonally dominant and irreducible, moreover by ordering the set of points \( \Omega_h \) in one of the ways described above the matrix A also possesses a feature called property (A) [YOUNG, 1971].

In the next section we shall concentrate on the different strategies used for obtaining the solution of the system (3.1.1).
3.2 METHODS FOR SOLVING A SET OF LINEAR EQUATIONS.

Suppose one wishes to solve the system (3.1.1) where $A$ is a square $N \times N$ matrix, $b$ is a given column matrix, and $u$ is an unknown column matrix. It will be assumed that $A$ is non-singular, hence a unique solution $u$ exists.

The methods used to obtain the solution of such systems mainly depends upon the structure of the coefficient matrix $A$.

That is, for large sparse matrices with only a few non-zero elements, we usually use iterative methods since these will not change the structure of the original matrix and therefore preserve sparsity. The disadvantages of these methods are mainly the problem of selecting a good initial vector with which the iterative process may commence and also the accuracy of the final solution. The chief advantage with iterative methods is that sparse matrix techniques can be used to store only the non-zero elements of the coefficient matrix and hence optimize the amount of storage used.

Direct methods, on the other hand, are used only when the matrix in question is small and dense. The advantages with these methods are that firstly no initial vector is required and secondly the accuracy of the final solution usually turns out to be satisfactory depending on the chosen word length of the machine. Direct methods cannot, unfortunately, be used for large sparse unordered matrices because of the problem of fill-ins which occurs during the elimination process. However, if the matrix is of regular shape, then special methods for sorting these matrix can be devised in order to minimise the amount of storage used.

We now turn our attention to describing some more well known direct and iterative methods in the following sections.
3.3 **DIRECT METHODS.**

In such methods a sequence of operations is performed, in general, once only which results in an approximation to the true result. The approximation enters only because multiplications and divisions, in particular, are stored and subsequently used with rounding errors.

We first describe in some detail the theory of Gaussian-elimination even though it is a widely known process. However, a detailed description of the method and some satisfactory error bounds can be found in [WILKINSON, 1965].

3.3.1 **Gaussian-elimination** is generally used to solve a set of linear equations of the form (3.1.1) by reducing $A$ to an upper triangular matrix. The elimination process in the matrix $A$ takes place in $(N-1)$ stages and involves evaluating matrix $A^{(k)}$, $k = 1, 2, \ldots, N-1$ where in general after the $(k-1)^{th}$ stage of elimination, the matrix $A^{(k-1)}$ has the form:

$$A^{(k-1)} = \begin{bmatrix}
 a_{1,1} & a_{1,2} & a_{1,3} & \cdots & a_{1,n} \\
 0 & a_{2,2} & a_{2,3} & \cdots & a_{2,n} \\
 0 & 0 & a_{3,3} & \cdots & a_{3,n} \\
 \vdots & \vdots & \vdots & \ddots & \vdots \\
 0 & 0 & 0 & \cdots & a_{n,n}
\end{bmatrix}$$

(3.3.1)
where the elements $a_{i,k}^{(k-1)}$, $i = k+1, k+2, \ldots, n$ are now eliminated by calculating

$$a_{i,j} = a_{i,j}^{(k-1)} - a_{i,k}^{(k-1)} \frac{a_{k,j}^{(k-1)}}{a_{k,k}^{(k-1)}} , \quad i,j = k+1, k+2, \ldots, n .$$

(3.3.2)

The accuracy of the final solution depends on the choice of the pivots. For an accurate result all the pivots have to be selected in such a way that the elements $a_{i,k}/a_{k,k}$ which represent the multipliers for the different stages of the elimination process (i.e. $k = 1, 2, \ldots, n-1$) should be less than or equal to unity. To achieve this we consider two of the most important pivotal strategies which are widely used.

1. Full pivoting. In this strategy at the start of the $k^{th}$ ($k = 1, 2, \ldots, n-1$) stage of the elimination process a search is carried out to find the maximum element in absolute value of the submatrix $(n-k+1) \times (n-k+1)$ which is formed by eliminating the $k^{th}$ row and column ($k = 2, 3, \ldots, n-1$) of the matrix and it is used as the pivotal element for the stage.

2. Partial pivoting. In this method of the $k^{th}$ stage ($k = 1, 2, \ldots, n-1$) of the elimination process a search is carried out only amongst the elements $a_{i,k}$ ($i = k, k+1, \ldots, n$) and the largest element in absolute value is selected as the pivot.

The Gaussian elimination process is usually programmed to accommodate $k$ right hand side vectors, in which case the storage required is $n(n+k)$ locations plus a possible further $nk$ for the final solution vectors. Finally, the amount of work involved,
can be seen to be:

\[
\begin{align*}
\text{division} & \quad \frac{1}{3} n^3 + n^2 - \frac{1}{3} n \\
\text{multiplications} & \quad \frac{1}{3} n^3 + \frac{1}{2} n^2 - \frac{5}{6} n \\
\text{additions} & \quad \end{align*}
\]

(3.3.3)

In practice, however, in the solution of sets of equations resulting from finite difference approximations to elliptic partial differential equations, this value of the work involved is never actually attained as the matrix \( A \) is never full. Instead \( A \) is in general a band matrix, i.e. \( A \equiv (a_{i,j}) \) where

\[
a_{i,j} = 0 \quad \begin{cases} 
    i - j > m \\
    j - i > m
\end{cases}
\]

so that the number of non-zero elements in each row is at most \( 2m+1 \) the bandwidth of the matrix. In this case, the number of multiplications is of order \( 2m^2 n \) in contrast to the factor \( n^3/3 \) for the full matrix [MARTIN, 1966].

3.3.2 LU Decomposition.

The LU factorisation of a matrix \( A \) can be obtained by Doolittle's method, which we illustrate here for a general \( n \times n \) matrix. In the relation \( A = LU \), or
\[
\begin{bmatrix}
a_{1,1} & \cdots & a_{1,j} & \cdots & a_{1,n} \\
\vdots & \ddots & \vdots & \ddots & \vdots \\
a_{i,1} & \cdots & a_{i,j} & \cdots & a_{i,n} \\
\vdots & \ddots & \vdots & \ddots & \vdots \\
a_{n,1} & \cdots & a_{n,j} & \cdots & a_{n,n}
\end{bmatrix}
= \begin{bmatrix}
1 & & & & \\
& \ddots & & & \\
& & \ell_{i,1} & \cdots & 1 \\
& \vdots & \vdots & \ddots & \vdots \\
& & \ell_{n,1} & \cdots & \ell_{n,i} & \cdots & 1 \\
\end{bmatrix}
\begin{bmatrix}
u_{1,1} & \cdots & u_{1,j} & \cdots & u_{1,n} \\
\vdots & \ddots & \vdots & \ddots & \vdots \\
v_{j,1} & \cdots & v_{j,j} & \cdots & v_{j,n} \\
\vdots & \ddots & \vdots & \ddots & \vdots \\
v_{n,1} & \cdots & v_{n,j} & \cdots & v_{n,n}
\end{bmatrix}
\]

(L is unit lower triangular and \( U \) is upper triangular) all the coefficients in \( L \) and \( U \) are initially unknown. The rule for matrix multiplication enables them to be found from the following equations:

\[
\begin{align*}
\text{for } j = i, i+1, \ldots, n & \quad u_{i,j} = a_{i,j} - \sum_{k=1}^{i-1} \ell_{i,k} u_{k,j} \\
\text{for } j = i+1, i+2, \ldots, n & \quad \ell_{j,i} = \frac{1}{u_{i,i}} \left[ a_{j,i} - \sum_{k=1}^{i-1} \ell_{j,k} u_{k,i} \right]
\end{align*}
\]

(3.3.4)

This factorisation method fails only if one of the diagonal elements \( u \), which are used as divisors in the second of (3.3.4) proves to be zero.

As in the Gaussian elimination process similar pivotal strategies must be employed in order to ensure sufficiently accurate results. A description of the Doolittle's method can be found in [Gault et al., 1974].

We now illustrate that the \( LU \) decomposition uniquely exists if \( A \) is a non-singular matrix. For this we have the following two Theorems.
Theorem 3.1 The \( n \times n \) matrix \( A \) has an \( LU \) decomposition where \( L \) and \( U \) are non-singular, if the submatrices

\[
A_k = \begin{bmatrix}
a_{1,1} & \cdots & a_{1,k} \\
\vdots & & \vdots \\
a_{k,1} & \cdots & a_{k,k}
\end{bmatrix},
\]

are all non-singular. (\( A_k \) is called the leading principal matrix of order \( k \).)

Proof.

We assume an inductive hypothesis such that

\[
A_{k-1} = L_{k-1} U_{k-1}
\]

where \( L_{k-1} \) is a unit lower triangular and \( U_{k-1} \) is an upper triangular matrix.

Let \( A_k \) be partitioned as:

\[
A_k = \begin{bmatrix}
A_{k-1} & b \\
\hline
a^T & a_{k,k}
\end{bmatrix},
\]

where

\[
a^T = \begin{bmatrix} a_{k,1} & \cdots & a_{k,k-1} \end{bmatrix} \quad \text{and} \quad b = \begin{bmatrix} a_{1,k} \\
\vdots \\
a_{k-1,k} \end{bmatrix}.
\]
and let $L_k$ and $U_k$ be defined by:

$$L_k = \begin{bmatrix} L_{k-1} & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix} \quad \text{and} \quad U_k = \begin{bmatrix} U_{k-1} & \mathbf{u} \\ \mathbf{0}^T & u_{k,k} \end{bmatrix},$$

where $\mathbf{f}^T = \begin{bmatrix} f_{k,1} & \cdots & f_{k,k-1} \end{bmatrix}$ and $\mathbf{u} = \begin{bmatrix} u_{1,k} \\ \vdots \\ u_{k-1,k} \\ u_{k-1,k} \end{bmatrix},$

then $\mathbf{f}^T$, $\mathbf{u}$ and $u_{k,k}$ can be uniquely determined such that

$$L_k U_k = \begin{bmatrix} L_{k-1} & U_{k-1} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} L_{k-1} \mathbf{u} \\ \mathbf{0} \end{bmatrix} + u_{k,k} = \begin{bmatrix} A_{k-1} & \mathbf{b} \\ \mathbf{a}^T \end{bmatrix} = A_k$$

This follows since $L_{k-1} U_{k-1} = A_{k-1}$ by hypothesis and then

$$\mathbf{f}^T = \mathbf{a}^T U_{k-1}^{-1} \quad \mathbf{u} = L_{k-1}^{-1} \mathbf{b} \quad \text{and} \quad u_{k,k} = a_{k,k} - \mathbf{f}^T \mathbf{u}$$

is uniquely determined since $L_{k-1}$ and $U_{k-1}$ are non-singular under the hypothesis. Now,

$$A_1 = \begin{bmatrix} a_{1,1} \end{bmatrix} = \begin{bmatrix} a_1,1 \end{bmatrix}$$ (3.3.9)
defines the decomposition of \( A_1 \). Hence, by induction, \( A = A_n \) has a LU decomposition.

Notes:
If \( A \) is a real matrix then \( L \) and \( U \) are also real.
Row or column diagonally dominant matrices satisfy the hypotheses of the Theorem.
Real symmetric positive definite matrices satisfy the hypotheses of the Theorem.

**Theorem 3.2.** If there exists an LU decomposition of a non-singular matrix \( A \), where \( L \) is unit lower triangular and \( U \) upper triangular matrix, then this decomposition is unique.

**Proof.**
Let \( A = L^{(1)} U^{(1)} = L^{(2)} U^{(2)} \) be two decompositions, where the matrices are non-singular, since \( A \) is non-singular, then

\[
L^{-1}_{(2)} L_{(1)} = U^{-1}_{(2)} U_{(1)} = I
\] (3.3.10)

since \( L^{-1}_{(2)} L_{(1)} \) is unit lower triangular and \( U^{-1}_{(2)} U_{(1)} \) is upper triangular.

**Equivalence of Gaussian elimination with LU Decomposition.**

Define the matrix \( M_k \) by
\[ M_k = \begin{bmatrix} 1 & \ldots & 0 & \ldots & 0 & \ldots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \ldots & 1 & \ldots & 0 & \ldots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \ldots & -m_{i,k} & \ldots & 1 & \ldots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \ldots & -m_{n,k} & \ldots & 0 & \ldots & 1 \end{bmatrix} \] (3.3.11)

(i.e. identity matrix with additional elements \(-m_{i,k}\) in \((i,k)\)th position for \(k + 1 \leq i \leq n\)). Then, the premultiplication of a \(n \times n\) matrix \(A\) by \(M_k\) has the effect of \(-m_{i,k}\) or row \(k\) being added to row \(i\) for \(k + 1 \leq i \leq n\).

The Gaussian elimination algorithm without pivoting can then be described theoretically by

\[
M_{n-1} \ast \cdots \ast M_k \ast \cdots \ast M_1 \ast A = \begin{bmatrix} a_{1,1}^{(1)} & \cdots & a_{1,k}^{(1)} & \cdots & a_{1,n}^{(1)} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ a_{k,k}^{(k)} & \cdots & a_{k,k}^{(k)} & \cdots & a_{k,n}^{(k)} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & a_{n,n}^{(n)} \end{bmatrix} \equiv U \quad \text{(say)}
\] (3.3.12)

where \(m_{i,k} = a_{i,k}^{(k)} / a_{k,k}^{(k)}\).

Thus, we have

\[ A = M_1^{-1} \ast \cdots \ast M_k^{-1} \ast \cdots \ast M_{n-1}^{-1} \ast U. \] (3.3.13)
But

\[ M_1^{-1} \cdots M_k^{-1} \cdots M_{n-1}^{-1} = \begin{bmatrix}
1 & & & & \\
& \ddots & & & \\
& & \ddots & & \\
& & & \ddots & \\
& & & & 1
\end{bmatrix} \equiv L \text{ (say)} \]  \hspace{1cm} (3.3.14)

Thus

\[ A = LU \]

where L is a unit lower triangular matrix and U is an upper triangular matrix.

It can be shown that the amount of work in the LU decomposition is the same as for Gauss' method. However, the disadvantage of the Gaussian elimination process in comparison with the LU decomposition is that each time an element of a reduced matrix is computed and stored a rounding error generally occurs. In the LU decomposition a rounding error can be largely avoided by the use of double-precision arithmetic in the calculation of the elements of L and U from equations (3.3.4). These results may then be rounded to single precision and reduced on the completion of each calculation. The removal of the need for computing and recording several intermediate matrices has, therefore, localised what might otherwise be a significant source of error to a single step in the determination of each element of L and U. The use of double-precision arithmetic in this step leads to a degree of accuracy comparable with that attained if the entire
Gaussian elimination process were carried out with double-precision, this latter is an unattractive proposition, in that it would require twice as much computer storage as the corresponding single precision solution.

3.3.3 The Crout Form of Factorisation for Symmetric Matrices.

This process will factorise $A$ in the form of $U^TDU$ in a recursive procedure where at the first step of the process, we write $A$ as:

$$A \equiv A^{(1)} = \begin{bmatrix}
a_1 & r_1^T \\
r_1 & C_1
\end{bmatrix}$$ (3.3.14)

where $a_1$ is a single element, $r_1$ is a $(n-1)\times1$ vector and $C_1$ is a $(n-1)\times(n-1)$ matrix. The first element of the diagonal matrix $D$, i.e. $d_{1,1}$, and the first row of the upper triangular matrix $U$ is now computed as:

$$u_{i,j} = \frac{r_{1,i}}{a_1} \quad , \quad j = 2, 3, \ldots, n$$ (3.3.15)

and

$$d_{1,1} = a_1$$

We then compute $A^{(2)}$ as:

$$A^{(2)} = \begin{bmatrix}
a_1 & r_1^T \\
r_1 & C_1 - \frac{r_1r_1^T}{a_1}
\end{bmatrix}$$ (3.3.16)
where the matrix \( \begin{bmatrix} r_1 & r_1^T \end{bmatrix} \) is subtracted element by element from \( C_1 \).

In general, at the \( k \)th step (1 \( \leq \) \( k \) \( \leq \) \( n-1 \)) we have \( A(k) \) as

\[
A(k) = \begin{bmatrix}
A_k & B_k^T \\
\vdots & \vdots \\
B_k & a_k & r_k^T \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots
\end{bmatrix}
\]

(3.3.17)

where \( A_k \) is a \((k-1) \times (k-1)\) matrix, \( B_k \) is a \((n-k+1) \times (k-1)\) array, \( a_k \) is a single element, \( r_k \) is a \((n-1) \times 1\) vector and \( C_k \) is a \((n-k) \times (n-k)\) matrix. We now have \( d_{k,k} = a_k \) and the \( k \)th row of the matrix \( U \) is replaced by \( r_k^T / a_k \). Finally, we compute \( A(k+1) \) as:

\[
A(k+1) = \begin{bmatrix}
A_k & B_k^T \\
\vdots & \vdots \\
B_k & a_k & r_k^T \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots
\end{bmatrix}
\]

(3.3.18)

3.3.4 **Choleski Decomposition (Square root method).**

Let the matrix \( A \) be symmetric and positive definite, then we can factorise \( A \) in the form \( LL^T \) where \( L \) is lower triangular. As an illustration, consider the decomposition of a general \( n \times n \) symmetric and positive definite matrix in the form:
Then, equating coefficients on and below the diagonal we obtain the unknown coefficients of $\mathcal{L}$ from the following equation

for $i = j$

$$ \xi_{i,i} = \left[ a_{j,j} - \sum_{k=1}^{i-1} \xi_{j,k}^2 \right]^{1/2} $$

for $i = j+1, j+2, \ldots, n$

$$ \xi_{i,j} = \frac{1}{\xi_{i,i}} \left[ a_{j,j} - \sum_{k=1}^{j-1} \xi_{i,k}^2 \right] $$

(3.3.19)

**Theorem 3.3** Let $A$ be a real symmetric and positive definite matrix, then there exists unique decompositions of the form:

$$ A = L D L^T = \mathcal{L} \mathcal{L}^T $$

(3.3.20)

where $L$ is a real unit lower triangular, $D$ is diagonal with real positive terms and $\mathcal{L}$ is a real lower triangular matrix with positive diagonal.

**Proof.**

If $A$ is real, symmetric and positive definite then its leading principal matrices $A_k$ are non-singular and moreover, are such that

$$ \det(A_k) > 0 ; \ k = 1,2,\ldots,n. $$
Thus by Theorem 3.1 there exists an unique decomposition

$$A = LU$$

where $L$ is a real unit lower and $U$ is a real upper triangular matrix.

Moreover, since

$$u_{k,k} = \begin{cases} a_{k,1}, & k = 1, \\ \frac{\det(A_k)}{\det(A_{k-1})}, & k > 1 \end{cases} \tag{3.3.21}$$

it follows that $U$ has positive diagonal terms. Now the decomposition may be rewritten as the unique form

$$A = L \, D \, V, \tag{3.3.22}$$

where $D = \text{diag}(u_{k,k})$ is a diagonal matrix with positive terms and $V = D^{-1}U$ is a real unit upper triangular matrix. Uniqueness and symmetry then imply that $V = L^T$, Thus:

$$A = LDL^T = LD\frac{1}{2}D\frac{1}{2}L^T = \mathcal{L}\mathcal{L}^T, \tag{3.3.23}$$

where $\mathcal{L} = LD\frac{1}{2}$, and $D\frac{1}{2}$, and hence $\mathcal{L}$ can be uniquely chosen to have positive diagonal terms.

Finally, it can be seen that the use of symmetric elimination methods, i.e. the Crout and the Choleski methods give a significant saving in computational cost. Both methods require $O(1/6 \, n^3)$ multiplications and $O(1/6 \, n^3)$ additions; however the Choleski method requires $n$ square roots as well. Therefore, Choleski's method is likely to be less efficient than the Crout method although it possesses superior accuracy and smaller storage requirements. The pivotal strategies can be applied with the same ease and would
involve no additional arithmetic, however, extra time is required to search for pivots and to rearrange the matrix accordingly. Note that for symmetric matrices only the lower or the upper part of the matrix need to be stored for elimination and backward/forward substitutions to solve the linear system.

3.4 **ITERATIVE METHODS.**

In any iterative method one begins with an initial approximation and then successively modifies the approximation according to some rules. To be useful the iteration must converge and it is not considered to be effective unless the convergence is rapid.

Suppose we wish to solve the system

$$Ax = b$$  \hspace{1cm} (3.4.1)

where $A$ is a square $n \times n$ matrix, $b$ is a given column matrix, and $x$ is an unknown column matrix. It will be assumed that $A$ is non-singular; hence a unique solution vector $x$ exists. This solution vector is given by

$$x = A^{-1}b$$  \hspace{1cm} (3.4.2)

A linear stationary iterative method for solving (3.4.1) is defined by [FORSYTHE, 1955]

$$x^{(n+1)} = Gx^{(n)} + k, \hspace{1cm} n \geq 0$$  \hspace{1cm} (3.4.3)

with $x^{(0)}$ arbitrary. It is easy to show that (3.4.3) will be consistent with (3.4.1), that is, $x^{(n+1)} = x$, where $x$ is the solution of (3.4.1), whenever $x^{(n)} = x$, if and only if
\[ x = (I-G) A^{-1} b. \quad (3.4.4) \]

It is also easy to show (Section 3.4.2) that a consistent method will converge to the solution of (3.4.1), i.e.

\[ \lim_{n \to \infty} x^{(n)} = x \quad (3.4.5) \]

for any vector \( b \) and for any starting vector \( x^{(0)} \) if and only if

\[ \rho(G) < 1 \quad (3.4.6) \]

Here \( \rho(G) \) is the spectral radius of \( G \). For a convergent method we define the rate of convergence \([\text{YOUNG, 1954}]\) as

\[ R_\infty(G) \equiv -\ln \rho(G) \quad (3.4.7) \]

This quantity is approximately inversely proportional to the number of iterations necessary for \( x^{(n)} \) to converge to \( x \), to within a specified tolerance. The larger the value of \( R(G) \), the more rapidly convergent the method is.

3.4.1 Basic Iterative Methods.

Several of the best known iterative methods are built around a partition of \( A \) into the form

\[ A = D - L - U \quad (3.4.8) \]

where \( D \) is the main diagonal elements of \( A \), \( -L \) is the strictly lower and \( -U \) is the strictly upper triangular elements of \( A \). It should be noted here that we assume the diagonal elements of matrix \( A \) do not vanish.
The equation (3.1.1) thus can be written as:

\[ D \mathbf{x} = (L+U) \mathbf{x} + \mathbf{b} \quad (3.4.9) \]

The Jacobi iteration is defined by

\[ D \mathbf{x}^{(n+1)} = (L+U) \mathbf{x}^{(n)} + \mathbf{b} \quad (3.4.10) \]

giving

\[ \mathbf{x}^{(n+1)} = D^{-1}(L+U) \mathbf{x}^{(n)} + D^{-1} \mathbf{b} \quad (3.4.11) \]

The matrix \( D^{-1}(L+U) \) is called the point Jacobi iteration matrix.

Related to the Jacobi method is the Simultaneous Over-relaxation method (JOR method). In this method the displacement vector

\[ d^{(n)} = x^{(n+1)} - x^{(n)} \]

of the JOR method is taken to be \( \omega \) times the displacement vector \( d^{(n)} \) defined by the Jacobi iteration.

By equation (3.4.10)

\[ Dd^{(n)} = D(x^{(n+1)} - x^{(n)}) = (L+U) x^{(n)} + b - Dx^{(n)} \quad (3.4.12) \]

hence the JOR iteration defined by

\[ d^{(n)} = \omega d^{(n)} \]

can be written as

\[ x^{(n+1)} - x^{(n)} = \omega D^{-1} \left\{ (L+U) x^{(n)} + b - D x^{(n)} \right\} \]

Therefore

\[ x^{(n+1)} = \omega D^{-1}(L+U) x^{(n)} + \omega D^{-1} b + (1-\omega)I x^{(n)} \]

or
The Gauss-Seidel iteration is defined by

\[ D\mathbf{x}^{(n+1)} = L\mathbf{x}^{(n+1)} + U\mathbf{x}^{(n)} + \mathbf{b} \]  

(3.4.14)

so that

\[ (D-L)\mathbf{x}^{(n+1)} = U\mathbf{x}^{(n)} + \mathbf{b} \]  

(3.4.15)

and

\[ \mathbf{x}^{(n+1)} = (D-L)^{-1} U\mathbf{x}^{(n)} + (D-L)^{-1} \mathbf{b} \]  

(3.4.16)

Since \( D-L \) is a non-singular matrix, (3.4.16) shows that the Gauss-Seidel point iteration matrix is \( (D-L)^{-1}U \). This iterative method has the computational advantage that it does not require the simultaneous storage of two approximations \( \mathbf{x}_1^{(n+1)} \) and \( \mathbf{x}_1^{(n)} \) in the course of computation as does the point Jacobi iteration method.

Related to the Gauss-Seidel iteration method is the Successive Over-relaxation method (SOR method). In this method, the displacement or correction vector \( \mathbf{d}^{(n)} = \mathbf{x}^{(n+1)} - \mathbf{x}^{(n)} \) of the SOR method is taken to be \( \omega \) times the displacement vector \( \mathbf{d}_1^{(n)} \) defined by the Gauss-Seidel iteration, therefore from equation (3.4.14) we have that

\[ D\mathbf{d}^{(n)} = D(\mathbf{x}^{(n+1)} - \mathbf{x}^{(n)}) = L\mathbf{x}^{(n+1)} + U\mathbf{x}^{(n)} + \mathbf{b} \]  

Hence the SOR iteration, defined by

\[ \mathbf{d}^{(n)} = \omega \mathbf{d}_1^{(n)} \]
can be written as

\[ \mathbf{x}^{(n+1)} - \mathbf{x}^{(n)} = \omega \mathbf{D}^{-1}(L \mathbf{x}^{(n+1)} + U \mathbf{x}^{(n)} + \mathbf{b} - D \mathbf{x}^{(n)}), \]

then

\[ \mathbf{x}^{(n+1)} = (I - \omega D^{-1}L)^{-1}\left\{(1-\omega)I + \omega D^{-1}U\right\}\mathbf{x}^{(n)} + (I - \omega D^{-1}L)^{-1}\omega D^{-1} \mathbf{b}, \]  

(3.4.17)

showing that the SOR point iteration matrix is

\[ \mathcal{L}_\omega = (I - \omega D^{-1}L)^{-1}\left\{(1-\omega)I + \omega D^{-1}U\right\} \]  

(3.4.18)

with \( \omega = 1 \), (i.e. \( \mathcal{L}_I \)) we have the Gauss-Seidel method.

### 3.4.2 Convergence of Point Iterative Methods.

As mentioned before, the general form of a stationary linear iterative method may be written in the form

\[ \mathbf{x}^{(n+1)} = G \mathbf{x}^{(n)} + \mathbf{k}, \]  

(3.4.19)

where \( G \) is the corresponding iteration matrix for the specific method, and \( \mathbf{k} \) is a column vector of known values. This equation was derived from the original equation (3.4.1) by rearranging them into the form

\[ \mathbf{x} = G \mathbf{x} + \mathbf{k} \]  

(3.4.20)

i.e. the unique solution of \( n \) linear equations \( A\mathbf{x} = \mathbf{b} \) is the solution of equation (3.4.20). (Alternatively, if we assume the iteration is convergent then, by (3.4.19)
\[ \lim_{n \to \infty} x^{(n+1)} = \lim_{n \to \infty} x^{(n)} = x, \]

hence

\[ \mathbf{x} = G \mathbf{x} + k \).

The error vector \( \mathbf{e}^{(n)} \) of the \( n \)th approximation to the exact solution \( \mathbf{x} \) is defined by

\[ \mathbf{e}^{(n)} = \mathbf{x} - \mathbf{x}^{(n)}, \quad (3.4.21) \]

therefore, equations (3.4.19) and (3.4.20) show that

\[ \mathbf{e}^{(n+1)} = G \mathbf{e}^{(n)} \quad \quad (3.4.22) \]

Hence

\[ \mathbf{e}^{(n)} = G \mathbf{e}^{(n-1)} = \ldots = G^n \mathbf{e}^{(0)}, \quad (3.4.23) \]

where \( \mathbf{e}^{(0)} = \mathbf{x} - \mathbf{x}^{(0)} \) and \( \mathbf{x}^{(0)} \) is an arbitrary but known set of initial values. The sequence of iterative values \( x^{(1)}, x^{(2)}, \ldots, x^{(n)}, \ldots \) will converge to \( \mathbf{x} \) as \( n \) tends to infinity if

\[ \lim_{n \to \infty} \mathbf{e}^{(n)} = 0. \quad (3.4.24) \]

Therefore, the iterative process will converge if and only if

\[ \lim_{n \to \infty} \mathbf{e}^{(n)} = 0. \quad (3.4.24) \]

Assume that the matrix \( G \) of order \( n \) has \( n \) linearly independent eigenvectors \( \mathbf{v}_i \), \( i = 1, 2, \ldots, n \), the arbitrary error vector \( \mathbf{e}^{(0)} \) can be expressed uniquely as a linear combination of the form
where the $\alpha_i; \ i = 1, 2, \ldots, n$ are scalars, hence

$$e(1) = G e(0) = \sum_{i=1}^{n} \alpha_i G v_i , \quad (3.4.27)$$

but $G v_i = \lambda_i v_i$, where $\lambda_i$ is the eigenvalue corresponding to $v_i$, therefore,

$$e(1) = \sum_{i=1}^{n} \alpha_i \lambda_i v_i . \quad (3.4.28)$$

Similarly we have

$$e(n) = \sum_{i=1}^{n} \alpha_i \lambda_i^n v_i . \quad (3.4.29)$$

So $e(n)$ will tend to the null vector as $n$ tends to infinity, for any arbitrary $e(0)$ if and only if $|\lambda_i| < 1$ for all $i$. i.e. the iteration will converge for any arbitrary $x(0)$ if and only if the spectral radius $\rho(G)$ of $G$ is less than unity.

**Corollary 3.4.1.** A sufficient condition for convergency is that

$$||G|| < 1 . \quad (3.4.30)$$

Since

$$G v_i = \lambda_i v_i \quad (3.4.31)$$

we have

$$||G v_i|| = ||\lambda_i v_i|| = |\lambda_i| ||v_i|| . \quad (3.4.32)$$
But for any matrix norm that is compatible with a vector norm \( \|v_i\| \) we have,

\[
\|G v_i\| \leq \|G\| \|v_i\| .
\]

(3.4.33)

Therefore

\[
|\lambda_i| \|v_i\| \leq \|G\| \|v_i\|
\]

(3.4.34)

so

\[
|\lambda_i| \leq \|G\|
\]

(3.4.35)

It follows from (3.4.35) that a sufficient condition for convergence is that \( \|G\| < 1 \). It is not a necessary condition because the norm of \( G \) can exceed unity, even when \( \rho(G) < 1 \).

Theorem 3.4.1 [VARGA, 1962].

Let \( A = (a_{ij}) \) be a strictly or irreducibly diagonally dominant \((n \times n)\) complex matrix. Then, both the associated point Jacobi and point Gauss-Seidel matrix are convergent, and the Jacobi iterative method and Gauss-Seidel iterative method for the matrix problem \( A \mathbf{x} = \mathbf{b} \) are convergent for any initial approximation vector \( \mathbf{x}(0) \).

3.4.3 Rate of Convergence.

Even if a method converges, it may converge too slowly to be of practical value. Therefore, it is essential to determine the effectiveness of each method. To accomplish this we should consider both the work required per iteration and the number of iterations.
necessary for convergence to a specified accuracy.

Assume that the eigenvalues of the iteration matrix $G$ are order as follows:

$$|\lambda_1| > |\lambda_2| > \ldots > |\lambda_n|,$$  \hspace{1cm} (3.4.36)

and that the matrix $G$ has $n$ linearly independent eigenvectors $v_i$, $i = 1, 2, \ldots, n$, namely.

Now equation (3.4.29), i.e.

$$e^{(k)} = \sum_{i=1}^{n} \alpha_i \lambda_i^k v_i,$$

can be rewritten as:

$$e^{(k)} = \lambda_1^k (\alpha_1 v_1) + \left(\frac{\lambda_2}{\lambda_1}\right)^k \alpha_2 v_2 + \ldots + \left(\frac{\lambda_n}{\lambda_1}\right)^k \alpha_n v_n.$$

\hspace{1cm} (3.4.37)

For large values of $k$ we have that

$$e^{(k)} \approx \lambda_1^k \alpha_1 v_1,$$  \hspace{1cm} (3.4.38)

similarly,

$$e^{(k+1)} \approx \lambda_1^{k+1} \alpha_1 v_1,$$  \hspace{1cm} (3.4.39)

so

$$e^{(k+1)} \approx \lambda_1 e^{(k)}.$$  \hspace{1cm} (3.4.40)

Hence

$$\ln \left( \frac{\|e^{(k)}\|}{\|e^{(k+1)}\|} \right) = \ln \frac{1}{\lambda_1} = - \ln (\rho(G)), $$  \hspace{1cm} (3.4.41)
Hence (3.4.41) shows that \(- \ln(p(G))\), for \(k\) sufficiently large, measures the number of decimal digits of accuracy gained per iteration. Since \(0 < p(G) < 1\) for a convergent iteration, the smaller the value of \(p(G)\), the greater will be the number of decimal digits of accuracy gained per iteration. The number \((- \ln(p(G)))\) provides a measure for the comparison of the rates of convergence of different iterative methods, provided \(k\) is sufficiently large. For this reason \(- \ln(p(G))\) is defined to be the asymptotic rate of convergence, and is denoted by \(R_\infty(G)\). The average rate of convergence \(R_k(G)\) after \(k\) iterations is defined by

\[
R_k(G) = -\frac{1}{k} \ln \|G^k\|.
\]  

(3.4.42)

It can be proved that [Varga, 1962]

\[
R_\infty(G) = \lim_{k \to \infty} R_k(G).
\]  

(3.4.43)

In comparing iterative methods, we can use (3.4.23) in the following manner,

\[
\|e^{(k)}\| \leq \|G^k\| \|e^{(0)}\|.
\]  

(3.4.44)

Hence after \(k\) iterations we have

\[
\|e^{(k)}\| / \|e^{(0)}\| \leq \|G^k\|.
\]  

(3.4.45)

if \(G\) is symmetric then

\[
\|G^k\| = p(G)^k = \rho(G)^k.
\]  

(3.4.46)
Suppose we wish to carry out the iterative process until

$$\frac{\|e^{(k)}\|}{\|e^{(0)}\|} \leq \varepsilon,$$

where $\varepsilon$ is a small positive number. Then from (3.4.45) we should choose $k$ such that

$$\|G^k\| \leq \varepsilon.$$  

Therefore, we have

$$\ln \|G^k\| \leq \ln \varepsilon,$$

and from (3.4.42), (3.4.43), if $k$ is large, we obtain

$$k \approx -\frac{\ln \varepsilon}{R_{\infty}(G)}.$$  

(3.4.50)

If $G$ is symmetric, then (3.4.50) becomes

$$k \geq -\frac{\ln \varepsilon}{R_{\infty}(G)}.$$  

(3.4.51)

Unless $G$ is symmetric, (3.4.50) may be very crude indeed, as pointed out by [VARGA, 1962]. Nonetheless, it does give a relatively easily computed number which may be used to compare the rate of convergence of certain iterative methods.

In order to compare different iterative methods and their related rates of convergence, we consider the following Theorem.

* $\approx$ means "is approximately equal to" for large $k"."
Theorem 3.4.2. [STEIN AND ROSENBERG, 1948]

Let the Jacobi matrix $B = L + U$, (where $L$ and $U$ are strictly lower triangular and upper triangular matrices respectively) be a non-negative $n \times n$ matrix with zero diagonal elements and let $\mathcal{L}_1$ be the Gauss-Seidel iteration matrix (i.e. $\mathcal{L}_1 \equiv (I-L)^{-1}U$). Then one and only one of the following mutually exclusive relations is valid;

1. $\rho(B) = \rho(\mathcal{L}_1) = 0$.
2. $0 < \rho(\mathcal{L}_1) < \rho(B) < 1$.
3. $1 = \rho(B) = \rho(\mathcal{L}_1)$.
4. $1 < \rho(B) < \rho(\mathcal{L}_1)$.

So, the Jacobi point iterative matrix $B$ and the Gauss-Seidel point iterative matrix $\mathcal{L}_1$ are either both convergent, or both divergent.

If the non-negative Jacobi matrix $B$ is such that $0 < \rho(B) < 1$ then

$$R_\infty(\mathcal{L}_1) > R_\infty(B)$$

i.e. the point Gauss-Seidel iterative method is asymptotically faster than the point Jacobi method.

We now give some useful results on the convergence of iteration schemes.

Theorem 3.4.3. [YOUNG, 1971].

If the Jacobi method converges, then the JOR method will also converge for all values of $\omega$ in the interval $(0, 1]$.
The following theorem is due to [KAHAN, 1958].

**Theorem 3.4.4.**

For the SOR iteration matrix $L_\omega$ and for all real $\omega$ we have that

$$\rho(L_\omega) > |\omega - 1|$$

and furthermore, the SOR method will converge for any value of $\omega$ defined as:

$$0 < \omega < 2.$$

**Theorem 3.4.5. [YOUNG, 1971].**

Let $A$ be an irreducible matrix with weak diagonal dominance, then

(i) The Jacobi method converges, and the JOR method converges for

$$0 < \omega \leq 1.$$

(ii) Both the Gauss-Seidel and the SOR methods converge for

$$0 < \omega \leq 1.$$

**Theorem 3.4.6. [YOUNG, 1971].**

Let $A$ be a real, symmetric, non-singular matrix with positive diagonal elements, then the JOR method converges if $A$ and $2\omega^{-1}D - A$ are positive definite matrices, where $D = \text{diag}(A)$.

It follows from the above Theorem that if $A$ and $2D - A$ are positive definite then the Jacobi point iterative method converges.
Theorem 3.4.7. \cite{OSTROWSKI, 1954}.

Let $A$ be a symmetric matrix with positive diagonal elements, then the SOR method converges if and only if $A$ is positive definite and $0 < \omega < 2$.

However, it is not easy to find the value of $\omega$ which yields the largest rate of convergence. On the other hand, such a value of $\omega$ can be found in certain special cases, which we shall discuss in the next section.

We now consider the iterative schemes based on the theory of "regular splitting".

Definition (3.4.1).

Consider the real $n \times n$ matrix $A$ which is partitioned in the form

$$A = M - N,$$ \hspace{1cm} (3.4.52)

where $M$ and $N$ are also $n \times n$ matrices. Then, the expression (3.4.52) is said to be a regular splitting of matrix $A$ if

(i) $M$ is non-singular with $M^{-1} \succ 0$,

(ii) $N \succ 0$.

Associated with this splitting of the matrix $A$ we have an iterative method of form

$$M \mathbf{x}^{(n+1)} = N \mathbf{x}^{(n)} + \mathbf{b}.$$ \hspace{1cm} (3.4.53)

Note that all the basic iterative methods considered so far can be described from this point of view, for example,
When $M = D$ and $N = L + U$ we have the point Jacobi method, or

When $M = 1/\omega \ (D - \omega L)$ and $N = 1/\omega \ (\omega U + (1 - \omega)D)$ we have the point SOR method.

**Theorem 3.4.8.** [VARGA, 1962].

If $A = M - N$ is a regular splitting of the matrix $A$ and $A^{-1} > 0$ then,

$$\rho(M^{-1}N) = \frac{\rho(A^{-1}N)}{1 + \rho(A^{-1}N)} < 1 .$$  (3.4.54)

Therefore, the iterative method of (3.4.53) converges for any initial vector $x^{(0)}$.

Suppose the matrix $A$ is a Stieltjes matrix (Definition 2.2.3) then we have the following corollary.

**Corollary 3.4.2.** Let $A = M - N$ be a regular splitting of the Stieltjes matrix $A$, where $N$ is a real symmetric matrix, then we have

$$\rho(M^{-1}N) < \frac{\rho(N) \cdot \rho(A^{-1})}{1 + \rho(N) \cdot \rho(A^{-1})} < 1 .$$

It can be proved that if $A$ is an $M$-matrix (Definition 2.2.4) and we choose the matrix $M$ from $A$ by setting certain off-diagonal elements of $A$ to zero, then $A = M - N$ is a regular splitting of $A$ and $\rho(M^{-1}N) < 1$. 
Suppose that

\[ A = M_1 - N_1 = M_2 - N_2, \]  

(3.4.56)

are two regular splittings of \( A \), where \( A^{-1} > 0 \), if \( N_2 > N_1 > 0 \)

where, neither \( N_1 \) nor \( N_2 - N_1 \) is the null matrix then we have

\[ 0 < \rho(M_1^{-1} N_1) < \rho(M_2^{-1} N_2) < 1 \]  

(3.4.57)

and

\[ R_\infty(M_2^{-1} N_2) < R_\infty(M_1^{-1} N_1). \]  

(3.4.58)

Theorem 3.4.8a. [VARGA, 1960].

Let \( A \) be a Stieltjes matrix and \( A = M_1 - N_1 = M_2 - N_2 \), two regular splittings of \( A \) with \( N_1 > N_2 > 0 \). If \( N_1 \) and \( N_2 \) are symmetric and \( \overline{\mu}(A^{-1} N_2) \to \infty \) then

\[ \frac{R_\infty(M_2^{-1} N_2)}{R_\infty(M_1^{-1} N_1)} \to \frac{\overline{\mu}(A^{-1} N_1)}{\overline{\mu}(A^{-1} N_2)}. \]

Moreover, if \( N_1 \) commutes with \( A \) (i.e. \( AN_1 = N_1 A \)), then

\[ \frac{\overline{\mu}(A^{-1} N_1)}{\overline{\mu}(A^{-1} N_2)} \to \frac{\overline{\mu}(N_1)}{\overline{\mu}(N_2)} > 1. \]

3.4.4 The Optimum Acceleration Parameter for the SOR Method.

A problem of paramount importance associated with the SOR method is the determination of a suitable value for the acceleration parameter
Ideally, we would like to have the optimum value of $\omega$ denoted by $\omega_b$ which minimises the spectral radius of the SOR iteration matrix and thereby maximises the rate of convergence of the method. At the present time no formula exists for the determination of $\omega_b$ for an arbitrary set of linear equations. But, it can be calculated for many of the difference equations approximating second order partial differential equations because their matrices are of a special type which possess property (A), and the significance of this was first revealed by [YOUNG, 1954]. He proved that when a matrix possesses property (A) then it can be transformed into what he termed a consistently ordered matrix. Under this condition the eigenvalues $\lambda$ of the point SOR iteration matrix $L_\omega$ associated with $A$ are related to the eigenvalues $\mu$ of the corresponding point Jacobi iteration matrix $B$ of $A$ by the equation

\[(\lambda + \omega - 1)^2 = \lambda \omega^2 \mu^2. \quad (3.4.59)\]

From this it can be seen that

\[\lambda^4 = \omega \mu \pm \sqrt{\omega^2 \mu^2 - 4(\omega - 1)} \quad (3.4.60)\]

From equation (3.4.29) it can be seen that the rate of convergence is dependent on $\lambda$ and so to optimise the rate of convergence, $\lambda$, the eigenvalue of maximum modulus of the SOR iteration matrix $L_\omega$ must be minimised. This is achieved by making the square root in equation (3.4.60) equal to zero for $\mu$, the spectral radius of the point Jacobi iteration matrix $B$, i.e.
\[ \frac{\omega^2 - \mu^2}{\mu^2} = 4(\omega - 1) \quad (3.4.61) \]

which yields the result,

\[ \omega_b = \frac{2}{1 + \sqrt{1 - \mu^2}} \quad (3.4.62) \]

Now, since the Gauss-Seidel method is the same as SOR with \( \omega = 1 \), then from equation (3.4.60) we obtain

\[ \rho(G) = \rho(J)^2 \quad (3.4.63) \]

where \( \rho(G) \) and \( \rho(J) \) are the spectral radii of the Gauss-Seidel and point Jacobi iteration matrices respectively. Using (4.3.63) it is clear from (3.4.62) that

\[ \omega_b = \frac{2}{1 + \sqrt{1 - \rho(G)}} \quad (3.4.64) \]

The estimation of \( \omega_b \) depends on whether \( \rho(J) \) or \( \rho(G) \) can be estimated. Several methods have been suggested by [CARRE, 1961] and [VARGA, 1962], one of which is the power method that may be described as follows.

Assuming the matrix of the finite difference equations is consistently ordered and has property (A), calculate the sequence of approximations \( x^{(1)}, x^{(2)}, \ldots, x^{(i)} \) to the solution of the system of equations \( Ax = b \) by the Gauss-Seidel method and then we have

\[ \rho(G) = \lim_{i \to \infty} \frac{\|x^{(i)}\|}{\|x^{(i-1)}\|} \quad (3.4.65) \]
where \( d^{(i)} \) is defined as

\[
d^{(i)} = \mathbf{x}^{(i)} - \mathbf{x}^{(i-1)} \tag{3.4.66}
\]

and

\[
\|d^{(i)}\| = \left\{ \sum_{j=1}^{n} (x_j^{(i)} - x_j^{(i-1)})^2 \right\}^{1/2}. \tag{3.4.67}
\]

Thus, using the power method we can approximate \( \rho(G) \), which, in turn, can be substituted into equation (3.4.64) to give an estimate of \( \omega_b \), the optimum acceleration factor.

### 3.4.5 Consistent Ordering.

The concept of consistent ordering is central to the theory of the SOR iterative method for solving the equation \( A \mathbf{x} = \mathbf{b} \), because, as mentioned before, the calculation of the optimum acceleration parameters is presently possible only for consistently ordered matrices.

The earliest definition was due to [Young, 1950] and was related to a class of matrices whose members possessed what was termed property (A).

**Definition (3.4.2).** A matrix \( A \) of order \( N \) has property (A) if there exist two disjoint subsets \( S \) and \( T \) of \( W \), the set of the first \( N \) positive integers, such that \( S \cup T = W \) and if \( a_{i,j} \neq 0 \) then either \( i \neq j \) or \( i \in S \) and \( j \in T \) or \( i \in T \) and \( j \in S \). If \( S \) or \( T \) is empty, then \( A \) is, of course, diagonal.

If a matrix \( A \) has property (A), it is always possible to rearrange the rows and corresponding columns of \( A \), in order to
obtain the matrix \( \tilde{A} \) which has either the block tridiagonal form

\[
\tilde{A} = \begin{bmatrix}
D_1 & A_1 & 0 \\
B_1 & D_2 & A_2 \\
B_2 & D_3 & A_3 \\
& & & & \ddots \\
0 & & & & & D_{k-2} & D_{k-1} & A_{k-1} \\
& & & & & B_{k-1} & D_k
\end{bmatrix}
\]  

(3.4.68)

or the form

\[
\tilde{A} = \begin{bmatrix}
D_1 & F_1 \\
E_1 & D_2
\end{bmatrix}
\]  

(3.4.69)

where \( D_i \) are square diagonal matrices not necessarily of the same order. i.e. there exists a permutation matrix \( P \) such that \( \tilde{A} = P^{-1} AP \) has either the form (3.4.68) or the form (3.4.69).

It can be proved that if \( A \) has property (A), then for any permutation \( P \) the matrix \( \tilde{A} = P^{-1} AP \) has property (A).

Also it can be seen that, a matrix \( A \) has property (A) if and only if there exists a row vector \( \gamma = (\gamma_1, \gamma_2, \ldots, \gamma_n) \) with integral components such that if \( a_{i,j} \neq 0 \) and \( i \neq j \), then \( |\gamma_i - \gamma_j| = 1 \). The vector \( \gamma \) with these properties is said to be an ordering vector for the matrix \( A \).
Definition (3.4.3) [YOUNG, 1954].

A matrix \( A \) is consistently ordered if there exists an ordering vector \( \mathbf{y} = (y_1, y_2, \ldots, y_n)^T \) such that \( a_{i,j} \neq 0, i \neq j \) implies

\[
(i) \quad |y_i - y_j| = 1 \quad (3.4.70)
\]

\[
(ii) \quad \begin{cases} y_i < y_j & \text{if } i < j \\ y_i > y_j & \text{if } i > j \end{cases} \quad (3.4.71)
\]

It follows that if the components of \( \mathbf{y} \) are in ascending order of magnitude and \( \mathbf{y} \) is an ordering vector for the matrix \( A \) then \( A \) is consistently ordered.

Theorem 3.4.9. [YOUNG, 1971].

If the matrix \( A \) has property (A) then there exists a permutation matrix \( P \) such that \( \tilde{A} = P^{-1}AP \) is consistently ordered.

It can be verified that a block tridiagonal matrix of the form (3.4.68) is consistently ordered, the converse is not necessarily true.

Considering the concept of property (A) and consistently ordered matrices, the following results can be established [YOUNG, 1954].

Theorem 3.4.10.

\( \mu \) is an eigenvalue of the Jacobi iteration matrix \( B \) if,

\[
(i) \quad \text{The coefficient matrix } A \text{ is consistently ordered,}
\]
(ii) the acceleration parameter $\omega$ is in the range $(0, 2)$,

(iii) $\mu$ satisfies the following relation

\[(\lambda + \omega - 1)^2 = \omega^2 \lambda \mu^2\] (3.4.72)

where $\lambda$ is a non-zero eigenvalue of SOR iteration matrix $L_\omega$.

Conversely, if $\mu$ is an eigenvalue of $B$ and if $\lambda$ satisfies (3.4.72), then $\lambda$ is an eigenvalue of $L_\omega$.

Theorem (3.4.11).

If $A$ is symmetric, positive definite and is consistently ordered and if

\[\omega_b = \frac{2}{1 + \sqrt{1 - \mu^2}}\] (3.4.73)

where $\mu$ is the spectral radius of Jacobi iteration matrix $B$ (i.e. $\mu = \rho(B)$.) Then,

\[\rho(L_{\omega_b}) < \rho(L_\omega), \quad \omega \neq \omega_b\] (3.4.74)

and

\[\rho(L_\omega) = \omega - 1 \quad \text{for} \quad \omega_b \leq \omega < 2\] (3.4.75)
CHAPTER 4

NEW PARALLEL STRATEGIES FOR SOLVING
LINEAR SYSTEMS OF EQUATIONS
4.1 INTRODUCTION.

It has been shown in the previous chapters that the major problem in the solution of Elliptic partial differential equations by both finite element and finite difference methods is that of solving the corresponding set of linear discretised equations. With the emergence of parallel computers in recent years much work and attention has been concentrated on designing efficient methods suitable to solve linear systems on this type of computer.

In this chapter, we present a thorough description and analysis of some new parallel methods for the solution of general linear systems. The new parallel strategies known as Quadrant Interlocking Factorisation (Q.I.F.) for solving linear systems directly and Quadrant Interlocking Iterative (Q.I.I.) methods for solving linear systems iteratively consider the interlocking matrix quadrants instead of the standard L and U triangular components of A.

In Section (4.2) we shall represent the alternative (Q.I.F.) methods developed by Evans and Hatzopoulos [1980], Evans and Hadjidimos [1980], Evans, Hadjidimos and Noutsos [1981] and Shanehchi and Evans [1981] together with their relative error and computational cost analysis. In Section (4.3) we present the new splitting which makes the corresponding Q.I. iterative methods suitable for use on parallel computers.
4.2 **DIRECT METHODS.**

In this section we shall study alternative direct methods developed to date for the solution of linear systems. The methods correspond to the standard sequential algorithms such as, LU decomposition, LDU factorisation, the Gauss–Jordan elimination method for general square matrices and the Choleski method for a symmetric positive definite coefficient matrix. For the application of these methods and their suitability for different types of parallel computers, see for example, SHANEHCHI, [1980], SHANEHCHI & EVANS, [1981], STONE, [1973], TRAUB, [1971], FLYNN, [1966], KUCK, [1973], and DUNBAR, [1978].

### 4.2.1 The Factorisation Process of the Q.I.F. Method.

Consider the following set of linear equations

\[ A \mathbf{x} = \mathbf{b} \quad , \]

where \( A \) is a non-singular \((n \times n)\) matrix, \( \mathbf{x} \) is an unknown \((n \times 1)\) column matrix and \( \mathbf{b} \) is a given \((n \times 1)\) column matrix. In this method, the matrix \( A \) is factorised into two matrices \( W \) and \( Z \) \([EVANS & HATZOPoulos, 1980]\) such that

\[ A = W \ast Z \]  

(4.2.2)

where the matrices \( W \) and \( Z \) are of the forms shown below;

\[ W = \begin{bmatrix} W_1 & W_2 & \cdots & W_n \end{bmatrix} \]

and

\[ Z^T = \begin{bmatrix} Z_1 & Z_2 & \cdots & Z_n \end{bmatrix} \]  

(4.2.3)
where $W_i$ and $Z_i \mid i = 1(1) n$ are the column vectors of the matrices $W$ and $Z^T$ which are of the following general forms:

(i) For $n$ odd,

$$W_i \equiv \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 & w_{i+1, i} & \cdots & w_{n-i, i} & 0 & \cdots & 0 \end{bmatrix}^T, \quad i = 1(1) \frac{n-1}{2}$$

$$Z_i \equiv \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \end{bmatrix}^T, \quad i = \frac{n+1}{2}$$

$$W_i \equiv \begin{bmatrix} 0 & 0 & \cdots & 0 & w_{n-i+2, i} & \cdots & w_{i-1, i} & 1 & 0 & \cdots & 0 \end{bmatrix}^T, \quad i = \frac{n+3}{2} (1) n$$

(4.2.4a)

and

$$Z_i \equiv \begin{bmatrix} 0 & 0 & \cdots & 0 & z_{i, n-i+1} & \cdots & z_{i, i} & 0 & \cdots & 0 \end{bmatrix}^T, \quad i = \frac{n+3}{2} (1) n$$

(4.2.4b)

(ii) For $n$ even,

$$W_i \equiv \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 & w_{i+1, i} & \cdots & w_{n-i, i} & 0 & \cdots & 0 \end{bmatrix}^T, \quad i = 1(1) \frac{n}{2} - 1$$

$$W_i \equiv \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \end{bmatrix}^T, \quad i = \frac{n}{2}, \frac{n}{2} + 1$$

$$W_i \equiv \begin{bmatrix} 0 & 0 & \cdots & 0 & w_{n-i+2, i} & \cdots & w_{i-1, i} & 1 & 0 & \cdots & 0 \end{bmatrix}^T, \quad i = \frac{n}{2} + 2(1) n$$

(4.2.5a)

and

$$Z_i \equiv \begin{bmatrix} 0 & 0 & \cdots & 0 & z_{i, n-i+1} & \cdots & z_{i, i} & 0 & \cdots & 0 \end{bmatrix}^T, \quad i = \frac{n}{2} + 1(1) n$$

(4.2.5b)
As examples, for \( n = 5 \), the matrices \( W \) and \( Z \) have the following forms:

\[
W = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
w_{2,1} & 1 & 0 & 0 & w_{2,5} \\
w_{3,1} & w_{3,2} & 1 & w_{3,4} & w_{3,5} \\
w_{4,1} & 0 & 0 & 1 & w_{4,5} \\
0 & 0 & 0 & 0 & 1
\end{bmatrix}, \quad Z = \begin{bmatrix}
z_{1,1} & z_{1,2} & z_{1,3} & z_{1,4} & z_{1,5} \\
0 & z_{2,2} & z_{2,3} & z_{2,4} & 0 \\
0 & 0 & z_{3,3} & 0 & 0 \\
0 & z_{4,2} & z_{4,3} & z_{4,4} & 0 \\
z_{5,1} & z_{5,2} & z_{5,3} & z_{5,4} & z_{5,5}
\end{bmatrix}
\]

whilst for \( n = 6 \), we have

\[
W = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
w_{2,1} & 1 & 0 & 0 & 0 & w_{2,6} \\
w_{3,1} & w_{3,2} & 1 & 0 & w_{3,5} & w_{3,6} \\
w_{4,1} & w_{4,2} & 0 & 1 & w_{4,5} & w_{4,6} \\
w_{5,1} & 0 & 0 & 0 & 1 & w_{5,6} \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}, \quad Z = \begin{bmatrix}
z_{1,1} & z_{1,2} & z_{1,3} & z_{1,4} & z_{1,5} & z_{1,6} \\
0 & z_{2,2} & z_{2,3} & z_{2,4} & z_{2,5} & 0 \\
0 & 0 & z_{3,3} & z_{3,4} & 0 & 0 \\
0 & 0 & z_{4,3} & z_{4,4} & 0 & 0 \\
0 & z_{5,2} & z_{5,3} & z_{5,4} & z_{5,5} & 0 \\
z_{6,1} & z_{6,2} & z_{6,3} & z_{6,4} & z_{6,5} & z_{6,6}
\end{bmatrix}
\]

From the given forms of the matrices \( W \) and \( Z \) in (4.2.4) or (4.2.5) and the equality (4.2.2) we can obtain the elements of
these matrices in $\left\lceil \frac{n-1}{2} \right\rceil$ steps where $\left\lceil \frac{n-1}{2} \right\rceil$ represents the largest integer number which is $\leq \frac{n-1}{2}$. It can easily be seen that the values of the elements of the first and last rows of the matrix $Z$ are as follows:

$$\begin{align*}
z_{1,i} &= a_{1,i} \\
z_{n,i} &= a_{n,i}
\end{align*}$$

for $i = 1(1)n$. \hspace{1cm} (4.2.6)

The elements of the first and last columns of the matrix $W$ are then evaluated by solving $(n-2)$ sets of $(2 \times 2)$ linear systems given by

$$\begin{align*}
z_{1,1} w_{i,1} + z_{n,1} w_{i,n} &= a_{i,1} \\
z_{1,n} w_{i,1} + z_{n,n} w_{i,n} &= a_{i,n}
\end{align*}$$

for $i = 2(1)n-1$. \hspace{1cm} (4.2.7)

Finally, to complete this step of the factorisation process the elements of the matrix $A$ are modified by the following formula:

$$a_{i,j} = a_{i,j} - w_{i,1} z_{1,j} - w_{i,n} z_{n,j} \quad i,j = 2(1)n-1 \hspace{1cm} (4.2.8)$$

In general, at $i^{th}$ step of the factorisation process we have the relationships,

$$\begin{align*}
z_{i,j} &= a_{i,j} \\
z_{n-i+1,j} &= a_{n-i+1,j}
\end{align*}$$

for $j = i(1)n-i+1$. \hspace{1cm} (4.2.9)

and the solution of the $(2 \times 2)$ linear systems:
\[
\begin{align*}
    z_{i,j}w_{j,i} + z_{n-i+1,j}w_{j,n-i+1} &= a_{j,i}, & j &= i+1(1)n-i, \\
    z_{i,n-i+1}w_{j,i} + z_{n-i+1,n-i+1}w_{j,n-i+1} &= a_{j,n-i+1}
\end{align*}
\]  
(4.2.11)

to give the unknown quantities \( w_{j,i} \) and \( w_{j,n-i+1} \) for \( j = i+1(1)n-i \),
and the modified \( a_{i,j} \)'s are evaluated from the formula:

\[
a_{k,\ell} = a_{k,\ell} - w_{k,i}z_{i,\ell} - w_{k,n-i+1}z_{n-i+1,\ell}, & k,\ell = i+1(1)n-i.
\]  
(4.2.12)

In order to solve a set of linear equations by the Q.I.F. method it can be seen from (4.2.1) and (4.2.2) that the system

\[
(WZ)x = b
\]  
(4.2.13)

can be solved instead of (4.2.1), where we need to solve two related and simpler linear systems of the form:

\[
W y = b
\]  
(4.2.14)

and

\[
Z x = y
\]  
(4.2.15)

The system (4.2.14) is first solved for the intermediate vector \( y \) and then the final solution of the system (4.2.1) can be obtained by solving the linear system (4.2.15) for \( x \).

The solution of linear system (4.2.14) can be obtained in \( \left[ \frac{n-1}{2} \right] \) steps where from the structure of the matrix \( W \) it can be seen that \( y_1 \) and \( y_n \) are evaluated first, then \( y_2, y_{n-1} \) and so on working in pairs from the top and rear of the vector \( y \). The following procedure typifies the step \( i \) (\( i = 1,2,\ldots, \left[ \frac{n-1}{2} \right] \)) of the
solution process:

\[ \begin{align*}
\text{a)} \quad & \begin{cases}
y_i = b_i \\
y_{n-i+1} = b_{n-i+1}
\end{cases} \\
\text{and} \\
\text{b)} \quad & b_j = b_j - w_{j,i} y_i - w_{j,n-i+1} y_{n-i+1}, \quad j = i+1(1)n-i
\end{align*} \]

(4.2.16)

and then we proceed to the next step.

For the solution of the linear system (4.2.15), we proceed as follows:

If \( n \) is odd, we can find that

\[ x_{\frac{n}{2}} = y_{\frac{n}{2}} / z_{\frac{n}{2}}, \quad \ell = \frac{n+1}{2} \]

(4.2.17)

and in preparation for the next step we have

\[ y_j = y_j - x_{\frac{n}{2}} * z_{\frac{n}{2}}, \quad j = 1(1)n \text{ and} \]

\[ j \neq \frac{n+1}{2}. \]

(4.2.18)

It can be seen that the elements of the vector \( x \) can be obtained in pairs by solving \( \frac{n-1}{2} \) sets of \((2\times2)\) linear systems in \( \frac{n-1}{2} \) distinct steps. In general, for the \( i^{th} \) step we solve the following \((2\times2)\) linear system:

\[ \begin{align*}
\text{a)} \quad & \begin{cases}
z_{i,i} x_i + z_{i,n-i+1} x_{n-i+1} = y_i \\
z_{n-i+1,i} x_i + z_{n-i+1,n-i+1} x_{n-i+1} = y_{n-i+1}
\end{cases}, \quad i = \ell-1(-1)1
\end{align*} \]

(4.2.19)
Furthermore, to compute \( x_i \) and \( x_{n-i+1} \), we then set

\[
y_j = y_j - x_i z_{j,i} - x_{n-i+1} z_{j,n-i+1}
\]

\[
i = \frac{j-1}{1(-1)} \quad \text{and} \quad j = \frac{n-1}{1(1)} n
\]

and proceed to the next step.

However, if \( n \) is even, then all the components of the vector \( x \) are found in pairs. To find all the pairs, the linear system (4.2.19) and the formula (4.2.20) are executed respectively in turn for \( i = \left\lfloor \frac{n}{2} \right\rfloor (-1) \).

4.2.2 The Computational Cost Analysis of the Q.I.F. Method.

In order to analyse the computational cost of the Q.I.F. method we adopt a similar procedure to that described in Shanehchi and Evans, [1981]. For this, we consider two different strategies which may be used to solve the set of \((2*2)\) linear equations encountered during the factorisation and the solution process.

1. Cramer's Rule: Each of the \((2*2)\) linear systems in (4.2.11) is solved by the following procedure.

1.a Define the quantities \( T_1, T_2 \) and \( T_3 \) (say) as:

\[
T_1 = z_{i,i} \ast z_{n-i+1,n-i+1} - z_{n-i+1,1} \ast z_{i,n-i+1}
\]

\[
T_2 = a_{j,i} \ast z_{n-i+1,n-i+1} - a_{j,n-i+1} \ast z_{n-i+1,i}
\]

\[
T_3 = a_{j,i} \ast z_{i,n-i+1} - a_{j,n-i+1} \ast z_{i,i}
\]
1.6 Compute \( w_{j,i} = \frac{T_2}{T_1} \) and \( w_{j,n-i+1} = \frac{T_3}{T_1} \).

It is clear from the definition of \( T_1 \), that if any of the quantities

\[
T_1 \equiv \det \begin{bmatrix}
b_{i,i} & b_{i,n-i+1} \\
b_{n-i+1,i} & b_{n-i+1,n-i+1}
\end{bmatrix} = 0 ,
\tag{4.2.21}
\]

then the method will break down. In order to avoid such an event we can adopt two well established pivotal strategies used in the well known standard elimination methods. These are as follows:

(i) Partial Pivoting: This corresponds to the partial pivoting strategy used in the Gaussian elimination process. In this strategy row interchanges are made wherever necessary to result in the highest determinantal value defined in (4.2.21). This method solves the linear system \( \mathbf{PAx} = \mathbf{Ph} \) instead of (4.2.1) where \( P \) is a permutation matrix. In order to locate the pivot at the \( i^{th} \) stage of the factorisation process a search is carried out amongst the elements \( a_{j,i}, a_{j,n-i+1} \) to find the value of \( \alpha \) defined as

\[
\alpha = \max \left\{ \left| \frac{a_{p,i}}{a_{q,i}} \right|, \frac{a_{p,n-i+1}}{a_{q,n-i+1}} \right\} , \quad i \leq p, q \leq n-i+1
\tag{4.2.22}
\]

then rows \( p \) and \( q \) of the matrix \( A \) and the right hand side vector \( \mathbf{b} \) are interchanged (if necessary) with rows \( i \) and \( n-i+1 \).
(or alternatively with rows n-i+1 and i) respectively and the factorisation process is continued. It can be easily seen that at each stage i, i = 1, 2, ..., \[ \left\lceil \frac{n+1}{2} \right\rceil \], of the scheme a number of \((n-2i+1)(n-2i+2)/2\) determinant evaluations are necessary to locate the required pivot.

(ii) Full Pivoting: In this method, the linear system (4.2.1) is changed to an equivalent system \(PAQy = Pb\) where \(y = Q^{-1}x\) and \(P\) and \(Q\) are two permutation matrices which hold the record of row and column interchanges. The pivot at stage i of the factorisation process is determined as follows: a search is carried out amongst the elements of the submatrix \(A_i\) which is formed from \(A\) by deleting all rows and columns which are less than i or greater than \((n-i+1)\). Then, we evaluate \(\alpha\) as:

\[
\alpha = \max \left\{ \text{abs} \left| \begin{array}{cc} a_{p,j} & a_{p,k} \\ a_{q,j} & a_{q,k} \end{array} \right| \right\}, \quad i < p, q, j, k \leq n-i+1
\]

(4.2.23)

and the necessary row and column interchanges are made before continuing the procedure. As can be seen this form of pivoting corresponds to the full pivotal strategy used in the standard Gauss-elimination process. It can be shown that \((n-2i+1)^2(n-2i+2)\) determinant evaluation are required at stage i, i = 1, 2, ..., \(\left\lceil \frac{n+1}{2} \right\rceil\) to locate the pivot.

2. Gauss-elimination Method: Alternatively in order to solve each of \((2\times2)\) linear system in (4.2.11), the Gauss-elimination method can be used. In this case the procedure is based on the following algorithms.
2.a Compute \( \rho = z_{i,i} - z_{i,n-i+1} \).

If \( \rho > 0 \) then the values of \( w_{j,i} \) and \( w_{j,n-i+1} \) are obtained by the following procedure:

2.b.1 \( m = \frac{z_{i,n-i+1}}{z_{i,i}} \)

2.c.1 \( w_{j,n-i+1} = \frac{(a_{j,n-i+1} - m \cdot a_{j,j})}{(z_{n-i+1,n-i+1} - m \cdot z_{n-i+1,i})} \)

and 2.d.1 \( w_{j,i} = \frac{(a_{j,i} - z_{n-i+1,i} \cdot w_{j,n-i+1})}{z_{i,i}} \) \hspace{1cm} (4.2.24)

whereas if \( \rho < 0 \) then we have

2.b.2 \( m = \frac{z_{i,i}}{z_{i,n-i+1}} \)

2.c.2 \( w_{j,n-i+1} = \frac{(a_{j,i} - m \cdot a_{j,n-i+1})}{(z_{n-i+1,i} - m \cdot z_{n-i+1,n-i+1})} \)

and 2.d.2 \( w_{j,i} = \frac{(a_{j,n-i+1} - z_{n-i+1,n-i+1} \cdot w_{j,n-i+1})}{z_{i,n-i+1}} \) \hspace{1cm} (4.2.25)

If we adopt the basic assumptions made in Shanehchi [1980] that a parallel replacement statement requires negligible time, while any other parallel arithmetic operation needs the same time which is referred to as a unit time step we can easily determine that for the complete factorisation process, if Cramer's rule is used for the solution of the \((2 \times 2)\) linear systems in (2.4.11) then we shall require either a total number of \( 6 \left\lfloor \frac{n-1}{2} \right\rfloor \) time steps and a maximum number of \( P \) processors where

\[
P \equiv \max \left\{ 6(n-2), 2(n-2)^2 \right\}, \hspace{1cm} (4.2.26)
\]
or a total number of \(7 \left\lceil \frac{n-1}{2} \right\rceil\) time steps and a maximum number of \(P\) processors where

\[
P = \max \left\{ 6(n-2), (n-2)^2 \right\}
\]  
(4.2.27)

and if the Gauss-elimination method is employed then we shall need either a total of \(11 \left\lceil \frac{n-1}{2} \right\rceil\) time steps and a maximum of \(2(n-2)^2\) number of processors or a total number of \(12 \left\lceil \frac{n-1}{2} \right\rceil\) time steps and a maximum number of \(P\) processors where

\[
P = \max \left\{ 2(n-2), (n-2)^2 \right\}
\]  
(4.2.28)

Moreover, it can be seen that the complete evaluation of the components of the auxiliary vector \(\mathbf{y}\) in (4.2.16) requires a total number of \(3 \left\lceil \frac{n-1}{2} \right\rceil\) time steps and a maximum number of \(2(n-2)\) processors. However, it can easily be seen that for the solution of the linear system (4.2.15), if Cramer's rule is used for solving the \((2\times2)\) linear systems in (4.2.19), then a total number of \(6 \left\lceil \frac{n-1}{2} \right\rceil + \frac{3}{2} (3+(-1)^n)\) time steps and a maximum number of \(6(n-2)\) processors are required, while if the Gauss-elimination method is employed then a total number of \(11 \left\lceil \frac{n-1}{2} \right\rceil + 7 + 4(-1)^n\) time steps and a maximum number of \(2(n-2)\) processors are needed.

Table (4.2.1) illustrates the alternative computational costs for the complete quadrant interlocking factorisation of the coefficient matrix \(A\) and the solution of the linear system (4.2.1).
Solution Method of the (2×2) linear system. | Total Number of Time Steps. | Maximum Number of Processors Working in Parallel.
---|---|---
Cramer's Rule | \[ 15 \left(\frac{n-1}{2}\right) + \frac{3}{2} \left[3+(-1)^n\right] \] | \[ \max\{6(n-2), 2(n-2)^2\} \]
| \[ 16 \left(\frac{n-1}{2}\right) + \frac{3}{2} \left[3+(-1)^n\right] \] | \[ \max\{6(n-2), (n-2)^2\} \]
Gauss-elimination Method | \[ 25 \left(\frac{n-1}{2}\right) + 7 + 4(-1)^n \] | \[ 2(n-2)^2 \]
| \[ 26 \left(\frac{n-1}{2}\right) + 7 + 4(-1)^n \] | \[ \max\{2(n-2), (n-2)^2\} \]

**TABLE (4.2.1)**

Because of the rounding errors which occur during the Q.I. Factorisation of a coefficient matrix A, it can be said that the computed quadrant factors of A, i.e. W and Z, correspond to the exact factorisation of A + E where E is a perturbation matrix.

It can be shown that the Q.I. Factorisation of a coefficient matrix A is true for computed values of the matrix W and Z such that

\[ W \ast Z = A + E \quad , \quad (4.2.29) \]

where E is given by

\[
\begin{cases} 
|E| < \rho \ast N_1 & , \quad \text{for } n \text{ odd} \\
|E| < \rho \ast N_2 & , \quad \text{for } n \text{ even} 
\end{cases} \quad (4.2.30)
\]
where

\[
P = \max \left\{ 2^{-t(3.09 + \frac{4.12g^2}{a})}, g^{2-t}e^{5.01} \right\}
\]  \hspace{1cm} (4.2.31)

and \(N_1, N_2\) are the matrices of the form:

\[
N_1 = \begin{bmatrix}
0 & 0 & 0 & \ldots & 0 & 0 & 0 \\
1 & 1 & 1 & \ldots & 1 & 1 & 1 \\
1 & 2 & 2 & \ldots & 2 & 2 & 1 \\
1 & 2 & 3 & \ldots & 3 & 2 & 1 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
1 & 2 & 3 & \ldots & 3 & 2 & 1 \\
1 & 2 & 2 & \ldots & 2 & 2 & 1 \\
1 & 1 & 1 & \ldots & 1 & 1 & 1 \\
0 & 0 & 0 & \ldots & 0 & 0 & 0
\end{bmatrix}
\]  \hspace{1cm} \text{for } n \text{ odd}  \hspace{1cm} (4.2.32a)

\[
N_2 = \begin{bmatrix}
0 & 0 & 0 & \ldots & 0 & 0 & 0 \\
1 & 1 & 1 & \ldots & 1 & 1 & 1 \\
1 & 2 & 2 & \ldots & 2 & 2 & 1 \\
1 & 2 & 3 & \ldots & 3 & 2 & 1 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
1 & 2 & 3 & \ldots & 3 & 2 & 1 \\
1 & 2 & 2 & \ldots & 2 & 2 & 1 \\
1 & 1 & 1 & \ldots & 1 & 1 & 1 \\
0 & 0 & 0 & \ldots & 0 & 0 & 0
\end{bmatrix}
\]  \hspace{1cm} \text{for } n \text{ even}  \hspace{1cm} (4.2.32b)
where

\[
\alpha = \text{Min} \left\{ \left| \text{det} \begin{bmatrix} a_{i,i} & a_{i,p} \\ a_{p,i} & a_{p,p} \end{bmatrix} \right|, \ i = 1, 2, \ldots, \frac{(n-1)}{2} \right\}
\]

(4.2.33)

and \( g \) is the largest in absolute value of the elements of the matrix \( Z \) evaluated during the factorisation process.

For a detailed discussion on this error analysis and also the error bounds for the solution of the linear system, see Shanehchi and Evans [1981] or Shanehchi [1980].
4.2.3 A Modification of the Q.I.F. Method.

In this section, we now present a modification to the Q.I.F. method, whereby the two matrix factors, \( W \) and \( Z \) are chosen to be slightly different in structure from the matrix factors involved in the original method, see Evans and Hadjidimos [1980].

Consider the system of equations (4.1.1) and assume that there exists two matrices \( W \) and \( Z \) such that

\[
A = W \times Z, \tag{4.2.34}
\]

where the matrices \( W \) and \( Z \) are defined as follows:

\[
W_i \text{ and } Z_i \quad i = 1(1)n
\]

are the column vectors of the matrices \( W \) and \( Z^T \), then we have

\[
W = \begin{bmatrix} W_1 & W_2 & \ldots & W_n \end{bmatrix}, \tag{4.2.35}
\]

\[
Z^T = \begin{bmatrix} Z_1 & Z_2 & \ldots & Z_n \end{bmatrix}, \tag{4.2.36}
\]

where the column vectors \( W_i \) and \( Z_i \) are of the following general forms:

\[
W_i \equiv \begin{cases} 
\begin{bmatrix} 0 & \ldots & 0 & 1 & w_{i+1,i} & \ldots & w_{n-i+1,i} & 0 & \ldots & 0 \end{bmatrix}^T, & i = 1(1) \left\lfloor \frac{n}{2} \right\rfloor \\
\begin{bmatrix} 0 & \ldots & 0 & 1 & 0 & \ldots & 0 \end{bmatrix}^T, & i = \left\lceil \frac{n}{2} \right\rceil + 1 \\
\begin{bmatrix} 0 & \ldots & 0 & w_{n-1+2,i} & \ldots & w_{i-1,i} & 1 & 0 & \ldots & 0 \end{bmatrix}^T, & i = \left\lceil \frac{n}{2} \right\rceil + 2(1)n
\end{cases}
\tag{4.2.37}
\]

and
\[
Z_i \equiv \begin{bmatrix} 0 & \ldots & 0 & z_{i,i} & \ldots & z_{i,n-i+2} & \ldots & z_{i,i} & 0 & \ldots & 0 \end{bmatrix}^T, \quad i = \begin{cases} 1, & \frac{n+1}{2} \end{cases} \\
\begin{bmatrix} 0 & \ldots & 0 & z_{i,n-i+1} & 0 & \ldots & 0 \end{bmatrix}^T, \quad i = \begin{cases} \frac{n+1}{2}, & + 1 \end{cases} \end{bmatrix}
\]

Also it can be easily seen, from (4.2.34), (4.2.35) and (4.2.36), that the following relationship holds:

\[
A = \sum_{i=1}^{n} W_i Z_i^T.
\]

As examples for \(n = 5\) we have that

\[
W = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
w_{2,1} & 1 & 0 & 0 & w_{2,5} \\
w_{3,1} & w_{3,2} & 1 & w_{3,4} & w_{3,5} \\
w_{4,1} & w_{4,2} & 0 & 1 & w_{4,5} \\
w_{5,1} & 0 & 0 & 0 & 1
\end{bmatrix}
\]

\[
Z = \begin{bmatrix}
z_{1,1} & z_{1,2} & z_{1,3} & z_{1,4} & z_{1,5} \\
0 & z_{2,2} & z_{2,3} & z_{2,4} & 0 \\
0 & 0 & z_{3,3} & 0 & 0 \\
0 & 0 & z_{4,3} & z_{4,4} & 0 \\
0 & 0 & z_{5,3} & z_{5,4} & z_{5,5}
\end{bmatrix}
\]

while for \(n = 6\), the matrices \(W\) and \(Z\) have the following forms:

\[
W = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
w_{2,1} & 1 & 0 & 0 & 0 & w_{2,6} \\
w_{3,1} & w_{3,2} & 1 & 0 & w_{3,5} & w_{3,6} \\
w_{4,1} & w_{4,2} & w_{4,3} & 1 & w_{4,5} & w_{4,6} \\
w_{5,1} & w_{5,2} & 0 & 0 & 1 & w_{5,6} \\
w_{6,1} & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\]

and
It can readily be seen that the elements of the matrices $W$ and $Z$ are evaluated in $\left[\frac{n+1}{2}\right]$ stages where at the commencement of the $k^{th}$ stage, we define the matrix $A_k$, $k = 1(1)\left[\frac{n-1}{2}\right]$ as follows:

$$A_1 = A,$$

$$A_k = A - \sum_{i=1}^{k-1} W_i Z_i^T - \sum_{i=n-k+2}^{n} W_i Z_i^T,$$

$$k = 2(1)\left[\frac{n-1}{2}\right].$$

(4.2.40)

It is easily seen that the first and the last $(k-1)$ rows and columns of any matrix $A_k$ defined by (4.2.40) are zero. Thus, to evaluate the elements which are different from 0 and 1 in the $k^{th}$ and $(n-k+1)^{th}$ rows and columns of $W$ and $Z$ we have the following algorithm:

(a) $z_{k,j} = a_{k,j}^{(k)}$, $j = k(1)n-k+1$

(b) $w_{j,k} = a_{j,k}^{(k)}/z_{k,k}$, $j = k+1(1)n-k+1$

(c) $z_{n-k+1,j} = a_{n-k+1,j}^{(k)} - w_{n-k+1,k} z_{k,j}$, $j = k+1(1)n-k+1$

(4.2.41)
(d) \[ w_{j,n-k+1} = \left( a_j^{(k)} w_{j,k} z_{k,n-k+1} \right) / z_{n-k+1,n-k+1} \]
\[ j = k+1(1)n-k \]

and

(e) \[ A_{k+1} = A_k - W_k Z_k^{T} Z_{n-k+1}^{T} \]
\[ z_{n-k+1,n-k+1} \]
for all \( k = 1(1) \left[ \frac{n-1}{2} \right] \).

To complete the computation of all the elements of the matrices \( W \) and \( Z \), a further step (4.2.41a) has to be carried out for \( n \) odd, to evaluate the centre element \( z_{n+1,n+1} \), while if \( n \) is even the further steps (4.2.41a) and (4.2.41c) have to be carried out in order to find the central elements \( z_{k,k}, z_{k,k+1}, z_{k+1,k} \) and \( z_{k+1,k+1} \), where \( k = \frac{n}{2} \).

For the solution of linear system (4.2.1) we again need to solve two linear systems of the form:

\[ Wx = b \] (4.2.42)

and

\[ Zx = y \] (4.2.43)

where the linear system (4.2.42) is first solved for the auxiliary vector \( y \) then the solution of the linear system (4.2.43) for the vector \( x \) which is the final solution.

For the solution of the system (4.2.42) we let

\[ Wy = b = b^{(1)} \] (4.2.44)

therefore,

\[ \sum_{i=1}^{n} y_i w_i = b^{(1)} \] (4.2.45)
Hence,

\[(a) \quad y_k = b_k^{(k)} \]

\[(b) \quad y_{n-k+1} = b_{n-k+1}^{(k)} - y_k \star w_{n-k+1} \quad (4.2.46) \]

\[(c) \quad b^{(k+1)} = b^{(k)} - y_k \star w_k - y_{n-k+1} \star w_{n-k+1} \]

for \( k = 1(1) \left[ \frac{n-1}{2} \right] \). If \( n \) is odd, step (4.2.46a) has to be executed for \( k = \left[ \frac{n+1}{2} \right] \) to find the centre element \( y_k \), while for \( n \) even, steps (4.2.46a) and (4.2.46b) have to be executed for \( k = \frac{n+1}{2} \) to find the two elements in the middle which are \( y_k \) and \( y_{k+1} \).

We now carry out the following algorithm for the solution of the linear system (4.2.43), first we let

\[ Z \bar{x} = \bar{y} = \bar{y}^{(1)} \quad (4.2.47) \]

so that

\[ \sum_{i=1}^{n} x_i z_i^* = \bar{y}^{(1)} \quad (4.2.48) \]

where \( z_i^* \mid i = 1(1)n \) are the column vectors of the matrix \( Z \), i.e.

\[ Z = \begin{bmatrix} z_1^* & z_2^* & \cdots & z_n^* \end{bmatrix} \quad (4.2.49) \]

Hence, if we set \( k = \left[ \frac{n+1}{2} \right] \) then for \( n \) odd we can find that

\[ x_{\bar{k}} = y_{\bar{k}}^{(1)} / z_{\bar{k}, \bar{k}} \quad (4.2.50) \]

and

\[ y^{(2)} = y^{(1)} - x_{\bar{k}} z_{\bar{k}}^* \quad (4.2.51) \]

while for \( n \) even, we have:
Then we proceed by using the following relationships:

(a) \[ x_{n-k+1} = \frac{y_{n-k+1}}{z_{n-k+1,n-k+1}} \]  
(b) \[ x_k = \frac{(y_k - z_k^* - x_{n-k+1}^* z_{k,n-k+1}^*)}{z_k,k} \]  
(c) \[ (4.2.52) \]

\[ y^{(2)} = y^{(1)} - x_k^* \]

for all \( k = \left[ \frac{n-1}{2} \right] \) \((-1)^1\) where stage \((4.2.54c)\) is not executed for \( k = 1 \).

Under the same assumption as before, (Section 4.2.2), it can be seen that the complete factorisation of the coefficient matrix \( A \) and the solution of the linear system \((4.2.1)\) need either a total number of \( 21 \left[ \frac{n-1}{2} \right] + \frac{9}{2} (1 + (-1)^n) \) time steps and a maximum number of \( 2(n-1)^2 \) processors or a total number of \( 22 \left[ \frac{n-1}{2} \right] + \frac{9}{2} (1 + (-1)^n) \) time steps and a maximum number of \((n-1)^2\) processors.
4.2.4 The W.D.Z. Factorisation.

In this section we study the factorisation of the matrix $A$ into three matrices $W$, $D$ and $Z$, [SHANEHCHI, 1980] such that

$$A = W \ast D \ast Z,$$

(4.2.55)

where the matrices $W$ and $Z$ have the following form:

If $W_i$ and $Z_i$ \, $i = 1(1)n$ are the column vectors of the matrices $W$ and $Z^T$, then we have

$$W = \begin{bmatrix} W_1 & W_2 & \ldots & W_n \end{bmatrix}$$

(4.2.56)

and

$$Z^T = \begin{bmatrix} Z_1 & Z_2 & \ldots & Z_n \end{bmatrix}$$

(4.2.57)

where the column vectors $W_i$'s are the following general form:

(a) For $n$ odd:

$$W_i = \begin{bmatrix} 0 & \ldots & 0 & 1 & w_{i+1,i} & \ldots & w_{n-i,i} & 0 & \ldots & 0 \end{bmatrix}^T, \quad i = 1(1)\frac{n-1}{2}$$

(b) for $n$ even:

$$W_i = \begin{bmatrix} 0 & \ldots & 0 & 1 & 0 & \ldots & 0 \end{bmatrix}^T, \quad i = \frac{n+1}{2}$$

(4.2.58)

$$W_i = \begin{bmatrix} 0 & \ldots & 0 & w_{n-i+2,i} & \ldots & w_{i-1,i} & 1 & 0 & \ldots & 0 \end{bmatrix}^T, \quad i = \frac{n+3}{2} (1)n$$

(4.2.59)

(4.2.58)
The column vectors $Z_i$'s are of the following form:

$$Z_i = \begin{cases} 
[0 \ldots 0 z_{i+1} \ldots z_{i,n-i+1} 0 \ldots 0]^T, & i = 1(1)\left\lfloor \frac{n+1}{2} \right\rfloor \\
[0 \ldots 0 z_{i,n-i+1} \ldots z_{i,i-1} 1 0 \ldots 0]^T, & i = \left\lceil \frac{n+3}{2} \right\rceil (1)n.
\end{cases}$$

(4.2.60)

As examples for $n = 5$, the matrices $W$ and $Z$ have the following forms:

$$W \equiv \begin{bmatrix} 
1 & 0 & 0 & 0 & 0 \\
w_{2,1} & 1 & 0 & 0 & w_{2,5} \\
w_{3,1} & w_{3,2} & 1 & w_{3,4} & w_{3,5} \\
w_{4,1} & 0 & 0 & 1 & w_{4,5} \\
0 & 0 & 0 & 0 & 1 
\end{bmatrix} \quad Z \equiv \begin{bmatrix} 
1 & z_{1,2} & z_{1,3} & z_{1,4} & z_{1,5} \\
0 & 1 & z_{2,3} & z_{2,4} & 0 \\
0 & 0 & 1 & 0 & 0 \\
0 & z_{4,2} & z_{4,3} & 1 & 0 \\
z_{5,1} & z_{5,2} & z_{5,3} & z_{5,4} & 1 
\end{bmatrix}$$

while for $n = 6$, we have

$$W \equiv \begin{bmatrix} 
1 & 0 & 0 & 0 & 0 & 0 \\
w_{2,1} & 1 & 0 & 0 & 0 & w_{2,6} \\
w_{3,1} & w_{3,2} & 1 & 0 & w_{3,5} & w_{3,6} \\
w_{4,1} & w_{4,2} & 0 & 1 & w_{4,5} & w_{4,6} \\
w_{5,1} & 0 & 0 & 0 & 1 & w_{5,6} \\
0 & 0 & 0 & 0 & 0 & 1 
\end{bmatrix}.$$
It can be seen that the matrices $W$ and $Z$ by the above definitions have the same structure as the $W$ and $Z$ matrices in the original Q.I.F. method, except that the diagonal elements of $Z$ are now all unity. The matrix $D$ is a diagonal matrix with its elements denoted by $d_{i,i}$.

For computation of the elements of the matrices $W$, $D$ and $Z$ we have a similar procedure as mentioned before for the Q.I.F. method in Section (4.2.2), i.e. the process is executed in $\left[\frac{n-1}{2}\right]$ steps where, in general, at the $i$th step we have the solution of (2x2) linear system:

$$
\begin{cases}
    a_{i,i} w_{j,i} + a_{n-i+1,i} w_{j,n-i+1} = a_{j,i} \\
    a_{i,n-i+1} w_{j,i} + a_{n-i+1,n-i+1} w_{j,n-i+1} = a_{j,n-i+1}
\end{cases}
$$

and the relationships,

$$
\begin{cases}
    z_{i,j} = a_{i,j}/a_{i,i} \\
    z_{n-i+1,j} = a_{n-i+1,j}/a_{n-i+1,n-i+1}
\end{cases}
$$

(4.2.61) (4.2.62)
\[
\begin{align*}
&d_{i,i} = a_{i,i} \\
&d_{n-i+1,n-i+1} = a_{n-i+1,n-i+1}
\end{align*}
\] (4.2.63)

Finally, in preparation for the next stage we calculate the quantities:

\[
\begin{align*}
&k_{i,i} = a_{i,i} - w_{k,i} a_{i,i} - w_{k,n-i+1} a_{n-i+1,n-i+1}, \quad k = i+1(l)n-i .
\end{align*}
\] (4.2.64)

For the solution of the linear system (4.2.1) we need to solve the following three simpler systems:

\[
\begin{align*}
&W y = b \\
&D u = y \\
&Z x = u
\end{align*}
\] (4.2.65) (4.2.66) (4.2.67)

The solution process of the systems (4.2.65), (4.2.66) and (4.2.67) for the auxiliary vectors \( y, u \) and the final solution vector \( x \) we have the following procedure:

1. The intermediate vector \( y \) can be obtained from the following algorithm: at step \( i \) for \( i = 1,2,\ldots, \left\lfloor \frac{n-1}{2} \right\rfloor \), we compute

\[
\begin{align*}
&(a) \quad \begin{align*}
&y_i = b_i \\
&y_{n-i+1} = b_{n-i+1}
\end{align*}
\] (4.2.68)

and

\[
\begin{align*}
&(b) \quad b_j = b_j - w_{j,i} y_i - w_{j,n-i+1} y_{n-i+1}, \quad j = i+1(l)n-i .
\end{align*}
\] (4.2.69)
We then proceed to the next step.

2. The auxiliary vector \( u \) can be found by formula

\[
\begin{align*}
u_i &= y_i / d_{i,i} , \quad i = 1(1)n ,
\end{align*}
\] (4.2.70)

3. The final solution vector \( x \) can be obtained as follows:

If \( n \) is odd, we have

\[
\begin{align*}
x_{\ell} &= u_{\ell} \quad , \quad \ell = \frac{n+1}{2} \quad (4.2.71)
\end{align*}
\]

and

\[
\begin{align*}
u_j &= u_j - x_{\ell} * z_{i,\ell} \quad , \quad j = 1(1)n \text{ and } j \neq \frac{n+1}{2} . \quad (4.2.72)
\end{align*}
\]

The rest of the elements of the vector \( x \) are found in pairs, by solving \( \left\lfloor \frac{n-1}{2} \right\rfloor \) systems of \((2\times2)\) linear system in \( \left\lfloor \frac{n-1}{2} \right\rfloor \) distinct steps. In general, at the \( i \)th step we solve the \((2\times2)\) linear system

\[
\begin{align*}
x_i + z_{i,n-i+1} x_{n-i+1} &= u_i , \quad i = \ell - 1(1)n . \quad (4.2.73)
\end{align*}
\]

\[
\begin{align*}
z_{n-i+1,i} x_i + x_{n-i+1} &= u_{n-i+1} , \quad i = \ell - 1(-1) \quad (4.2.74)
\end{align*}
\]

Then we set:

\[
\begin{align*}
u_j &= u_j - x_i z_{j,i} - x_{n-i+1} z_{j,n-i+1} \quad , \quad j = 1(1)i-1 \text{ and } n-i+2(1)n .
\end{align*}
\]

If \( n \) is even, then all the components of the vector \( x \) are found in pairs, by the solution of the linear system (4.2.73) and the formula (4.2.74) respectively for \( i = \frac{n}{2}(-1) \).

The following table illustrates the alternative computational costs for the complete factorisation of a coefficient matrix \( A \) in
the form $A = W \times D \times Z$ together with the computational cost for
the solution of the linear system $A \mathbf{x} = \mathbf{b}$.

<table>
<thead>
<tr>
<th>Solution Method of the $(2\times 2)$ Linear System.</th>
<th>Total Number of Time Steps.</th>
<th>Maximum Number of Processors Working in Parallel.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cramer's Rule</td>
<td>$17 \left[ \frac{n-1}{2} \right] + 2(2+(-1)^n)$</td>
<td>$\max{6(n-2), 2(n-2)^2}$</td>
</tr>
<tr>
<td></td>
<td>$18 \left[ \frac{n-1}{2} \right] + 2(2+(-1)^n)$</td>
<td>$\max{6(n-2), (n-2)^2}$</td>
</tr>
<tr>
<td>Gauss-elimination Method</td>
<td>$27 \left[ \frac{n-1}{2} \right] + 7 + 4(-1)^n$</td>
<td>$2(n-2)^2$</td>
</tr>
<tr>
<td></td>
<td>$28 \left[ \frac{n-1}{2} \right] + 7 + 4(-1)^n$</td>
<td>$\max{2(n-2), (n-2)^2}$</td>
</tr>
</tbody>
</table>

**TABLE 4.2.2.**
4.2.5 The Choleski Form of the Q.I.F. Method.

We now study the Q.I.F. method in which the coefficient matrix $A$ of (4.2.1) is a symmetric positive definite matrix. It will be shown that [SHANEHCHI, 1980] the symmetric positive definite matrix $A$ can be factorised as follows:

$$A = W^* W^T$$  \hspace{1cm} (4.2.75)

where the matrix $W$ has the following form: if $W_i$, $i = 1, 2, \ldots, n$ is a column vector of $W$, then we have

$$W = \begin{bmatrix} W_1 & W_2 & \cdots & W_n \end{bmatrix}$$  \hspace{1cm} (4.2.76)

and where the general form of $W_i$, $i = 1, 2, \ldots, n$ is as follows:

(a) For $n$ odd,

$$W_i = \begin{cases} \begin{bmatrix} 0 & \ldots & 0 & w_{i,i} & \ldots & w_{n-i+1,i} & 0 & \ldots & 0 \end{bmatrix}^T, & i = 1(1)\frac{n-1}{2} \\ \begin{bmatrix} 0 & \ldots & 0 & w_{i,1} & 0 & \ldots & 0 \end{bmatrix}^T, & i = \frac{n+1}{2} \\ \begin{bmatrix} 0 & \ldots & 0 & w_{n-i+2,i} & \ldots & w_{i,i} & 0 & \ldots & 0 \end{bmatrix}^T, & i = \frac{n+3}{2} \end{cases}$$  \hspace{1cm} (4.2.77)

(b) For $n$ even,

$$W_i = \begin{cases} \begin{bmatrix} 0 & \ldots & 0 & w_{i,i} & \ldots & w_{n-i+1,i} & 0 & \ldots & 0 \end{bmatrix}^T, & i = 1(1)\frac{n}{2} \\ \begin{bmatrix} 0 & \ldots & 0 & w_{i,1} & 0 & \ldots & 0 \end{bmatrix}^T, & i = \frac{n}{2} + 1 \\ \begin{bmatrix} 0 & \ldots & 0 & w_{n-i+2,i} & \ldots & w_{i,i} & 0 & \ldots & 0 \end{bmatrix}^T, & i = \frac{n}{2} + 2(1)n \end{cases}$$  \hspace{1cm} (4.2.78)
Thus, for \( n = 5 \) we have

\[
W = \begin{bmatrix}
w_{1,1} & 0 & 0 & 0 & 0 \\
w_{2,1} & w_{2,2} & 0 & 0 & w_{2,5} \\
w_{3,1} & w_{3,2} & w_{3,3} & w_{3,4} & w_{3,5} \\
w_{4,1} & w_{4,2} & 0 & w_{4,4} & w_{4,5} \\
w_{5,1} & 0 & 0 & 0 & w_{5,5}
\end{bmatrix}
\]

while for \( n = 6 \), we have

\[
W = \begin{bmatrix}
w_{1,1} & 0 & 0 & 0 & 0 & 0 \\
w_{2,1} & w_{2,2} & 0 & 0 & 0 & w_{2,6} \\
w_{3,1} & w_{3,2} & w_{3,3} & 0 & w_{3,5} & w_{3,6} \\
w_{4,1} & w_{4,2} & w_{4,3} & w_{4,4} & w_{4,5} & w_{4,6} \\
w_{5,1} & w_{5,2} & 0 & 0 & w_{5,5} & w_{5,6} \\
w_{6,1} & 0 & 0 & 0 & 0 & w_{6,6}
\end{bmatrix}
\]

The elements of the matrix \( W \) can be computed in \( k = \left\lceil \frac{n+1}{2} \right\rceil \) distinct stages and we define at the start of the \( k \)th stage, the matrix \( A_k, k = 1(1) \left\lceil \frac{n-1}{2} \right\rceil \), with elements \( a_{i,j}^{(k)} \), \( i,j = 1(1)n \) as follows:

\[
A_1 = A
\]

\[
A_k = A - \sum_{i=1}^{k-1} W_i * W_i^T - \sum_{i=n-k+2}^{n} W_i * W_i^T, \quad k = 2(1) \left\lceil \frac{n-1}{2} \right\rceil.
\]

(4.2.79)

It can be seen, from the definition (4.2.79) that the first and last \((k-1)\) rows and columns of any \( A_k \) matrix are zero.
Thus, by starting with \( A_1 = A \) we evaluate at the \( k \)th step, 
\( k = 2(1) \left\lfloor \frac{n-1}{2} \right\rfloor \), the elements which are non-zero in the \( k \)th and 
\( (n-k+1) \)th columns of \( W \) by the following algorithm:

(a) \( w_{k,k} = \sqrt{a_{k,k}} \),

(b) \( w_{i,k} = a_{i,k} / w_{k,k} \), \( i = k+1(1)n-k+1 \)

(c) \( w_{n-k+1,n-k+1} = \left( a_{n-k+1,n-k+1} - \frac{2}{w_{n-k+1,k}} \right)^{1/2} \), \( (4.2.80) \)

(d) \( w_{i,n-k+1} = \left( a_{i,n-k+1} - w_{i,k} w_{n-k+1,k} \right)/w_{n-k+1,n-k+1} \), \( i = k+1(1)n-k \)

(e) \( A_{k+1} = A_k - W_k \ast W_k^T - W_{n-k+1} \ast W_{n-k+1}^T \)

To complete the computation of all the elements of \( W \) we need one more step in which, for \( n \) odd, the \( \left\lfloor \frac{n+1}{2} \right\rfloor \)th step is carried out to evaluate the centre element \( w_{i,i} \) for \( i = \frac{n+1}{2} \) from the formula \( (4.2.80a) \) and for \( n \) even, the \( \left\lfloor \frac{n}{2} \right\rfloor \)th step is carried out to evaluate the elements \( w_{k,k} \), \( w_{k+1,k} \) and \( w_{k+1,k+1} \) from the formula \( (4.2.80a) \), \( (4.2.80b) \) and \( (4.2.80c) \).

In order to solve the linear system of (4.2.1), where the matrix \( A \) is factorised as (4.2.73) we need first to solve the linear system

\[
W \chi = b
\]

(4.2.81)

for the auxiliary vector \( \chi \) with its components denoted by 
\( y_i \) \( i = 1(1)n \), and then the final solution \( x \) is obtained by solving the linear system

\[
W^T x = \chi
\]

(4.2.82)
We can solve the linear system (4.2.81) for \( k = 1(1) \left[ \frac{n-1}{2} \right] \) by the following algorithms:

(a) \( y_k = \frac{b^{(k)}_k}{w_{k,k}} \),

(b) \( y_{n-k+1} = \frac{(b^{(k)}_{n-k+1} - y_k \* w_{n-k+1,k})}{w_{n-k+1,n-k+1}} \) \( (4.2.83) \)

(c) \( b^{(k+1)} = b^{(k)} - y_k \* W_k - y_{n-k+1} \* W_{n-k+1} \).

If \( n \) is odd, step (4.2.83a) has to be carried out for \( k = \frac{n+1}{2} \) to find the element in the centre of the auxiliary vector, \( y_k \), whilst if \( n \) is even, then steps (4.2.83a) and (4.2.83b) have to be executed for \( k = \frac{n}{2} \) to find the two centre elements \( y_{\frac{n}{2}} \) and \( y_{\frac{n}{2} + 1} \) of the vector \( y \).

For the solution of the linear system (4.2.82) we shall use the algorithm presented by Evans and Hadjidimos [1980]. See also Section (4.2.3) for the solution of the linear system \( Z \hat{x} = y \), since the structure of the matrices \( Z \) and \( W^T \) is the same.

We first let

\[
W^T \hat{x} = y = y^{(1)},
\]

so that

\[
\sum_{i=1}^{n} x_i W_{1}^* = y^{(1)} \quad (4.2.84)
\]

where \( W_{1}^* \), \( i = 1, 2, \ldots, n \) are the column vectors of the matrix \( W^T \), i.e.

\[
W^T = \begin{bmatrix}
W_1^* & W_2^* & \ldots & W_n^*
\end{bmatrix}
\quad (4.2.85)
\]
Hence, if we set \( l = \left\lceil \frac{n+1}{2} \right\rceil \) then for \( n \) odd, the centre element of the vector \( x \), i.e. \( x_l \) can be evaluated as:

\[
x_l = \frac{y_l^{(1)}}{\text{wt}_{l,l}}.
\]

(4.2.86)

and \( y^{(2)} = y^{(1)} - x_l^* W_l^* \),

while for \( n \) even we find that

\[
\begin{align*}
\begin{cases}
x_{l+1} = \frac{y_l^{(1)}}{\text{wt}_{l+1,l+1}}/\text{wt}_{l+1,l+1} \\
x_l = (y_l^{(1)} - x_{l+1} \text{wt}_{l+1,l+1})/\text{wt}_{l,l}.
\end{cases}
\end{align*}
\]

(4.2.87)

and \( y^{(2)} = y^{(1)} - x_l^* W_l^* - x_{l+1}^* W_{l+1}^* \),

where \( \text{wt}_{i,j} \), \( i,j = 1,2,\ldots,n \) denotes the elements of the matrix \( W^T \).

Then we proceed using the following algorithm:

(a) \( x_{n-k+1} = \frac{y_{n-k+1}^{(l-k+1)}}{\text{wt}_{n-k+1,n-k+1}} \),

(b) \( x_k = (y_k^{(l-k+1)} - x_{n-k+1} \text{wt}_{k,n-k+1})/\text{wt}_{k,k} \),

(4.2.88)

(c) \( y^{(l-k+2)} = y^{(l-k+1)} - x_k^* W_k^* - x_{n-k+1}^* W_{n-k+1}^* \),

for all \( k = \left\lceil \frac{n-1}{2} \right\rceil \) \((-1)^l\) where step (4.2.88 c) is not executed for \( k = 1 \).

Under the same assumptions as before (Section 4.2.2), it can be seen that the complete factorisation of the matrix \( A \) and the solution of the linear system (4.2.1) requires either a total
number of $15n + 23 \left\lfloor \frac{n-1}{2} \right\rfloor + 6 + 5(-1)^n$ time steps and a maximum
of $2(n-1)^2$ processors, or a total number of $15n + 24 \left\lfloor \frac{n-1}{2} \right\rfloor + 6 + 5(-1)^n$
time steps and a maximum number of $(n-1)^2$ processors.

It can be shown that there exists a relationship between the
Choleski form of the Q.I.F. method and the Choleski factorisation
method.

Let $A$ be a symmetric positive definite matrix then by the
Choleski factorisation method we would obtain a lower triangular
matrix $L$ (say) with its elements $l_{i,j}, i = 1(1)n$ and $j = 1(1)i$
defined by

$$l_{i,i} = (a_{i,i} - \sum_{k=1}^{i-1} l_{i,k}^2)^{1/2}$$

and

$$l_{i,j} = (a_{i,i} - \sum_{k=1}^{j-1} l_{i,k} l_{j,k})/l_{j,j}, \quad j < i$$

such that

$$A = L \cdot L^T.$$  \hfill (4.2.90)

Consider the following permutation matrix $P$ with its element
denoted by $p_{i,j}; j, i = 1, 2, \ldots, n$ where the non-zero elements of $P$
are defined as:

(a) For $n$ odd,

$$p_{i,2i-1} = 1, \quad i = 1(1) \frac{n+1}{2}$$  \hfill (4.2.91a)

$$p_{\frac{n+1}{2} + i, n-2i+1} = 1, \quad i = 1(1) \frac{n-1}{2}$$
(b) For \( n \) even,

\[
P_{i,2i-1} = 1, \quad i = 1(1) \frac{n}{2}
\]

\[
P_{\frac{n}{2} + i, n-2i+2} = 1, \quad i = 1(1) \frac{n}{2}.
\]

SHANEHCHI & EVANS [1981] proved the following theorem:

Theorem (4.2.1): In the Choleski form of the quadrant interlocking factorisation method of the matrix \( PA\text{ }P^T \) where \( A \) is a symmetric positive definite matrix and \( P \) a permutation matrix defined by (4.2.91a) and (4.2.91b) the resulting matrix \( W \) which is of the form shown in (4.2.77) or (4.2.78) is a permuted form of the matrix \( L \) where \( L \) is the lower triangular matrix defined by (4.2.89) for which the relation (4.2.90) holds, i.e.

\[
W = P \text{ } L \text{ } P^T.
\]

4.3 ITERATIVE METHODS.

4.3.1 Introduction:

As we have seen previously, all the known iterative methods such as Jacobi, Gauss-Seidel and the S.O.R. method for solving the linear system of equations \( Ax = b \) are built around a partition (or splitting) of \( A \) into the form

\[
A = D - L - U
\]

where \( D \) is the main diagonal elements of \( A \), \(-L\) and \(-U\) are strictly lower and strictly upper triangular elements of \( A \) respectively.
In this part of the thesis, we introduce a new splitting of \( A \) and develop corresponding iterative methods for solving a linear system of equations. As mentioned before, we will call these iterative schemes Quadrant Interlocking Iterative (Q.I.I.) methods. In section (4.3.2) we will introduce the Quadrant Interlocking Splitting, in section (4.3.3) we will present the basic iterative methods based on quadrant interlocking splitting, in section (4.3.4) we will prove the essential theorems for convergence of the basic iterative schemes and in section (4.3.5) we have the practical implementation of these parallel schemes.

4.3.2 The Quadrant Interlocking Splitting Technique.

Consider the linear system of equations

\[
A \mathbf{x} = \mathbf{b},
\]

where \( A \) is a non-singular matrix of order \( n \), with elements \( a_{i,j} \), \( i,j = 1(1)n \) and \( \mathbf{x} \) and \( \mathbf{b} \) are two \( n \)-dimensional vectors with \( \mathbf{x} \) (unknown) and \( \mathbf{b} \) (known) given by

\[
\mathbf{x} = \begin{bmatrix} x_1, x_2, \ldots, x_n \end{bmatrix}^T
\]

\[
\mathbf{b} = \begin{bmatrix} b_1, b_2, \ldots, b_n \end{bmatrix}^T
\]

We can partition the matrix \( A \) into the form

\[
A = X - W - Z,
\]

where the structure of the matrices \( X \), \( -W \) and \( -Z \) are defined as follows:
If \( X_i, W_i \) and \( Z_i, i = 1, 2, \ldots, n \) are the column vectors of the matrix \( X \), \(-W\) and \(-Z^T\) we shall have:

\[
X = \begin{bmatrix} X_1, X_2, \ldots, X_n \end{bmatrix},
\]

\[
-W = \begin{bmatrix} W_1, W_2, \ldots, W_n \end{bmatrix},
\]

and

\[
-Z^T = \begin{bmatrix} Z_1, Z_2, \ldots, Z_n \end{bmatrix}.
\]

The column vectors \( X_i \) have the following general form:

\[
X_i = \begin{bmatrix} 0 & \ldots & 0 & a_i, i & 0 & \ldots & 0 & a_{n-i+1}, i & 0 & \ldots & 0 \end{bmatrix}^T, \quad i = 1(1) \left\lfloor \frac{n+1}{2} \right\rfloor
\]

\[
X_i = \begin{bmatrix} 0 & \ldots & 0 & a_{n-i+1}, i & 0 & \ldots & 0 & a_i, i & 0 & \ldots & 0 \end{bmatrix}^T, \quad i = \left\lceil \frac{n+3}{2} \right\rceil (1)n
\]

where the symbol \( \lfloor \alpha \rfloor \) denotes the largest integer \( \leq \alpha \). The column vectors \( W_i \) and \( Z_i \) have the following general forms:

(a) For \( n \) odd,

\[
W_i = \begin{bmatrix} 0 & \ldots & 0 & a_{i+1}, i & \ldots & a_{n-i}, i & 0 & \ldots & 0 \end{bmatrix}^T, \quad i = 1(1) \left\lfloor \frac{n-1}{2} \right\rfloor
\]

\[
W_i = \begin{bmatrix} 0 & \ldots & 0 \end{bmatrix}^T, \quad i = \frac{n+1}{2} \quad (4.3.8a)
\]

\[
W_i = \begin{bmatrix} 0 & \ldots & 0 & a_{n-i+2}, i & \ldots & a_{i-1}, i & 0 & \ldots & 0 \end{bmatrix}^T, \quad i = \frac{n+3}{2} (1)n
\]

and
(b) For \( n \) even,

\[
\begin{bmatrix}
0 & \ldots & 0 & a_{i+1, i} & \ldots & a_{n-i, i} & 0 & \ldots & 0 \\
\vdots & & & & & & & & \\
0 & \ldots & 0 & 0 & \ldots & 0 & \ldots & & 0 \\
\end{bmatrix}^T, \quad i = 1(1) \frac{n-1}{2}
\]

\[
Z_i = \begin{bmatrix}
0 & \ldots & 0 & a_{i, n-i+2} & \ldots & a_{i, i+1} & 0 & \ldots & 0 \\
\vdots & & & & & & & & \\
0 & \ldots & 0 & 0 & \ldots & 0 & \ldots & & 0 \\
\end{bmatrix}^T, \quad i = \frac{n+3}{2}(1)n
\]

and

\[
\begin{bmatrix}
0 & \ldots & 0 & a_{i, i+1} & \ldots & a_{i, n-i} & 0 & \ldots & 0 \\
\vdots & & & & & & & & \\
0 & \ldots & 0 & 0 & \ldots & 0 & \ldots & & 0 \\
\end{bmatrix}^T, \quad i = 1(1) \frac{n-1}{2} - 1
\]

\[
W_i = \begin{bmatrix}
0 & \ldots & 0 & a_{i, n-i+2} & \ldots & a_{i, i+1} & 0 & \ldots & 0 \\
\vdots & & & & & & & & \\
0 & \ldots & 0 & 0 & \ldots & 0 & \ldots & & 0 \\
\end{bmatrix}^T, \quad i = \frac{n}{2} + 2(1)n
\]

As examples for \( n = 5 \), we have

\[
X = \begin{bmatrix}
a_{1,1} & 0 & 0 & 0 & a_{1,5} \\
0 & a_{2,2} & 0 & a_{2,4} & 0 \\
0 & 0 & a_{3,3} & 0 & 0 \\
0 & a_{4,2} & 0 & a_{4,4} & 0 \\
a_{5,1} & 0 & 0 & 0 & a_{5,5}
\end{bmatrix}
\]

\[
-W = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 \\
a_{2,1} & 0 & 0 & 0 & a_{2,5} \\
a_{3,1} & a_{3,2} & 0 & a_{3,4} & a_{3,5} \\
a_{4,1} & 0 & 0 & 0 & a_{4,5} \\
0 & 0 & 0 & 0 & 0
\end{bmatrix}
\]
while for \( n = 6 \), we shall have

\[
\begin{bmatrix}
0 & a_{1,2} & a_{1,3} & a_{1,4} & 0 \\
0 & 0 & a_{2,3} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & a_{4,3} & 0 & 0 \\
0 & a_{5,2} & a_{5,3} & a_{5,4} & 0 \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
a_{1,1} & 0 & 0 & 0 & 0 & a_{1,6} \\
0 & a_{2,2} & 0 & 0 & a_{2,5} & 0 \\
0 & 0 & a_{3,3} & a_{3,4} & 0 & 0 \\
0 & 0 & a_{4,3} & a_{4,4} & 0 & 0 \\
0 & a_{5,2} & 0 & 0 & a_{5,5} & 0 \\
a_{6,1} & 0 & 0 & 0 & 0 & a_{6,6} \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
a_{2,1} & 0 & 0 & 0 & 0 & a_{2,6} \\
\underbrace{a_{3,1} a_{3,2}} & 0 & 0 & a_{3,5} & \underbrace{a_{3,6}} \\
\underbrace{a_{4,1} a_{4,2}} & 0 & 0 & a_{4,5} & \underbrace{a_{4,6}} \\
\underbrace{a_{5,1} a_{5,2}} & 0 & 0 & 0 & \underbrace{a_{5,6}} \\
0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
0 & a_{1,2} & a_{1,3} & a_{1,4} & a_{1,5} & 0 \\
0 & 0 & a_{2,3} & a_{2,4} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & a_{5,3} & a_{5,4} & 0 & 0 \\
0 & a_{6,2} & a_{6,3} & a_{6,4} & a_{6,5} & 0 \\
\end{bmatrix}
\]
4.3.3 Basic Quadrant Interlocking Iterative (Q.I.I.) Methods.

We now define four basic (Q.I.I.) methods. We will illustrate the methods for the system:

\[
\begin{bmatrix}
  a_{1,1} & a_{1,2}, \ldots, a_{1,n} \\
  a_{2,1} & a_{2,2}, \ldots, a_{2,n} \\
  \vdots & \vdots & \ddots & \vdots \\
  a_{n,1} & a_{n,2}, \ldots, a_{n,n}
\end{bmatrix}
\begin{bmatrix}
  x_1 \\
  x_2 \\
  \vdots \\
  x_n
\end{bmatrix}
= 
\begin{bmatrix}
  b_1 \\
  b_2 \\
  \vdots \\
  b_n
\end{bmatrix}
\]  

(4.3.10)

We can obtain the elements of the vector \( x \) in \( \lfloor \frac{n+1}{2} \rfloor \) distinct steps, where in each step we solve \((2\times2)\) linear systems. In general, at the \( i \)th step we solve the following linear system:

\[
\begin{aligned}
  a_{i,i} x_i + a_{i,n-i+1} x_{n-i+1} &= c_i \\
  a_{n-i+1,i} x_i + a_{n-i+1,n-i+1} x_{n-i+1} &= c_{n-i+1}
\end{aligned}
\]

(4.3.11)

where

\[

\begin{aligned}
  c_i &= - \sum_{j=1}^{n} a_{i,j} x_j + b_i \\
  c_{n-i+1} &= - \sum_{j=1}^{n} a_{n-i+1,j} x_j + b_{n-i+1}
\end{aligned}
\]

(4.3.12)

therefore, if

\[
\Delta_i = \det \begin{bmatrix} a_{i,i} & a_{i,n-i+1} \\ a_{n-i+1,i} & a_{n-i+1,n-i+1} \end{bmatrix} \neq 0 ,
\]

(4.3.13)

we obtain the unknown \( x_i \) and \( x_{n-i+1} \), \( i = 1, 2, \ldots, \lfloor \frac{n+1}{2} \rfloor \) from the
formulae

$$x_i = (c_i \cdot a_{n-i+1,n-i+1} - c_{n-i+1,i} \cdot a_{i,n-i+1}) / \Delta_i$$  \hspace{1cm} (4.3.14)

and

$$x_{n-i+1} = (c_{n-i+1,i} \cdot a_{i,i} - c_i \cdot a_{n-i+1,i}) / \Delta_i .$$  \hspace{1cm} (4.3.15)

Note that if $n$ is odd (4.3.11) is a single equation at the step $i = \frac{1}{2}(n+1)$

We have thus replaced the system (4.3.1) by the equivalent system

$$x = B \cdot x + c$$  \hspace{1cm} (4.3.16)

where

$$B = X^{-1} (\mathbf{W} + Z)$$  \hspace{1cm} (4.3.17)

and

$$c = X^{-1} b$$  \hspace{1cm} (4.3.18)

Clearly, if the $\Delta_i \neq 0$, \hspace{1cm} $i = 1, 2, \ldots, \left\lceil \frac{n+1}{2} \right\rceil$ then $X^{-1}$ exists.

(i) With the Simultaneous Quadrant Interlocking Iterative method (S.Q.I.I.) we choose arbitrary starting values $x^{(0)}_i$, \hspace{1cm} $i = 1, 2, \ldots, n$ and compute $x^{(1)}_i$ from (4.3.14) and (4.3.15) in pairs, using $x^{(0)}_i$ in the right hand side vector (4.3.12) then determine $x^{(2)}_i$ from the values $x^{(1)}_i$ etc. Thus, in general, given $x^{(k)}_i$ we can determine $x^{(k+1)}_i$ by

$$
\begin{cases}
  a_{i,i} x_i^{(k+1)} + a_{i,n-i+1} x_{n-i+1}^{(k+1)} = c_i^{(k)} , & i = 1(1) \left\lceil \frac{n+1}{2} \right\rceil , \\
  a_{n-i+1,i} x_i^{(k+1)} + a_{n-i+1,n-i+1} x_{n-i+1}^{(k+1)} = c_{n-i+1}^{(k)} \end{cases}
$$  \hspace{1cm} (4.3.19)

where
\[
\begin{align*}
\mathbf{c}^{(k)}_i &= - \sum_{j=1}^{n} a_{i,j} \mathbf{x}^{(k)}_j + b_i \quad \text{for } j \neq i \text{ and } n-i+1, \\
\mathbf{c}^{(k)}_{n-i+1} &= - \sum_{j=1}^{n} a_{n-i+1,j} \mathbf{x}^{(k)}_j + b_{n-i+1} \\
\end{align*}
\]

(4.3.20)

(Note that for \( n \) odd, the linear system (4.3.19) reduces to a single equation, or equivalently)

\[
\mathbf{x}^{(k+1)} = \mathbf{B} \mathbf{x}^{(k)} + \mathbf{c}
\]

(4.3.21)

(ii) Related to the (S.Q.I.I.) method we have Simultaneous Over-relaxation Quadrant Interlocking Iterative method (S.O.Q.I.I.). In this method we choose a real parameter \( \omega \) and replace (4.3.19) by

\[
\begin{align*}
\begin{cases}
a_{i,i} \mathbf{x}^{(k+1)}_i + a_{i,n-i+1} \mathbf{x}^{(k+1)}_{n-i+1} &= \omega \mathbf{c}^{(k)}_i + (1-\omega) (a_{i,i} \mathbf{x}^{(k)}_i + a_{i,n-i+1} \mathbf{x}^{(k)}_{n-i+1}) \\
a_{n-i+1,i} \mathbf{x}^{(k+1)}_i + a_{n-i+1,n-i+1} \mathbf{x}^{(k+1)}_{n-i+1} &= \omega \mathbf{c}^{(k)}_{n-i+1} + (1-\omega) (a_{n-i+1,n-i+1} \mathbf{x}^{(k)}_i + a_{n-i+1,n-i+1} \mathbf{x}^{(k)}_{n-i+1})
\end{cases}
\end{align*}
\]

(4.3.22)

where \( \mathbf{c}^{(k)}_i \) and \( \mathbf{c}^{(k)}_{n-i+1} \) are defined as (4.3.20), then equivalently the (S.O.Q.I.I.) is defined as

\[
\mathbf{x}^{(k+1)} = \mathbf{B}_\omega \mathbf{x}^{(k)} + \omega \mathbf{c}
\]

(4.3.23)

where

\[
\mathbf{B}_\omega = \omega \mathbf{B} + (1-\omega) \mathbf{I}
\]

(4.3.24)

With \( \omega = 1 \), we have the (S.Q.I.I.) method.

(iii) The Successive Simultaneous Quadrant Interlocking Iterative method (S.S.Q.I.I.) is defined as follows:
This method is the same as the (S.Q.I.I.) method except that at each step we use the values of $x_i^{(k+1)}$ when available. Thus, instead of (4.3.20) we have

$$
\begin{align*}
\mathbf{c}_i^{(k)} &= \frac{i-1}{\sum_{j=1}^{n} a_{i,j} x_j^{(k+1)} - \sum_{j=n-i+2}^{n} a_{i,j} x_j^{(k+1)} - \sum_{j=n-i+1}^{n} a_{i,j} x_j^{(k)} + b_i} \\
\mathbf{c}_{n-i+1}^{(k)} &= \frac{i-1}{\sum_{j=1}^{n} a_{n-i+1,j} x_j^{(k+1)} - \sum_{j=n-i+2}^{n} a_{n-i+1,j} x_j^{(k+1)} - \sum_{j=n-i+1}^{n} a_{n-i+1,j} x_j^{(k)} + b_{n-i+1}} \\
\end{align*}
$$

so the (S.S.Q.I.I.) method can be defined as

$$
\mathbf{x}^{(k+1)} = (\mathbf{X} - \mathbf{W})^{-1} \mathbf{Z} \mathbf{x}^{(k)} + (\mathbf{X} - \mathbf{W})^{-1} \mathbf{b}
$$

or

$$
\mathbf{x}^{(k+1)} = \mathbf{L} \mathbf{x}^{(k)} + \mathbf{c}
$$

where

$$
\mathbf{L} = (\mathbf{X} - \mathbf{W})^{-1} \mathbf{Z}
$$

and

$$
\mathbf{c} = (\mathbf{X} - \mathbf{W})^{-1} \mathbf{b}
$$

(iv) We now define the Successive Simultaneous Over-relaxation Quadrant Interlocking Iterative method (S.S.O.Q.I.I.) which is related to the (S.S.Q.I.I.) method. We choose a real parameter $\omega$ and replace (4.3.25) by

$$
\begin{align*}
\mathbf{c}_i^{(k)} &= \omega \mathbf{c}_i^{(k)} + (1-\omega) \left( a_{i,i} x_i^{(k)} + a_{i,n-i+1} x_{n-i+1}^{(k)} \right) \\
\mathbf{c}_{n-i+1}^{(k)} &= \omega \mathbf{c}_{n-i+1}^{(k)} + (1-\omega) \left( a_{n-i+1,i} x_i^{(k)} + a_{n-i+1,n-i+1} x_{n-i+1}^{(k)} \right)
\end{align*}
$$
or equivalently we have

\[ x^{(k+1)} = \mathcal{L}_\omega x^{(k)} + \omega c \quad \text{,} \quad \tag{4.3.31} \]

where

\[ \mathcal{L}_\omega = (X - wW)^{-1}(\omega Z + (1-\omega)X) \quad . \tag{4.3.32} \]

If \( \omega = 1 \), then the (S.S.O.Q.I.I.) method reduced to the (S.S.Q.I.I.) method.

### 4.3.4 Convergence of the basic Quadrant Interlocking Interactive Methods

In this section, the convergency of the basic (Q.I.I.) methods which were considered in section (4.3.3) is studied. We will first give theorems on the convergency of the (S.O.Q.I.I.) and (S.S.O.Q.I.I.) methods which are applicable for any matrix and we will then study the cases where the coefficient matrix \( A \) is diagonally dominant, positive definite, or where \( A \) is an L-matrix. We assume that the crossed diagonals matrix \( X \) of the matrix \( A \) is non-singular.

**Theorem (4.3.1):** For any coefficient matrix \( A \), if the (S.Q.I.I.) method converges, then the (S.O.Q.I.I.) method will also converge for

\[ 0 < \omega < 1 \quad . \]

**Proof.**

By relation (4.3.24) i.e.

\[ B_\omega = \omega B + (1-\omega)I, \]

we have the following relationship between the eigenvalues \( \rho_i \) of
(S.O.Q.I.I.) and the eigenvalues \( \mu_i \) of (S.Q.I.I.) methods:

\[
\rho_i = \omega \mu_i + (1-\omega) \quad (4.3.33)
\]

Consider \( \mu_i = r e^{i\theta} \) where \( r < 1 \) due to the convergence of the S.Q.I.I. method. Thus we have:

\[
|\rho_i|^2 = \omega^2 r^2 + 2\omega(1-\omega)r \cos \theta + (1-\omega)^2 \quad (4.3.34)
\]

If \( 0 < \omega < 1 \), then we obtain

\[
|\rho_i|^2 < 1 \quad (4.3.35)
\]

which implies the convergence of the (S.O.Q.I.I.) method.

Following [KAHAN, 1958] we have the following general convergence theorem for the (S.S.O.Q.I.I.) method.

**Theorem (4.3.2).** For the S.S.O.Q.I. iteration matrix \( \mathbf{L}_0 \) and for all real \( \omega \) we have that

\[
\rho(\mathbf{L}_0) > |\omega-1| \quad (4.3.36)
\]

Moreover, if the (S.S.O.Q.I.I.) method converges, then we have the result

\[
0 < \omega < 2 \quad (4.3.37)
\]

**Proof.**

Let \( \tilde{w} = \mathbf{X}^{-1}w \) and \( \tilde{z} = \mathbf{X}^{-1}z \), then from (4.3.32) we have

\[
\mathbf{L}_0 = (\mathbf{I} - \tilde{w} \mathbf{I})^{-1}(\omega \mathbf{Z} + (1-\omega)\mathbf{I}) \quad (4.3.38)
\]

We know that
\[ \det \mathbf{L}_\omega = \prod_{i=1}^{n} \lambda_i, \]

where \( \lambda_i \mid i = 1(1)n \) are the eigenvalues of \( \mathbf{L}_\omega \). Since \( \hat{\mathbf{W}} \) and \( \hat{\mathbf{Z}} \) have the same structure as the matrices \( \mathbf{W} \) and \( \mathbf{Z} \) respectively and since.

\[ \det (\mathbf{I} - \omega \hat{\mathbf{W}}) = 1 \]

then we have

\[ \det \mathbf{L}_\omega = \det(\omega \hat{\mathbf{Z}} + (1-\omega)\mathbf{I}) \]

\[ = (1-\omega)^n. \]

Hence

\[ \rho(\mathbf{L}_\omega) = (|1-\omega|^n)^{1/n} = |\omega-1|. \]

Now, if the (S.S.O.Q.I.I.) method converges then we have

\[ \rho(\mathbf{L}_\omega) < 1. \]

Hence

\[ |\omega-1| < 1, \]

or

\[ 0 < \omega < 2. \]

We now seek to establish convergence theorems where the coefficient matrix \( \mathbf{A} \) is diagonally dominant.

**Theorem (4.3.3).** Let \( \mathbf{A} \) be an \((n \times n)\) diagonally dominant matrix. Then the (S.Q.I.I.) method converges.

**Proof.**

By (4.3.17) (i.e. \( \mathbf{B} = \mathbf{X}^{-1}(\mathbf{W} + \mathbf{Z}) \)) we have

\[
\begin{align*}
\mathbf{b}_{i,i} &= 0, \\
\mathbf{b}_{n-i+1,i} &= 0
\end{align*}
\]

\[ , \quad i = 1(1)n . \tag{4.3.39} \]
Then

\[
b_{i,j} = (a_{n-i+1,n-i+1} \cdot a_{i,j} - a_{i,n-i+1} \cdot a_{n-i+1,j}) / (a_{i,i} \cdot a_{n-i+1,n-i+1} - a_{i,n-i+1} \cdot a_{n-i+1,i})
\]

(4.3.40)

For proof of the theorem, it is sufficient to show that

\[
\sum_{j \neq i} |b_{i,j}| < 1 \quad \text{for all } i.
\]

(4.3.41)

From (4.3.40) we have

\[
\sum_{j \neq i} |b_{i,j}| = (\frac{1}{a_{i,j} \cdot a_{n-i+1,n-i+1} - a_{i,n-i+1} \cdot a_{n-i+1,i}})
\]

(4.3.42)

Since \(A\) has diagonal dominance, it can be seen that

\[
\sum_{j \neq i} \left( |a_{n-i+1,n-i+1} \cdot a_{i,j} - a_{i,n-i+1} \cdot a_{n-i+1,j}| \right) < |a_{n-i+1,n-i+1}|
\]

\[
\sum_{j \neq i} \left( |a_{i,j}| + |a_{i,n-i+1}| \right) \sum_{j \neq i} \left( |a_{n-i+1,j}| \right)
\]

\[
< |a_{n-i+1,n-i+1}| (|a_{i,i}| - |a_{i,n-i+1}|) + |a_{i,n-i+1}|
\]

\[
(|a_{n-i+1,n-i+1}| - |a_{n-i+1,i}|)
\]

\[
= |a_{n-i+1,n-i+1}| \cdot a_{i,j} - |a_{i,n-i+1}| \cdot a_{n-i+1,i}
\]

\[
< |a_{n-i+1,n-i+1}| \cdot a_{i,i} - |a_{i,n-i+1}| \cdot a_{n-i+1,i}
\]

\[
< |a_{n-i+1,n-i+1}| \cdot a_{i,i} - |a_{i,n-i+1}| \cdot a_{n-i+1,i}.
\]
Thus, by (4.3.42) it is clear that

\[ \sum_{j \neq i} |b_{i,j}| < 1 \quad \text{for all } i. \]

**Theorem (4.3.4).** Let \( A \) be an irreducible and diagonally dominant matrix, then the (S.S.Q.I.I.) method converges, and the (S.S.O.Q.I.I.) method converges for \( 0 < \omega \leq 1 \).

**Proof.**

Suppose that \( \rho(L_\omega) > 1 \) then for some eigenvalue \( \lambda \) of \( L_\omega \) we have

\[ |\lambda| > 1 \quad \text{(i.e. } |\lambda^{-1}| < 1) \]

and

\[ \det(L_\omega - \lambda I) = \lambda \cdot \det(I - \lambda^{-1}L_\omega) = 0. \quad (4.3.43) \]

Let \( \tilde{\omega} = x^{-1}w \) and \( \tilde{z} = x^{-1}z \) as before, then from (4.3.32) we have

\[ I - \lambda^{-1}L_\omega = I - \lambda^{-1}\left[(I - \omega \tilde{\omega})^{-1}(\omega \tilde{\omega} + (1-\omega)I)\right] \quad (4.3.44) \]

or

\[ \lambda(I - \omega \tilde{\omega})(I - \lambda^{-1}L_\omega) = (\lambda + \omega - 1)I - \lambda \omega \tilde{\omega} - \omega \tilde{z}. \quad (4.3.45) \]

If \( 0 < \omega \leq 1 \), then \( \lambda + \omega - 1 \neq 0 \) and we have

\[ \frac{\lambda}{\lambda+\omega-1} (I - \omega \tilde{\omega})(I - \lambda^{-1}L_\omega) = I - \frac{\lambda \omega}{\lambda+\omega-1} \tilde{\omega} - \frac{\omega}{\lambda+\omega-1} \tilde{z}. \quad (4.3.46) \]

Since (4.3.43) we have

\[ \det \left[I - \frac{\lambda \omega}{\lambda+\omega-1} \tilde{\omega} - \frac{\omega}{\lambda+\omega-1} \tilde{z}\right] = 0. \quad (4.3.47) \]
Let \( \lambda^{-1} = \delta e^{i\theta} \), where \( \delta \) and \( \theta \) are real and \( \delta < 1 \). Thus

\[
\left| \frac{\lambda \omega}{\lambda + \omega - 1} \right| = \frac{\omega}{\left[ 1 - 2\delta(1 - \omega) \cos \theta + \delta^2(1 - \omega)^2 \right]^{1/2}} \leq \frac{\omega}{1 - \delta(1 - \omega)} .
\] (4.3.48)

It can easily be seen that

\[
1 - \frac{\omega}{1 - \delta(1 - \omega)} = \frac{(1 - \delta)(1 - \omega)}{1 - \delta(1 - \omega)} \geq 0 .
\] (4.3.49)

Hence, we have

\[
\left| \frac{\omega}{\lambda + \omega - 1} \right| \leq \left| \frac{\omega \lambda}{\lambda + \omega - 1} \right| \leq 1 .
\] (4.3.50)

Since \( A \) is a diagonally dominant matrix, it can readily be seen that the matrix \( \lambda^{-1} A \) expressed as:

\[
\lambda^{-1} A = I - \lambda \omega - \gamma
\]

\[
= I - \delta
\]

is also a diagonally dominant matrix, so by (4.3.50), it follows that the matrix

\[
I - \frac{\lambda \omega}{\lambda + \omega - 1} \omega - \frac{\omega}{\lambda + \omega - 1} \gamma
\]

is diagonally dominant and it is also irreducible, then we have

\[
\det \left[ I - \frac{\lambda \omega}{\lambda + \omega - 1} \omega - \frac{\omega}{\lambda + \omega - 1} \gamma \right] \neq 0 ,
\] (4.3.51)

which is a contradiction. Therefore, \( \rho(\mathcal{L}_\omega) < 1 \) and the (S.S.O.Q.I.I.)
method converges.

We now develop some theorems in the case where the matrix $A$ is positive definite. To begin with, we first mention the following Lemma due to [WACHSPRESS, 1966].

Lemma (4.3.1).

If $A$ is a positive definite matrix, then the iterative method with iteration matrix $G$ is convergent if the matrix

$$M = Q + Q^T - A$$

is positive definite, where

$$Q = A(I - G)^{-1}.$$  

Moreover, we have

$$\|A^{-\frac{1}{2}} G A^{-\frac{1}{2}}\| < 1.$$  

Conversely, if (4.3.54) holds, then $M$ is positive definite.

Theorem (4.3.5). Let $A$ be a real, symmetric, non-singular matrix, and let the crossed diagonals matrix of $A$ (i.e. $X$) be positive definite. Then the (S.O.Q.I.I.) method converges if and only if $A$ and $2\omega^{-1}X - A$ are positive definite. The condition that $2\omega^{-1}X - A$ is positive definite may be replaced by the condition

$$0 < \omega < 2/(1 - \mu_{\text{min}}) < 2,$$  

where $\mu_{\text{min}} < 0$ is the smallest eigenvalue of the S.Q.I. Iteration matrix $B$.  

Proof.

Since \( X \) is positive definite then \( X^\frac{1}{2} \) exists, we define

\[
B = X^\frac{1}{2} B X^{-\frac{1}{2}} = X^{-\frac{1}{2}}(W + Z)X^{-\frac{1}{2}} \tag{4.3.56}
\]

which is symmetric since \( W + Z = X - A \) is. Hence \( B \) and therefore \( B \) have real eigenvalues. Since \( B = \omega B + (1-\omega)I \), it follows that the eigenvalues \( \rho_i \) of \( B \) are given in terms of the eigenvalues \( \mu_i \) of \( B \) by

\[
\rho_i = \omega \mu_i + 1 - \omega \tag{4.3.57}
\]

Thus, the (S.O.Q.I.I.) method will converge if and only if

\[
-1 < \omega \mu_i + 1 - \omega < 1 \tag{4.3.58}
\]

for all eigenvalues \( \mu_i \) of \( B \). Since trace \( (B) = 0 \) and since the eigenvalues of \( B \) are real, it follows that \( \mu_{\min} \leq 0 \), then the condition (4.3.58) can be written as follows:

\[
\begin{cases}
1 - \mu_i > 0, \quad \omega > 0 \\
\mu_i > 1 - 2\omega^{-1} \\
\omega > 0
\end{cases}
\tag{4.3.59}
\]

Since from (4.3.56) we have

\[
\hat{A} = X^{-\frac{1}{2}} A X^{-\frac{1}{2}} = X^{-\frac{1}{2}}(X - W - Z)X^{-\frac{1}{2}} = I - B \tag{4.3.60}
\]

by the first of the conditions (4.3.59) it can be seen that all the eigenvalues of \( \hat{A} \) are positive. Hence \( \hat{A} \) and, by theorem (2.2.2) \( A \) must be positive definite. From (4.3.60) it can be seen that the eigenvalues \( \hat{\nu}_i \) of \( \hat{A} \) satisfy
\[ \hat{v}_i = 1 - \mu_i \]  \hspace{1cm} (4.3.61)

Therefore, by the second of the conditions (4.3.59) we have

\[ \hat{v}_i < 2\omega^{-1} \]  \hspace{1cm} (4.3.62)

thus, the matrix \( 2\omega^{-1}I - \hat{A} \) is positive definite. Moreover, since

\[ X^\dagger(2\omega^{-1}I - \hat{A})X^\dagger = 2\omega^{-1}X - A \]  \hspace{1cm} (4.3.63)

it follows from theorem (2.2.2) that \( 2\omega^{-1}X - A \) is positive definite.

The sufficiency of the conditions of the theorem follows from Lemma (4.3.1). If \( A \) is positive definite, then, we have

\[
Q = A * (I - B_\omega)^{-1} \\
= A * \left[ I - \omega B - (1-\omega)I \right]^{-1} \\
= A * [\omega(I - B)]^{-1} \\
= \omega^{-1}A(X^{-1}A)^{-1} = \omega^{-1}X
\]

Since \( X \) is symmetric then \( X^T = X \), we obtain

\[ M = 2\omega^{-1}X - A, \]

which is a positive definite matrix.

Now, by (4.3.60) we have,

\[ 2\omega^{-1}I - \hat{A} = (2\omega^{-1} - 1)I + \hat{B} \]  \hspace{1cm} (4.3.64)

the matrix \( (2\omega^{-1} - 1)I + \hat{B} \) is positive definite since \( 2\omega^{-1}I - \hat{A} \) is. Then we have

\[ 2\omega^{-1} - 1 + \mu_i > 0 \]  \hspace{1cm} (4.3.65)
where the $\mu_i$ are the eigenvalues of $B$. Thus, we have

$$0 < \omega < 2/(1-\mu_{\min}) \leq 2.$$ 

Therefore (4.3.55) can be replaced by the condition that $2\omega^{-1}X - A$ is positive definite and the proof of the theorem is complete.

**Corollary (4.3.1).** Under the hypotheses of theorem (4.3.5), the (S.Q.I.I.) method converges if and only if $A$ and $2X - A$ are positive definite.

**Theorem (4.3.6).** If $A$ is a real symmetric matrix with positive definite crossed diagonals matrix $X$, then the (S.S.O.Q.I.I.) method converges if and only if $A$ is positive definite and $0 < \omega < 2$.

**Proof.**

Consider that $\lambda$ is an eigenvalue of $L_\omega$, then for some $v \neq 0$ we have $L_\omega v = \lambda v$ and hence

$$(\omega Z + (1-\omega)X)v = \lambda(X - \omega W)v.$$ (4.3.66)

Multiplying both sides on the left by $v^H$ and solving for $\lambda$ we obtain

$$\lambda = \frac{[\omega(v, Zv) + (1-\omega)(v, Xv)]}{(v, Xv) - \omega(v, Wv)}.$$ (4.3.67)

where $(.,.)$ is an inner product. Since $A$ is symmetric then by definition of the matrices $W$ and $Z$ it is evident that $Z^T = W$, thus we have

$$(v, Zv) = (Z^T v, v) = (Wv, v) = (v, W^HWv).$$ (4.3.68)
If we let
\[ y = (\psi, W \psi)/(\psi, X \psi), \]  
we have
\[ \lambda = [\overline{\omega \psi} + 1 - \omega]/[1 - \omega \psi]. \]  
(4.3.70)

Suppose that \( y = re^{i\theta} \), where \( r \) and \( \theta \) are real, then we obtain
\[ |\lambda|^2 = 1 - \left[ \omega(2 - \omega)(1 - 2r \cos \theta) \right]/\left[ (1 - \omega r \cos \theta)^2 + \omega^2 r^2 \sin^2 \theta \right]. \]  
(4.3.71)

Therefore, we will have \( |\lambda| < 1 \) for \( 0 < \omega < 2 \), if and only if
\[ 1 - 2r \cos \theta > 0. \]  
(4.3.72)

But, by (4.3.68) and (4.3.69) we have
\[ 2r \cos \theta = 2 \text{Re}(y) = y + \overline{y} = (\psi, W \psi)/(\psi, X \psi) + (\psi, Z \psi)/(\psi, X \psi) \]
or
\[ 2r \cos \theta = (\psi, R \psi)/(\psi, X \psi) \]  
(4.3.73)

where
\[ R = W + Z = X - A. \]  
(4.3.74)

Therefore, we have
\[ 2r \cos \theta = 1 - (\psi, A \psi)/(\psi, X \psi) < 1, \]  
(4.3.75)

if \( A \) is positive definite since \( (\psi, X \psi) > 0 \). Thus \( |\lambda| < 1 \) and the convergence follows.

The sufficiency of the conditions of the theorem follows from Lemma (4.3.1), if the corresponding matrix \( M \) is positive definite.
We have
\[ Q = \omega^{-1} X - W. \]

Since the matrix \( A \) is symmetric then we have
\[ X^T = X, \quad W^T = Z. \]

Thus, we obtain
\[ M = Q + Q^T - A = (2\omega^{-1} - 1)X, \]
which is positive definite for \( 0 < \omega < 2 \) since \( X \) is. This completes the proof of the theorem.

**Corollary (4.3.2).** Under the hypotheses of theorem (4.3.6), the (S.S.Q.I.I.) method converges if and only if \( A \) is positive definite.

We now consider the case where the coefficient matrix \( A \) is an \( L \)-matrix and study the convergence properties of the basic Q.I.I. methods in order to compare the spectral radii of the methods.

Our comparison is based on the Stein and Rosenberg Theory, [1948].

Before going into any further details of the process, we state the following Lemmata.

**Lemma (4.3.2).** (Oldenberger, [1940]). Let \( A \) and \( B \) be two \( n \times n \) matrices if \( A \succ B \), then
\[ \rho(A) \succ \rho(B). \]

**Lemma (4.3.3).** (Frobenius, [1908]). If matrix \( A \) is a non-negative matrix, i.e. \( A \succ 0 \), then, \( \rho(A) \) is an eigenvalue of \( A \), and there exists a non-negative eigenvector of \( A \) associated with \( \rho(A) \).
Lemma (4.3.4). Let $A$ be an L-matrix, and its crossed diagonals matrix $X$ be positive definite. Then, $A$ is an M-matrix if and only if $\rho(B) < 1$, where

$$B = X^{-1}(W + Z)$$

and

$$A = X - W - Z.$$ 

Proof.

If $\rho(B) < 1$, then $I - B$ is non-singular and it can be shown that the series $I + B + B^2 + \ldots$ converges to $(I - B)^{-1}$. Since $X$ is a positive definite L-matrix, then $X$ is, in fact, a Stieltjes matrix, it can be seen, Young [1971] that a Stieltjes matrix is an M-matrix, thus we have $X^{-1} > 0$. Since $W + Z \geq 0$, we have $B \geq 0$ and therefore $(I - B)^{-1} \geq 0$, the matrix $A = X(I - B)$ is a non-singular since $X$ and $I - B$ are non-singular. Then we have

$$A^{-1} = [X(I - B)]^{-1} = (I - B)^{-1} X^{-1} > 0,$$

it follows that $A$ is an M-matrix.

Now, we will show if $A$ is an M-matrix then we have $\rho(B) < 1$. We define the matrix $\hat{A}$ as follows

$$\hat{A} = X^{-\frac{1}{2}} A X^{-\frac{1}{2}} = X^{-\frac{1}{2}} (X - W - Z) X^{-\frac{1}{2}} = I - X^{-\frac{1}{2}} (W + Z) X^{-\frac{1}{2}}.$$

It can readily be seen that $\hat{A}$ is also an M-matrix. Consider the matrix $\hat{B}$ which is similar to $B$ by the following definition

$$\hat{B} = X^{\frac{1}{2}} B X^{-\frac{1}{2}}.$$
By direct calculation we see that

\[(I - \hat{\beta}) * (I + \hat{\beta} + \hat{\beta}^2 + \ldots + \hat{\beta}^m) = I - \hat{\beta}^{m+1} \]

From the definition of \( \hat{A} \) and \( \hat{\beta} \) it is clear that

\[\hat{A} = I - X^{-1}(X \beta)X^{-1} = I - \beta \]

hence we have

\[(\hat{A})^{-1} = (I - \hat{\beta})^{-1} = (I + \hat{\beta} + \hat{\beta}^2 + \ldots + \hat{\beta}^m) + (I - \hat{\beta})^{-1} \hat{\beta}^{m+1} \]

Since \((I - \hat{\beta})^{-1} \geq 0\) and \(\hat{\beta} \geq 0\), it follows that the elements of the matrix \(I + \hat{\beta} + \hat{\beta}^2 + \ldots + \hat{\beta}^m\) must be non-decreasing functions of \(m\) and each element is bounded by the corresponding element of \((I - \hat{\beta})^{-1}\). Thus, we have the series \(I + \hat{\beta} + \hat{\beta}^2 + \ldots + \hat{\beta}^m\) is convergent and therefore, \(\rho(\hat{\beta}) = \rho(\beta) < 1\).

Let the crossed diagonals matrix \(X\) of \(A\) be non-singular, i.e. \(X^{-1}\) exists, then, we can express the matrix \(B\) of (4.3.17) as

\[B = \hat{\nu} + \hat{\nu} = \hat{W} + \hat{Z} \tag{4.3.76}\]

and the matrix \(L_\omega\) of (4.3.32) as

\[L_\omega = (I - \hat{\nu} \hat{W})^{-1}(\hat{\nu} \hat{Z} + (1-\omega)I) \], \tag{4.3.77}

where

\[\hat{\nu} \hat{W} = X^{-1} \hat{W} \], \(\hat{\nu} \hat{Z} = X^{-1} \hat{Z} \tag{4.3.78}\]

and it is evident that the matrices \(\hat{W}\) and \(\hat{Z}\) have the same structure as the definition (4.3.8) or (4.3.9). We can now prove the following theorem.
Theorem (4.3.7). Let $A$ be an $L$-matrix with positive definite crossed diagonals matrix $X$, and $0 < \omega \leq 1$ then,

(a) $\rho(B) < 1$, if and only if $\rho(L_\omega) < 1$.

(b) $\rho(B) < 1$ (and $\rho(L_\omega) < 1$), if and only if $A$ is an $M$-matrix if $\rho(B) < 1$,

then

$$\rho(L_\omega) \leq 1 - \omega + \omega \rho(B).$$

(c) If $\rho(B) \geq 1$ and $\rho(L_\omega) \geq 1$, then

$$\rho(L_\omega) \geq 1 - \omega + \omega \rho(B) \geq 1.$$

Proof.

By the structure of the matrix $\hat{W}$ it can easily be seen that

$$\hat{W} \frac{n+1}{2} = 0,$$

and since $A$ is an $L$-matrix and $0 < \omega \leq 1$, we have

$$\left(1 - \omega \hat{W}\right)^{-1} = I + \omega \hat{W} + \omega^2 \hat{W}^2 + \ldots + \omega \frac{n+1}{2} \hat{W}^{n+1} \geq 0$$

and

$$L_\omega = \left(1 - \omega \hat{W}\right)^{-1}(\omega \hat{Z} + (1-\omega)I) \geq 0.$$ (4.3.79)

(4.3.80)

Let $\lambda = \rho(L_\omega)$ and $\mu = \rho(B)$, then by Lemma (4.3.3) $\lambda$ is an eigenvalue of $L_\omega$ and for some $\mu \neq 0$ we have

$$L_\omega \mu = \lambda \mu$$ (4.3.81)

and

$$(\lambda \hat{W} + \hat{Z})\mu = \left[(\lambda + \omega - 1)/\omega\right] \mu.$$ (4.3.82)

Thus, $(\lambda + \omega - 1)/\omega$ is an eigenvalue of the matrix $\lambda \hat{W} + \hat{Z}$. 


Then, we have
\[
\lambda + \omega - 1 \leq \omega \rho(\lambda \nabla \hat{w} + \hat{z}) .
\] (4.3.83)

If \( \lambda < 1 \), then
\[
\lambda \nabla \hat{w} + \hat{z} \leq \nabla \hat{w} + \hat{z} .
\] (4.3.84)

From Lemma (4.3.2) we have
\[
\rho(\lambda \nabla \hat{w} + \hat{z}) \leq \rho(\hat{w} + \hat{z}) = \mu ,
\] (4.3.85)

hence,
\[
\lambda \leq \omega \mu + 1 - \omega ,
\]
on the other hand, if \( \lambda > 1 \), then
\[
\lambda \nabla \hat{w} + \hat{z} \leq \lambda(\nabla \hat{w} + \hat{z}) .
\] (4.3.86)

Therefore, we obtain the result
\[
(\lambda + \omega - 1)/\omega \leq \rho(\lambda \nabla \hat{w} + \hat{z}) \leq \rho(\lambda \nabla \hat{w} + \lambda \hat{z}) = \lambda \mu ,
\] (4.3.87)

and hence
\[
\mu \geq (\lambda + \omega - 1)/(\omega \lambda) = 1 + [(1-\omega)(\lambda-1)/(\omega \lambda)] > 1 .
\] (4.3.88)

By the above discussion we have the following results:

(i) if \( \lambda < 1 \), then \( \lambda < \omega \mu + 1 - \omega \)

(ii) if \( \lambda > 1 \), then \( \mu > 1 \)

(iii) if \( \mu < 1 \), then \( \lambda < 1 \).
Since $X$ is positive definite and $A$ is an L-matrix, then $X$ is an M-matrix, i.e. $X^{-1} > 0$. Therefore, $B > 0$ and thus from Lemma (4.3.3) $\mu$ is an eigenvalue of $B$. Hence, for some $v \neq 0$ we have $Bv = \mu v$.

If we let

$$R = (I - \alpha W)^{-1} (\omega Z + (1-\omega)I)$$

and

$$\alpha = \omega / (1 - \omega + \omega \mu) ,$$

then we have

$$L v = (1 - \omega + \omega \mu)v .$$

Hence

$$1 - \omega + \omega \mu \leq \rho(R) .$$

Since $\alpha \leq \omega$, therefore it can be seen that if $\mu < 1$ we have

$$\left( I - \alpha W \right)^{-1} = I + \alpha W + \ldots + \alpha^{n+1} \left( \frac{1}{\omega} \sum_{i=0}^{n+1} \right)^{-1} .$$

Therefore, we have $R \leq L_\omega$ and hence

$$1 - \omega + \omega \mu \leq \rho(R) \leq \rho(L_\omega) = \lambda .$$

Thus, it has been shown that:

(iv) if $\mu > 1$, then $\lambda > 1 - \omega + \omega \mu > 1$,

(v) if $\lambda < 1$, then $\mu < 1$. 

By (iii) and (v) we have (a). Also by (i) and Lemma (4.3.4) we have (b). Finally, by (iv) we have (c). This completes the proof of the theorem.

Corollary (4.3.3). Let A be an L-matrix with positive definite crossed diagonals matrix X, then

(a) \( \rho(B) < 1 \), if and only if \( \rho(L) < 1 \).

(b) \( \rho(B) < 1 \) and \( \rho(L) < 1 \), if and only if A is an M-matrix, it can also be shown that in the case where \( \rho(B) < 1 \), then

\[
\rho(L) < \rho(B),
\]

(4.3.95)

(c) if \( \rho(B) \geq 1 \) and \( \rho(L) \geq 1 \), then

\[
\rho(L) \geq \rho(B).
\]

(4.3.96)

Theorem (4.3.8). If A is an M-matrix and if

\[
0 < \omega < 2 / [1 + \rho(B)]
\]

(4.3.97)

then

\[
\rho(L_\omega) < 1.
\]

Proof.

If \( 0 < \omega < 1 \), the convergence follows from theorem (4.3.7).

If \( \omega \geq 1 \), then by (4.3.79) we have

\[
T_\omega = (I - \omega W)^{-1} \left[ \omega Z + (\omega - 1) I \right] > 0
\]

(4.3.98)
If we let $\overline{\nu} = \rho(T_\omega)$, from (4.3.98) and Lemma (4.3.3) for some $v \neq 0$ we have $T_\omega v = \overline{\nu} v$ and

$$(\omega \overline{Z} + \omega \overline{\nu} \overline{W}) v = (\overline{\nu} + 1 - \omega) v.$$  \hfill (4.3.100)

Hence,

$$\overline{\nu} + 1 - \omega \leq \rho(\omega \overline{Z} + \omega \overline{\nu} \overline{W})$$  \hfill (4.3.101)

and if $\overline{\nu} \geq 1$, then by Lemma (4.3.2)

$$\overline{\nu} + 1 - \omega \leq \omega \overline{\nu} \rho(B)$$  \hfill (4.3.102)

or

$$\omega \geq (1 + \overline{\nu})/(1 + \overline{\nu} \rho(B)) \geq 2[1 + \rho(B)],$$  \hfill (4.3.103)

therefore, if (4.3.97) holds, then we must have $\overline{\nu} < 1$.

By (4.3.99) and Lemma (4.3.2), it follows that $\rho(L_\omega) \leq \overline{\nu} < 1$ and the theorem is proved.

We now develop a relation between the eigenvalues of the matrix $L_\omega$ associated with the (S.S.O.Q.I.I.) method and the eigenvalues of the matrix $B$ associated with the (S.Q.I.I.) method.

We first derive this relation in the case where the coefficient matrix $A$ is a tridiagonal matrix.

Theorem (4.3.9). Let $A$ be a tridiagonal matrix of the form
where \( k = \left\lfloor \frac{n+1}{2} \right\rfloor \). Consider the Quadrant Interlocking Splitting of the matrix \( A \) of the form \( A = X - W - Z \), then

\[
\det(X - W - Z) = \det(X - \alpha W - \alpha^{-1} Z)
\]

where \( \alpha \) is any non-zero number.

**Proof.**

Consider the diagonal matrix \( R \) of the form

\[
R = \begin{bmatrix}
1 & \alpha & \alpha^2 & \ldots & \alpha^{\frac{n-3}{2}} & 0 \\
\alpha & \frac{n-1}{2} & \alpha^2 & \ldots & \alpha^{\frac{n-3}{2}} & 0 \\
\alpha^2 & \frac{n-3}{2} & \alpha^2 & \ldots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
\alpha^{k, k-1} & \alpha^{k, k} & \alpha^{k, k+1} & \ldots & 0 & 0 \\
\alpha^{n, n-1} & \alpha^{n, n} & 0 & \ldots & \alpha^{n, n-1} & 0 \\
\end{bmatrix}, \text{ for } n \text{ odd}
\]

(4.3.108)
or

\[
\begin{bmatrix}
1 & \alpha & \alpha^2 & \ldots & \frac{n}{\alpha^2} - 2 & \frac{n}{\alpha^2} - 1 & \frac{n}{\alpha^2} - 1 & \ldots & 0 \\
\alpha & 1 & \alpha^2 & \ldots & \frac{n}{\alpha^2} - 2 & \frac{n}{\alpha^2} - 1 & \frac{n}{\alpha^2} - 1 & \ldots & 0 \\
\alpha^2 & \alpha & 1 & \ldots & \frac{n}{\alpha^2} - 2 & \frac{n}{\alpha^2} - 1 & \frac{n}{\alpha^2} - 1 & \ldots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \ldots & 1 & \alpha & \alpha & \ldots & 1 \\
\alpha^2 & \alpha & 1 & \ldots & \frac{n}{\alpha^2} - 2 & \frac{n}{\alpha^2} - 1 & \frac{n}{\alpha^2} - 1 & \ldots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \ldots & 0 & 1 & \alpha & \alpha & \ldots & 1 \\
\alpha^2 & \alpha & 1 & \ldots & 0 & 0 & 1 & \alpha & \ldots & 1 \\
\end{bmatrix}, \text{ for } n \text{ even} \tag{4.3.110b}
\]

it can easily be seen that

\[
X - \alpha W - \alpha^{-1} Z = R A R^{-1} . \tag{4.3.111}
\]

Therefore,

\[
\det(X - \alpha W - \alpha^{-1} Z) = \det(R A R^{-1})
\]

\[
= \det(R) \cdot \det(A) \cdot \det(R^{-1})
\]

\[
= \det(A).
\]

**Theorem (4.3.10).** Let the matrix \( A \) be of the form (4.3.108),

then the S.Q.I. Iteration matrix \( B \) corresponding to \( A \) is

\[
B = X^{-1}(W + Z) = \overset{\vee}{W} + \overset{\vee}{Z} , \tag{4.3.112}
\]

where \( X^{-1} W = \overset{\vee}{W} \) and \( X^{-1} Z = \overset{\vee}{Z} \) are of the form
\[
\mathbf{w} = \begin{bmatrix}
0 & 0 & \cdots & \cdots & \cdots & 0 \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
0 & 0 & \cdots & \cdots & \cdots & 0
\end{bmatrix},
\]

or

\[
\mathbf{w} = \begin{bmatrix}
0 & 0 & \cdots & \cdots & \cdots & 0 \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
0 & 0 & \cdots & \cdots & \cdots & 0
\end{bmatrix},
\]

\[
\mathbf{z} = \begin{bmatrix}
0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0
\end{bmatrix},
\]

for \( n \) even

(4.3.103a)
Then we have

(i) the non-zero eigenvalues of the matrix $B$ occur in pairs $\pm \mu_i$,

(ii) there exists a relationship between the non-zero eigenvalues $\lambda$ of the S.S.O.Q.I. iteration matrix $L_\omega$ and the non-zero eigenvalues $\mu$ of $B$, of the form

$$ (\lambda + \omega - 1)^2 = \lambda \omega^2 \mu^2. \quad (4.3.114) $$

Proof.

(1) The characteristic polynomial of $B$ is

$$ P(\mu) = \det[\mu I - B] $$

$$ = \det[\mu I - W - Z], $$

where $\mu I - B$ is a tridiagonal matrix. Therefore, by theorem (4.3.9) we have
\[
\det \left[ \mu I - \tilde{W} - \tilde{Z} \right] = \det \left[ \mu I - \alpha \tilde{W} - \alpha^{-1} \tilde{Z} \right]. 
\quad (4.3.115)
\]

By choosing \( \alpha = -1 \) in (4.3.115) we have
\[
\det \left[ \mu I - \tilde{W} - \tilde{Z} \right] = \det \left[ \mu I + \tilde{W} + \tilde{Z} \right] = (-1)^n \det \left[ -\mu I - \tilde{W} - \tilde{Z} \right]
\]
i.e.
\[
P(\mu) = (-1)^n P(-\mu) \quad . \quad (4.3.116)
\]

It follows therefore, that if \( n \) is odd, \( P(\mu) = -P(-\mu) \), so \( P(\mu) \) is a polynomial in \( \mu \) of odd degree. Similarly, \( P(\mu) \) is an even polynomial of \( \mu \) when \( n \) is even. Hence \( P(\mu) \), for some integer \( r \geq 0 \) will either have the form
\[
P(\mu) = \mu^{2r} g(\mu^2),
\]
when \( n \) is even, or the form
\[
P(\mu) = \mu^{2r+1} f(\mu^2),
\]
when \( n \) is odd where \( g(x) \) and \( f(x) \) denote polynomials in \( x \) such that \( g(0) \neq 0 \) and \( f(0) \neq 0 \) then the non-zero roots of \( P(\mu) = 0 \) are given either by \( g(\mu^2) = 0 \) or by \( f(\mu^2) = 0 \), thus proving that the non-zero eigenvalues of \( B \) occur in pairs \( \pm \mu_i \).

(2) We know that the S.S.O.Q.I. Iteration matrix is of the form
\[
\mathcal{L}_\omega = (I - \omega^{-1} X^{-1} \tilde{W})^{-1} \left\{ (1-\omega)I + \omega X^{-1} \tilde{Z} \right\} \quad (4.3.117)
\]

Let \( \tilde{W} = X^{-1} W \) and \( \tilde{Z} = X^{-1} Z \), then we have
\[
\mathcal{L}_\omega = (I - \omega \tilde{W})^{-1} \left\{ (1-\omega)I + \omega \tilde{Z} \right\} \quad . \quad (4.3.118)
\]
The eigenvalues $\lambda$ of $\mathcal{L}_\omega$ are the roots of
\[
\det(\lambda I - \mathcal{L}_\omega) = 0.
\]

By expressing $\lambda I$ as
\[
\lambda I = \lambda(I - \omega \hat{\nu})^{-1}(I - \hat{\nu}),
\]
we can define $\lambda I - \mathcal{L}_\omega$ as
\[
\lambda I - \mathcal{L}_\omega = (I - \omega \hat{\nu})^{-1}\left\{\lambda(I - \omega \hat{\nu}) - (1 - \omega)I - \omega \hat{\nu}\right\}
\]
\[= (I - \omega \hat{\nu})^{-1}\left\{(\lambda + \omega - 1)I - \lambda \omega \hat{\nu} - \omega \hat{\nu}\right\}.
\]
Hence
\[
\det(\lambda I - \mathcal{L}_\omega) = \det(I - \omega \hat{\nu})^{-1} \ast \det\left\{(\lambda + \omega - 1)I - \lambda \omega \hat{\nu} - \omega \hat{\nu}\right\}.
\]

Now, we have
\[
\begin{bmatrix}
1 & \cdots & 0 \\
-\omega \hat{\nu}_{2,1} & 1 & \cdots & 0 \\
\vdots & \ddots & \ddots & \vdots \\
-\omega \hat{\nu}_{k,k-1} & \cdots & 1 & \cdots & 0 \\
0 & \cdots & 0 & \cdots & 1
\end{bmatrix}, \text{ for } n \text{ even}
\]
\]

or
\[
\begin{bmatrix}
1 & \cdots & 0 \\
-\omega \hat{\nu}_{2,1} & 1 & \cdots & 0 \\
\vdots & \ddots & \ddots & \vdots \\
-\omega \hat{\nu}_{k,k-1} & \cdots & 1 & \cdots & 0 \\
0 & \cdots & 0 & \cdots & 1
\end{bmatrix}, \text{ for } n \text{ even}
\]
\]
Then, it is not difficult to see that
\[
\det(I - \omega^n) = 1, \quad (4.3.121)
\]
therefore, we have
\[
\det(I - \omega^n)^{-1} = 1, \quad (4.3.122)
\]
so
\[
\det(\lambda I - L_\omega) = \det \left\{ (\lambda + \omega - 1)I - \lambda \omega^n - \omega^n \right\}. \quad (4.3.123)
\]

Since \( A \) is a tridiagonal matrix then \( \lambda \omega^n \) and \( \omega^n \) are of the form (4.3.113) and thus \( (\lambda + \omega - 1)I - \lambda \omega^n - \omega^n \) is also a tridiagonal matrix from Theorem (4.3.9) by choosing \( \alpha = \frac{1}{\sqrt{\lambda}} \) we obtain
\[
\det \left\{ (\lambda + \omega - 1)I - \frac{\lambda \omega}{\sqrt{\lambda}} - \frac{\omega^n}{\sqrt{\lambda}} \right\} = 0. \quad (4.3.124)
\]

If the order of the coefficient matrix \( A \) is \( n \) then equation (4.3.124) can be re-written as
\[
\omega^n \lambda^{n/2} \det \left\{ \frac{\lambda + \omega - 1}{\sqrt{\lambda}} I - (\omega^n + \omega^n) \right\} = 0, \quad (4.3.125)
\]
which implies that \( \frac{\lambda + \omega - 1}{\sqrt{\lambda} \, \omega} \) is an eigenvalue of matrix \( B = \hat{\omega} + \hat{Z} \), therefore, if we denote the eigenvalue of matrix \( B \) by \( \mu \), then we have

\[
\frac{\lambda + \omega - 1}{\sqrt{\lambda} \, \omega} = \mu \tag{4.3.126}
\]

or

\[
(\lambda + \omega - 1)^2 = \lambda \, \omega^2 \, \mu^2 \tag{4.3.127}
\]

and the proof of the Theorem is complete.

We now calculate the value of \( \omega \) which minimises the spectral radius of the S.S.O.Q.I. Iteration matrix.

Consider \( \mu_i^2 < 1 \), i.e. a convergent S.Q.I. Iteration with real eigenvalues. Let

\[
y_1(\lambda) = \frac{(\lambda + \omega - 1)}{\omega} = \frac{1}{\omega} \lambda + 1 - \frac{1}{\omega}, \quad \omega \neq 0 \tag{4.3.128}
\]

and

\[
y_2(\lambda) = \lambda^{\frac{1}{2}} \mu \tag{4.3.129}
\]

Equation (4.3.128) is the equation of a line through the point \((1,1)\) whose slope \( \frac{1}{\omega} \) decreases as \( \omega \) increases. Equation (4.3.127) can be re-written in the form

\[
\frac{\lambda + \omega - 1}{\omega} = \pm \lambda^{\frac{1}{2}} \mu \tag{4.3.130}
\]

which gives the points of intersection of the line (4.3.128) and the parabola (4.3.129). It can be seen that the largest co-ordinate \( \lambda_i \) of the points of intersection for a fixed \( \mu = \mu_i \) decreases as \( \omega \) increases until the line is a tangent to the parabola. This occurs when
For this value of $\omega$, $\min_{\omega} \lambda_0 = \omega_0 - 1$, when $\omega > \omega_1$, the roots of equation (4.3.127) are complex and it is easily shown that $|\lambda| = \omega - 1$, which increases as $\omega$ increases. Therefore, $\omega_0$ is the value of $\omega$ that minimises the largest root (i.e., $\lambda$) of equation (4.3.127) for a fixed $\mu_i$, where $\mu_i^2 < 1$. For the maximum rate of convergence of the S.S.O.Q.I.I. method we are concerned with finding the minimum value of the largest $\lambda_i(\omega) = \lambda_i(\omega)$, say. It can be seen from equation (4.3.127) that for $\omega = 1$ we have

$$\lambda_1(1) = \max_i \mu_i^2 = \mu_1^2$$

, say.

As $\omega$ increases from 1, $\lambda_1(\omega)$ decreases until $\omega = \omega_0$ is reached, where
\[ \omega_{\text{opt}} = \omega_0 = \frac{2}{1 + \sqrt{1 - \omega_1^2}} \quad (4.3.132) \]

For this value of \( \omega \), we have

\[ \min_{\omega} \lambda_i(\omega) = \min_{\omega} \text{(Spectral radius of the S.S.O.Q.I. Iteration matrix)} \]

\[ = \omega_0 - 1 \quad (4.3.133) \]

Note that, by equation (4.3.131), \( \omega_0 \) is larger than \( \omega_i \), \( i = 2, 3, \ldots \), which minimises \( \lambda_i(\omega) \) corresponding to the root pairs \( \pm \mu_i \), where

\[ |\mu_1| > |\mu_2| > |\mu_3| > \ldots \quad (4.3.134) \]

Hence, the roots \( \lambda \) of equation (4.3.127) are complex for \( \omega = \omega_0, \mu = \mu_1 \), and each root has a modulus equal to \( (\omega_0 - 1) \); i.e. \( \omega = \omega_0 \) every eigenvalue of the S.S.O.Q.I. Iteration matrix \( C_\omega \) has the same modulus \( (\omega_0 - 1) \). In other words, each eigenvalue is represented by a point on the circle \( |\lambda| = \omega_0 - 1 \).

In the next chapter we will show that much of the above theoretical results on the tridiagonal matrices can be easily extended to hold for a wider class of matrices, namely the class of group consistently ordered (G.C.O.) matrices. We first develop a relation between the Quadrant Interlocking Iterative methods and a special case of group iterative methods.

**Definition (4.3.1).** Consider the \((n \times n)\) coefficient matrix \( A \), we define an ordered grouping of the \( n \) first integer numbers 1, 2, \ldots, \( n \) as follows:
\[ G_1 = \{1, n\}, \quad G_2 = \{2, n-1\}, \quad \ldots, \quad G_{\left\lfloor \frac{n+1}{2} \right\rfloor} = \left\{ \left\lfloor \frac{n+1}{2} \right\rfloor, \quad n - \left\lfloor \frac{n+1}{2} \right\rfloor + 1 \right\}. \] (4.3.135)

(Note that if \( n \) is odd, then \( G_{\left\lfloor \frac{n+1}{2} \right\rfloor} \) has a single element \( \left\lfloor \frac{n+1}{2} \right\rfloor \).)

We divide the matrix \( A \) into submatrices \( A_{k,l} \), \( k, l = 1, 2, \ldots, \left\lfloor \frac{n+1}{2} \right\rfloor \), where each \( A_{k,l} \) is formed from \( A \) by deleting all the rows except those corresponding to \( G_k \) and all the columns except those corresponding to \( G_l \).

For example, for \( n = 5 \) we have

\[ G_1 = \{1, 5\}, \quad G_2 = \{2, 4\}, \quad G_3 = \{3\}, \]

and submatrices

\[
\begin{align*}
A_{1,1} &= \begin{pmatrix} a_{1,1} & a_{1,5} \\ a_{5,1} & a_{5,5} \end{pmatrix}, & A_{2,1} &= \begin{pmatrix} a_{1,2} & a_{1,4} \\ a_{5,2} & a_{5,4} \end{pmatrix}, & A_{1,3} &= \begin{pmatrix} a_{1,3} \\ a_{5,3} \end{pmatrix} \\
A_{2,1} &= \begin{pmatrix} a_{2,1} & a_{2,5} \\ a_{4,1} & a_{4,5} \end{pmatrix}, & A_{2,2} &= \begin{pmatrix} a_{2,2} & a_{2,4} \\ a_{4,2} & a_{4,4} \end{pmatrix}, & A_{2,3} &= \begin{pmatrix} a_{2,3} \\ a_{4,3} \end{pmatrix} \\
A_{3,1} &= \begin{pmatrix} a_{3,1} & a_{3,5} \end{pmatrix}, & A_{3,2} &= \begin{pmatrix} a_{3,2} & a_{3,4} \end{pmatrix}, & A_{3,3} &= \begin{pmatrix} a_{3,3} \end{pmatrix}
\end{align*}
\]

We now define a permutation matrix \( P \) with elements \( p_{i,j} \), \( 1, 2, \ldots, n \), such that its non-zero elements are as follows:
Case (i) for \( n \) odd,

\[
\begin{align*}
  p_{i,2i-1} &= 1, \quad i = 1(1) \frac{n+1}{2} \\
  p_{\frac{n+1}{2} + i,n-2i+1} &= 1, \quad i = 1(1) \frac{n-1}{2}
\end{align*}
\]

(4.3.136a)

and

Case (ii) for \( n \) even,

\[
\begin{align*}
  p_{i,2i-1} &= 1, \quad i = 1(1) \frac{n}{2} \\
  p_{\frac{n}{2} + i,n-2i+2} &= 1
\end{align*}
\]

(4.3.136b)

It can easily be seen that

\[
P^T A P = \tilde{A} = \begin{bmatrix}
A_{1,1} & A_{1,2} & \cdots & A_{1,\left\lfloor \frac{n+1}{2} \right\rfloor} \\
A_{2,1} & A_{2,2} & \cdots & A_{2,\left\lfloor \frac{n+1}{2} \right\rfloor} \\
& \ddots & \ddots & \ddots \\
& & A_{\left\lfloor \frac{n+1}{2} \right\rfloor,1} & A_{\left\lfloor \frac{n+1}{2} \right\rfloor,2} & \cdots & A_{\left\lfloor \frac{n+1}{2} \right\rfloor,\left\lfloor \frac{n+1}{2} \right\rfloor}
\end{bmatrix}
\]

(4.3.137)

where each \( A_{k,l} \) is a \((2\times2)\) matrix as defined in definition (4.3.1).

We now have the following Theorem.

**Theorem (4.3.11).** Consider the \((n\times n)\) matrix \( A \) and a permutation matrix \( P \) defined by (4.3.136). Let the splitting of the matrix \( \tilde{A} = P^T A P \) be of the form,

\[
\tilde{A} = D - C
\]

(4.3.138)
where

\[
D = \begin{bmatrix}
A_{11} & & & \\
& A_{2,2} & & \\
& & & 0 \\
& & 0 & A_{n+1/2, n+1/2}
\end{bmatrix}
\]  

(4.3.139)

and

\[C = L + U,
\]

where \(L\) and \(U\) are respectively strictly lower and strictly upper block triangular matrices. Then, if we split \(A\) as \(A = X - W - Z\) so we have

\[
X = PDP^T,
\]  

(4.3.140)

\[
W = PLP^T,
\]  

(4.3.141)

\[
Z = PUP^T,
\]  

(4.3.142)

where \(X\) is defined by (4.3.4) and \(W\) and \(Z\) defined by (5.4.5) and (4.3.6).
4.3.5 Practical Implementation of Parallel Schemes.

The basic classifications made by FLYNN, [1966] define a general class of parallel computers of which the most important two are Single Instruction Stream Multiple Data Stream (SIMD) and Multiple Instruction Stream Multiple Data Stream (MIMD) computers. See also STONE, [1973].

The SIMD type of parallel computer consists of an array of processors. Each of these processors execute the same string of instructions on different sets of data. Unlike a sequential computer, each of the processors in a SIMD computer are unable to generate their own instructions and the instructions are provided by a central control unit which is usually a computer by itself. The data streams provide the data for each of the processors from private memories each of which is associated with a processor. Each processor executes the same instruction generated by the control unit on its own data simultaneously.

The MIMD computer or multiprocessor system is basically a minicomputer network. Unlike the SIMD computer, each processor of an MIMD computer generates its own instruction stream which it executes on its own data stream. In an MIMD computer system, each processor has its own control unit and so is able to generate its own and generally different instruction stream. Therefore, it is possible to execute different instructions simultaneously. The processors need not be identical because they are independent of one another, however, they are compatible with each other. Each processor has its own data stream which is obtained from two sources; a large primary shared memory and a private memory associated with each processor.
For a full description of SIMD and MIMD and other parallel computers, see DUNBAR, [1978] and FLYNN, [1966].

Generally, a parallel computer with p identical processors is potentially p times as fast as a single computer. However, this limit can only be achieved in idealized situations where the following assumptions made by KUCK, [1973] hold:

a) Any number of processors and memories may be used at any time.

b) Each processor may perform any of the basic arithmetic operations (i.e. addition, subtraction, multiplication and division) at any time, but different processors may perform different operations at the same time.

c) No time is required to communicate data between the processors and memories.

d) Instructions are always available for execution as required and are never held up by a control unit.

e) There are never accessing conflicts in the memory.

f) Each operation takes the same amount of time, which will be referred to as a unit step.

Indeed, the above assumptions are used in the computational costs of the methods which we mentioned in Section (4.2) and will be our chosen essential criteria in the computational costs of the methods which we will discuss in this section.
Consider the system of equations $A \mathbf{x} = \mathbf{b}$ and the Q.I. splitting of the $(n \times n)$ coefficient matrix $A$ in the form $A = X - W - Z$, we now define the alternative parallel schemes for solution of the systems with the S.Q.I. and S.S.O.Q.I. iteration methods.

In order to illustrate the scheme, we consider the S.Q.I. iteration (4.3.21) in the matrix form as follows with $n$ as an even number,

\[
\begin{bmatrix}
    a_{1,1} & a_{1,n} & a_{1,n-1} \\
    a_{2,1} & a_{2,2} & a_{2,n-1} \\
    \vdots & \vdots & \vdots \\
    a_{n,1} & a_{n,n-1} & a_{n,n}
\end{bmatrix}
\begin{bmatrix}
    x_1^{(k+1)} \\
    x_2^{(k+1)} \\
    \vdots \\
    x_n^{(k+1)}
\end{bmatrix}
= \begin{bmatrix}
    0 & 0 & 0 & 0 \\
    0 & 0 & 0 & 0 \\
    \vdots & \vdots & \vdots & \vdots \\
    0 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
    x_1^{(k)} \\
    x_2^{(k)} \\
    \vdots \\
    x_n^{(k)}
\end{bmatrix}
+ \begin{bmatrix}
    b_1 \\
    b_2 \\
    \vdots \\
    b_n
\end{bmatrix}
\]

(4.3.143)

where, $a = \frac{n}{2}$.

As mentioned before, the unknowns $x_i^{(k+1)}$, $i = 1(1)n$, can be found in pairs, by solving $\frac{n}{2}$ systems of $(2 \times 2)$ linear systems of equations,
\[
\begin{align*}
    a_{i,i} x_i^{(k+1)} + a_{i,n-i+1} x_{n-i+1}^{(k+1)} &= c_i \quad , i = 1(1)\frac{n}{2} \\
    a_{n-i+1,i} x_i^{(k+1)} + a_{n-i+1,n-i+1} x_{n-i+1}^{(k+1)} &= c_{n-i+1}
\end{align*}
\]

(4.3.144)

In (4.3.143) the right hand side vector \( c^{(k)} \) (say) with components \( c_i^{(k)} \), \( i = 1,2,\ldots,n \) can be found by the following formulae,

\[
c_i^{(k)} = \sum_{j=1}^{n} a_{\ell,j} x_j^{(k)} + b_i \quad , \quad \ell = 1(1)n
\]

\( \ell \neq i \notin n-i+1 \) \hspace{1cm} (4.3.145)

The implementation of the above method on an SIMD type of computer where many processors are available can be formulated as follows:

(i) The matrix vector multiplication of the right hand side (4.3.143) can on most SIMD computers be performed in "1 + \log_2(n-2)" time steps, this is illustrated in the following manner.

The first step consists of the evaluation of each \( a_{i,j} \times x_j \) for \( i,j = 1,2,\ldots,n \), \( j \neq i \notin n-i+1 \). All such multiplications can be performed simultaneously in one time step (i.e. one multiplication) if \( n(n-2) \) number of processors are available. This would yield a matrix with elements \( r_{i,j} = a_{i,j} \times x_j \), \( i,j = 1,2,\ldots,n \) and \( j \neq i \notin n-i+1 \), the final step of the product vector is evaluated by summing up of all the elements of each row which can be done in \( \log_2(n-2) \) time steps and therefore requires \( (n-2)/2 \) processors.
(ii) The summation of the above vector evaluated in (i) to the vector \( b \) which is carried out in one step with \( n \) processors (i.e. one addition).

(iii) Once the \( c_i^{(k)} \), \( i = 1,2,\ldots,n \), are evaluated then the solution in the \((k+1)\)th step of iteration, \( k = 0,1,2,\ldots \), is determined by solving the \((2\times2)\) linear systems in (4.3.144). This can be achieved as follows:

a) Evaluate

\[
\Delta_i = a_{i,i} \cdot a_{p,p} - a_{i,p} \cdot a_{p,i}, \quad i = 1,2,\ldots,\frac{n}{2}, \tag{4.3.146}
\]

where \( p = n-i+1 \).

This would require 2 time steps (one multiplication and one subtraction) and 2 processors for each \( \Delta_i \) which results in requiring \( n \) processors. To evaluate \( \Delta_i \) for all \( i = 1,2,\ldots,\frac{n}{2} \).

b) We then evaluate

\[
\chi_i^{(k+1)} = (c_i^{(k)} \cdot a_{p,p} - c_{p}^{(k)} \cdot a_{i,p})/\Delta_i, \quad i = 1,2,\ldots,n \tag{4.3.147}
\]

where \( \Delta_i = \Delta_{n-i+1}^i, \quad i = 1,2,\ldots,\frac{n}{2} \),

which again requires 3 time steps (one multiplication, one subtraction and one division) and 2 processors per \( \chi_i^{(k+1)} \) and hence \( 2n \) processors to evaluate all \( \chi_i^{(k+1)} \) values.

Therefore, the complete solution of the method in each iteration requires \( 7 + \log_2(n-2) \) time steps and a maximum number of \( p \) processors where

\[
P = \max\{2n, n(n-2)\} \tag{4.3.148}
\]
We now consider the S.S.O.Q.I. Iteration method for solving the linear system $Ax = b$. By equation (4.3.31) we have

$$(X - \omega W)x^{(k+1)} = ((1-\omega)X + \omega Z)x^{(k)} + \omega b \quad (4.3.149)$$

or in the matrix form,

$$(4.3.150)$$

where $n$ is again an even number and $\alpha = \frac{n}{2}$.

With the value of $x^{(k)}$ the right hand side of (4.3.150) can easily be computed, i.e.
\[ c_i^{(k)} = \sum_{j=i+1}^{n-i} -w_{i,j} x_j^{(k)} + (1-\omega) (a_{i,i} x_i^{(k)} + a_{i,p} x_p^{(k)}) + \omega b_i, \quad i = 1(1)\frac{n}{2} \]

\[ c_p^{(k)} = \sum_{j=i+1}^{n-i} -w_{p,j} x_j^{(k)} + (1-\omega) (a_{p,i} x_i^{(k)} + a_{p,p} x_p^{(k)}) + \omega b_p, \]

where \( p = n-i+1 \).

Then, the components of unknown vector \( x^{(k+1)} \) can be found in pairs, by solving \( \frac{n}{2} \) linear systems of (2x2) equations in \( \frac{n}{2} \) distinct steps.

The solution process of the S.S.O.Q.I.I. method for each iteration can be formulated as follows:

The first step is to evaluate the right hand side vector \( c^{(k)} \).

It can be achieved as follows:

a) The evaluation of each \( -w a_{i,j} \) for \( i = 1(1)\frac{n}{2} -1, \) \( j = i+1(1)n-i \) and \( i = \frac{n}{2} +2(1)n, \) \( j = n-i+2(1)i-1. \) All these multiplications can be performed in one time step (i.e. one multiplication) and requires \( n(\frac{n}{2}-1) \) processors.

b) \( (1-\omega) a_{i,i} \) and \( (1-\omega) a_{i,n-i+1} \) for \( i = 1(1)n. \) These can be evaluated in one time step (i.e. one multiplication) if \( 2n \) number of processors are available.

c) This step consists of the evaluation of each \( a_{i,j} x_j \) for \( i(1)\frac{n}{2} , \) \( j = i(1)n-i+1 \) and \( i = \frac{n}{2} +1(1)n, \) \( j = n-i+1(1)i, \) where \( a_{i,j} \) is the element of the right hand side matrix after the evaluations in steps a) and b). All such multiplications can be performed simultaneously in one time step on \( \frac{n}{2} (n+2) \) number of processors.
d) Summing up all the elements of each row can again be done in $\log_2 n$ time steps and requires $\frac{n}{2}$ number of processors.

e) The evaluation of $w^*b$ which requires $n$ number of processors and needs just one time step (i.e. one multiplication) to complete.

f) The summation of the vectors evaluated in steps d) and e) can be carried out in one time step (i.e. one addition) with $n$ processors.

With the right hand side vector $c^{(k)}$ then the solution process in the $(k+1)^{th}$ iteration is carried out as follows:

i) Evaluate

$$\Delta_i = a_{i,i}^* a_{p,p} - a_{i,p}^* a_{p,i}, \quad i = 1(1)\frac{n}{2}$$

where $p = n-i+1$. As we have seen previously, this can be performed in 2 time steps and requires $n$ processors.

ii) The solution of the $(2\times2)$ linear systems,

$$
\begin{align*}
a_{1,1} x_1^{(k+1)} + a_{1,n} x_n^{(k+1)} &= c_1^{(k)} \\
a_{n,1} x_1^{(k+1)} + a_{n,n} x_n^{(k+1)} &= c_n^{(k)}
\end{align*}
$$

(4.3.152)

by using the formulae,

$$
\begin{align*}
x_1^{(k+1)} &= (c_1^{(k)} a_{n,n} - c_n^{(k)} a_{1,n}) / \Delta_1 \\
x_n^{(k+1)} &= (c_n^{(k)} a_{1,1} - c_1^{(k)} a_{n,1}) / \Delta_1
\end{align*}
$$

(4.3.153)

(Cramer's Rule)
It can be seen that the solution of this system requires just 3 time steps (one multiplication, one subtraction and one division) with a total number of 4 processors. Then before proceeding to evaluate the next pair we modify the right hand side vector \( \mathbf{c}^{(k)} \) as

\[
\begin{align*}
\mathbf{c}_j^{(k)} &= \mathbf{c}_j^{(k)} - \omega (a_{j,1} x_{1}^{(k+1)} + a_{j,n} x_{n}^{(k+1)}), \quad j = 2(1)\frac{n}{2} \\
\mathbf{c}_{n-j+1}^{(k)} &= \mathbf{c}_{n-j+1}^{(k)} - \omega (a_{n-j+1,1} x_{1}^{(k+1)} + a_{n-j+1,n} x_{n}^{(k+1)})
\end{align*}
\]

This requires 4 time steps (2 multiplication, 1 addition and 1 subtraction) and a total number of 2(n-2) processors to modify all the elements of the vector \( \mathbf{c}^{(k)} \). This completes this step of the algorithm.

In general, at the beginning of the \( \ell \)th stage, \( \ell = 1, 2, \ldots, \frac{n}{2} \), we have the solution of (2x2) systems,

\[
\begin{align*}
\mathbf{a}_{\ell,\ell} x_{\ell}^{(k+1)} + \mathbf{a}_{\ell,p} x_{p}^{(k+1)} &= \mathbf{c}_{\ell}^{(k)} \\
\mathbf{a}_{p,\ell} x_{\ell}^{(k+1)} + \mathbf{a}_{p,p} x_{p}^{(k+1)} &= \mathbf{c}_{p}^{(k)}
\end{align*}
\]

where \( p = n-\ell+1 \), to give the unknowns \( x_{\ell}^{(k+1)} \) and \( x_{p}^{(k+1)} \) and the modification of the right hand side vector according to the following updating formulae,

\[
\begin{align*}
\mathbf{c}_j^{(k)} &= \mathbf{c}_j^{(k)} - \omega (a_{j,\ell} x_{\ell}^{(k+1)} + a_{j,p} x_{p}^{(k+1)}) \\
\mathbf{c}_{n-j+1}^{(k)} &= \mathbf{c}_{n-j+1}^{(k)} - \omega (a_{n-j+1,\ell} x_{\ell}^{(k+1)} + a_{n-j+1,p} x_{p}^{(k+1)})
\end{align*}
\]

N.B. For \( \ell = \frac{n}{2} \) the formula (4.3.156) is not evaluated.
Therefore, the complete solution of the method in the \((k+1)\)th step of the iteration requires \(14 + \log_2 n\) time steps and a total of \(\frac{n}{2}(n+2)\) number of processors.

The above algorithm was also implemented on the Neptune system of the Department of Computer Studies of Loughborough University which is an MIMD type system with 4 Texas 990 mini-computers able to work simultaneously on different tasks.

The way in which the algorithm was designed for use on the Neptune system is briefly described below.

The shared memory of the system was arranged to hold the coefficient matrix \(A\), the right hand side vector \(b\) and the solution at each iteration so that each processor can access any part of the above data. The solution process at each iteration was carried out in \(\left[\frac{n+1}{2}\right]\) distinct stages where at the stage \(i\) the components \(x_{i}^{(k+1)}\) and \(x_{n-i+1}^{(k+1)}\), \(i = 1, 2, \ldots, \left[\frac{n+1}{2}\right]\) and \(k = 0, 1, 2, \ldots\), were evaluated. In this way, the algorithm uses only 2 processors at any one time, where all the computation related to the determination of \(x_{1}^{(k+1)}\) and \(x_{n-i+1}^{(k+1)}\) are carried out on the 2 processors independently and simultaneously. As soon as a component is evaluated it is tested for convergency and the process will continue to the next iteration if the accuracy test is not satisfied for all the components.
CHAPTER 5

GROUP AND BLOCK ITERATIVE METHODS
5.1 INTRODUCTION.

The algorithms described in Chapter 3 belong to the class of algorithms known as point iterative methods, that is, at any one time only a single equation of the system is treated. As mentioned in Chapter 4, the Quadrant Interlocking Iterative schemes can be shown to be equivalent to methods belonging to a distinct class of iterative processes known as group iterative methods.

The principle inherent in the group iterative method is to group a certain number of individual equations (mesh points) and treat this group similar to the way a single point is treated in the point iterative method. The solution of the points within each such group can be carried out either directly or iteratively.

In this chapter we study group iterative methods where the solution of each group of \( m \) (say) points is obtained using direct methods.

Our prime concern here is to construct new grouping of the mesh points into small size groups of 4 and 9 points and to investigate their advantages.

In Section 5.2 we shall present a brief review of the theory of the group iterative methods and their relevant properties. This follows with a section on the introduction of group iterative algorithms whereby the size of each group is taken to be 4 or 9. Some impressive results indicating the efficiency of such grouping strategies over some of the well known group iterative methods are given.
5.2 **GROUP ITERATIVE METHODS.**

Consider a system of linear equations,

\[ \sum_{j=1}^{n} a_{i,j} x_j = b_i , \quad i = 1,2,\ldots,n \quad (5.2.1) \]

which can be written in matrix form as

\[ A \bar{x} = \bar{b} \quad (5.2.2) \]

In the group iterative methods the equations in (5.2.1) are first assigned into different groups or subsets numbered \(1, 2, \ldots, N\), such that each of the above equations belongs to one and only one group. Then, the corresponding unknowns \(x_i\) of each group are solved in which the other unknowns belonging to the remaining groups are treated as known quantities.

A particular case of grouping the equations is "partitioning" where in this case the equations for \(i = 1, 2, \ldots, n_1\) form the first group, those for \(i = n_1+1, n_1+2, \ldots, n_2\) belong to the second group and, in general, the equations for \(i = n_r+1, n_r+2, \ldots, n_{r+1}\) constitute the \((r+1)st\) group. Note that in the above grouping \(n_N = n\) where \(N\) represents the number of distinct groups or subsets.

The iterative procedure based on "partitioning" of the equations as described above are usually known as "Block Iterative" methods.

In the case of the solution of the Dirichlet problem by finite difference approximation if we use the natural labeling of the mesh points and take all the points on each row as a group of equations, then such a grouping will correspond to a partitioning. However, if any other numbering of the mesh points apart from the natural ordering is used by assigning the points on each row
to a group then such groupings will not in general correspond to a partitioning. However, it can be shown that there exists a permutation for any arbitrary grouping which transforms the grouping to a partitioning.

In practice, we normally select a direct method to solve the equations belonging to the individual subsets since the size of these subsystems are much smaller than the original problem. Moreover, it usually happens that the matrices constituting each subsystem possess some properties, i.e. tridiagonal, quindiagonal, which help in solving the subsystem more efficiently and more economically.

We shall now describe the principle of constructing the group iterative methods. Consider the set of linear equations of order n expressed as in (5.2.2). In order to construct a group iterative method to solve the above system we first divide the integers 1, 2, \ldots, n into N ≤ n distinct sets such that each integer belongs to one and only one set. Note that it is not necessary for the groups to consist of consecutive integers although the groups are ordered by the following definition.

**Definition (5.2.1).** [YOUNG, 1971]. An ordered grouping π of T, the set of the first n positive integers, is a subdivision of T into disjoint subsets G_1, G_2, \ldots, G_N such that G_1 \cup G_2 \cup \ldots \cup G_N = T.

Two ordered groupings π and π' defined by G_1, G_2, \ldots, G_N and G'_1, G'_2, \ldots, G'_N, respectively, are identical if N = N' and if G_1 = G'_1, G_2 = G'_2, \ldots, G_N = G'_N. As examples for n = 5, we have the following ordered groupings:
\[ \pi_o : G_1 = \{1\}, \quad G_2 = \{2\}, \quad G_3 = \{3\}, \quad G_4 = \{4\}, \quad G_5 = \{5\} \]

\[ \pi_1 : G_1 = \{1,2\}, \quad G_2 = \{3,4\}, \quad G_3 = \{5\} \]

\[ \pi_2 : G_1 = \{1,5\}, \quad G_2 = \{2,4\}, \quad G_3 = \{3\} \]

\[ \pi_3 : G_1 = \{1,3,5\}, \quad G_2 = \{2,4\} \]

\[ \pi_4 : G_1 = \{3\}, \quad G_2 = \{1,2,4,5\} \]

\[ \pi_5 : G_1 = \{4\}, \quad G_2 = \{1,2,5\}, \quad G_3 = \{3\}, \quad \text{etc.} \]

It is evident that \( \pi_o \) and \( \pi_1 \) constitute partitionings.

We now apply the above definition to the linear system (5.2.2) and define an ordered grouping \( \pi \) of the equations in that system. We define the submatrices \( A_{k,l} \), for \( k,l = 1,2,\ldots,N \), such that each \( A_{k,l} \) is formed from the matrix \( A \) by deleting all rows except those corresponding to \( G_k \) and all columns except those corresponding to \( G_l \). We also define the vectors \( X_k \) and \( B_k \), for \( k = 1,2,\ldots,N \), such that each \( X_k \) and \( B_k \) are formed from vector \( x \) and \( b \) respectively by deleting all elements except those corresponding to group \( G_k \). We can now rewrite the system (5.2.2) in the equivalent form of

\[ \sum_{l=1}^{N} A_{k,l} X_l = B_k, \quad k = 1,2,\ldots,N. \quad (5.2.3) \]

For example, in the case \( n = 5 \) where the ordered grouping is defined by \( G_1 = \{1,3,5\}, \quad G_2 = \{2,4\} \), we have
\[ A_{1,1} = \begin{bmatrix} a_{1,1} & a_{1,3} & a_{1,5} \\ a_{3,1} & a_{3,3} & a_{3,5} \\ a_{5,1} & a_{5,3} & a_{5,5} \end{bmatrix}, \quad A_{1,2} = \begin{bmatrix} a_{1,2} & a_{1,4} \\ a_{3,2} & a_{3,4} \\ a_{5,2} & a_{5,4} \end{bmatrix}, \]

\[ X_1 = \begin{bmatrix} x_1 \\ x_3 \\ x_5 \end{bmatrix}, \quad B_1 = \begin{bmatrix} b_1 \\ b_3 \\ b_5 \end{bmatrix}, \]

\[ A_{2,1} = \begin{bmatrix} a_{2,1} & a_{2,3} & a_{2,5} \\ a_{4,1} & a_{4,3} & a_{4,5} \end{bmatrix}, \quad A_{2,2} = \begin{bmatrix} a_{2,2} & a_{2,4} \\ a_{4,2} & a_{4,4} \end{bmatrix}, \]

\[ X_2 = \begin{bmatrix} x_2 \\ x_4 \end{bmatrix}, \quad B_2 = \begin{bmatrix} b_2 \\ b_4 \end{bmatrix}. \]

Therefore, (5.2.3) becomes

\[
\begin{align*}
\begin{bmatrix} a_{1,1} & a_{1,3} & a_{1,5} \\ a_{3,1} & a_{3,3} & a_{3,5} \\ a_{5,1} & a_{5,3} & a_{5,5} \end{bmatrix} \begin{bmatrix} x_1 \\ x_3 \\ x_5 \end{bmatrix} &+ \begin{bmatrix} a_{1,2} & a_{1,4} \\ a_{3,2} & a_{3,4} \\ a_{5,2} & a_{5,4} \end{bmatrix} \begin{bmatrix} x_2 \\ x_4 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_3 \\ b_5 \end{bmatrix} \\
\begin{bmatrix} a_{2,1} & a_{2,3} & a_{2,5} \\ a_{4,1} & a_{4,3} & a_{4,5} \end{bmatrix} \begin{bmatrix} x_1 \\ x_3 \\ x_5 \end{bmatrix} &+ \begin{bmatrix} a_{2,2} & a_{2,4} \\ a_{4,2} & a_{4,4} \end{bmatrix} \begin{bmatrix} x_2 \\ x_4 \end{bmatrix} = \begin{bmatrix} b_2 \\ b_4 \end{bmatrix}.
\end{align*}
\]
In order to define the basic group iterative methods, we shall assume from now on that all submatrices $A_{k,k}$ are non-singular.

The group Jacobi iterative method is defined by

$$A_{k,k} x_k^{(n+1)} + \sum_{\ell=1, \ell \neq k}^{N} A_{k,\ell} x_{\ell}^{(n)} = B_k, \quad k = 1, 2, \ldots, N \quad (5.2.4)$$

or

$$x_k^{(n+1)} = \sum_{\ell=1, \ell \neq k}^{N} B_{k,\ell} x_{\ell}^{(n)} + C_k, \quad k = 1, 2, \ldots, N \quad (5.2.5)$$

where

$$B_{k,\ell} = \begin{cases} -A_{k,k}^{-1} A_{k,\ell} & \text{if } k \neq \ell \\ 0 & \text{if } k = \ell \end{cases} \quad (5.2.6)$$

and

$$C_k = A_{k,k}^{-1} B_k$$

It can be easily seen that (5.2.5) can be written in the matrix form as

$$x^{(n+1)} = B^{(\pi)} x^{(n)} + C^{(\pi)}, \quad (5.2.8)$$

where

$$B^{(\pi)} = (D^{(\pi)})^{-1} C^{(\pi)} \quad (5.2.9)$$

$$C^{(\pi)} = (D^{(\pi)})^{-1} b \quad (5.2.10)$$

$$C^{(\pi)} = D^{(\pi)} - A \quad (5.2.11)$$

$$D^{(\pi)} = \text{diag}(\pi) A \quad (5.2.12)$$
It is evident that the diag\((\pi)\) \(A\) is a matrix formed from \(A\) by replacing all \(a_{i,j}\) by zeros unless \(i\) and \(j\) belong to the same group.

The group JOR iterative method is defined by
\[
\mathbf{x}^{(n+1)} = B^{(\pi)}_{\omega}(\mathbf{x}^{(n)}) + \omega\mathbf{c}^{(\pi)}
\]  
(5.2.13)

where
\[
B^{(\pi)}_{\omega} = \omega B^{(\pi)} + (1-\omega)\mathbf{I}.
\]  
(5.2.14)

The group Gauss-Seidel iterative method is defined by
\[
A_{k,k} \mathbf{x}^{(n+1)} + \sum_{\ell=1}^{k-1} A_{k,\ell} \mathbf{x}^{(n+1)} + \sum_{\ell=k+1}^{N} A_{k,\ell} \mathbf{x}^{(n)} = B_{k,k},
\]  
\(k = 1, 2, \ldots, N\)  
(5.2.15)

or
\[
\mathbf{x}^{(n+1)} = \sum_{\ell=1}^{k-1} B_{k,\ell} \mathbf{x}^{(n+1)} + \sum_{\ell=k+1}^{N} B_{k,\ell} \mathbf{x}^{(n)} + \mathbf{c}_k,
\]  
\(k = 1, 2, \ldots, N\)  
(5.2.16)

Here, we may write (5.2.16) in the matrix form as
\[
\mathbf{x}^{(n+1)} = \mathcal{L}^{(\pi)} \mathbf{x}^{(n)} + (I - L^{(\pi)}(\pi))^{-1} \mathbf{c}^{(\pi)}
\]  
(5.2.17)

where
\[
\mathcal{L}^{(\pi)} = (I - L^{(\pi)}(\pi))^{-1} \mathbf{u}^{(\pi)}
\]  
(5.2.18)

and
\[
L^{(\pi)} = (D^{(\pi)})^{-1} C^{(\pi)}_L, \quad \mathbf{u}^{(\pi)} = (D^{(\pi)})^{-1} C^{(\pi)}_U
\]  
(5.2.19)

with \(C^{(\pi)}_L\) and \(C^{(\pi)}_U\) being the matrices formed from \(A\) by replacing
all elements of A by zero except those $a_{i,j}$ for which $i$ and $j$
belong to different groups and such that the group containing $i$
comes after and before, respectively, the group containing $j$.

The group S.O.R. iterative method is defined by

$$x^{(n+1)} = L_{\omega}(\pi)x^{(n)} + (I - \omega L(\pi))^{-1} \omega_c(\pi), \quad (5.2.20)$$

where

$$L_{\omega}(\pi) = (I - \omega L(\pi))^{-1} (\omega U(\pi) + (1 - \omega)I). \quad (5.2.21)$$

We shall investigate, in the next section, the characteristics
of special group iterative methods known as the Block iterative
method in more detail. However, we conclude this section by
representing the fact that every arbitrary group iterative method
can, by applying some interchanges, be expressed as an equivalent
Block iterative method. For this we have the following definitions.

**Definition (5.2.2).** Any permutation matrix corresponds to a
permutation function, $\sigma(i)$, defined for $i = 1, 2, \ldots, n$ by

$$\sigma(i) = j$$

where $p_{i,j} = 1$.

**Definition (5.2.3).** Consider the matrix A and an ordered
grouping $\pi$, then there exists a permutation matrix $P(\pi)$ associated
with the permutation function $\sigma$ where

a) $\sigma(i) < \sigma(j)$ or $\sigma(i) > \sigma(j)$ if the group containing $i$ comes
before or after, respectively, the group containing $j$.,
b) if \( i \) and \( j \) belong to the same group and \( i \neq j \), then
\( \sigma(i) < \sigma(j) \) or \( \sigma(i) > \sigma(j) \) if \( i < j \) or \( i > j \), respectively,
such that the ordered grouping \( \Pi \) defines a partitioning for the matrix

\[
\overline{A}^{(\Pi)} = (P^{(\Pi)})^{-1} A P^{(\Pi)} .
\]

\( (5.2.22) \)

\[
\begin{bmatrix}
A_{1,1} & A_{1,2} & \ldots & A_{1,N} \\
A_{2,1} & A_{2,2} & \ldots & A_{2,N} \\
\vdots & \vdots & \ddots & \vdots \\
A_{N,1} & A_{N,2} & \ldots & A_{N,N}
\end{bmatrix}
\]

As an example, consider the matrix

\[
A =
\begin{bmatrix}
\begin{array}{cccc}
a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\
a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \\
a_{31} & a_{32} & a_{33} & a_{34} & a_{35} \\
a_{41} & a_{42} & a_{43} & a_{44} & a_{45} \\
a_{51} & a_{52} & a_{53} & a_{54} & a_{55}
\end{array}
\end{bmatrix}
\]

and the ordered grouping \( \pi \) defined by

\[
G_1 = \{1,3,5\}, \quad G_2 = \{2,4\}
\]

then we have

\( \sigma(1) = 1, \quad \sigma(3) = 2, \quad \sigma(5) = 3, \quad \sigma(2) = 4 \) and \( \sigma(4) = 5 \)

so that the corresponding permutation matrix is
Therefore, we obtain
\[
\overline{A}(\pi) = (P(\pi))^{-1} A(P(\pi)) = \\
\begin{bmatrix}
  a_{11} & a_{13} & a_{15} & a_{12} & a_{14} \\
  a_{31} & a_{33} & a_{35} & a_{32} & a_{34} \\
  a_{51} & a_{53} & a_{55} & a_{52} & a_{54} \\
  a_{21} & a_{23} & a_{25} & a_{22} & a_{24} \\
  a_{41} & a_{43} & a_{45} & a_{42} & a_{44}
\end{bmatrix}
\]

5.3 BLOCK ITERATIVE METHODS.

Consider the linear system of equations \( A x = b \). Let \( \pi \) be a partition of the matrix \( A \) of the form
\[
A = \\
\begin{bmatrix}
  A_{1,1} & A_{1,2} & \ldots & A_{1,N} \\
  A_{2,1} & A_{2,2} & \ldots & A_{2,N} \\
  \vdots & \vdots & \ddots & \vdots \\
  A_{N,1} & A_{N,2} & \ldots & A_{N,N}
\end{bmatrix}
\]

(5.3.1)

where \( N < n \) and \( A_{k,k} \), \( k = 1, 2, \ldots, N \) are square matrices, we shall assume that all \( A_{k,k} \) are non-singular. The following notation will be used throughout this section. Let
\[
A = D(\pi) - C(\pi)
\]

(5.3.2)
where

\[
D^{(\pi)} = \begin{bmatrix}
A_{1,1} & & & \\
 & A_{2,2} & & \\
& & & 0 \\
& & & A_{N,N}
\end{bmatrix},
\]  

(5.3.3)

and

\[
C^{(\pi)} = E^{(\pi)} - F^{(\pi)},
\]

(5.3.4)

where \( E^{(\pi)} \) and \( F^{(\pi)} \) are respectively strictly lower and strictly upper triangular matrices. Since \( A_{k,k} \) are non-singular then \( (D^{(\pi)})^{-1} \) exists, we can define the following matrices

\[
B^{(\pi)} = (D^{(\pi)})^{-1} C^{(\pi)} = L^{(\pi)} + U^{(\pi)}
\]

\[
L^{(\pi)} = (D^{(\pi)})^{-1} E^{(\pi)}
\]

(5.3.5)

and

\[
U^{(\pi)} = (D^{(\pi)})^{-1} F^{(\pi)}
\]

where \( L^{(\pi)} \) and \( U^{(\pi)} \) are again strictly lower and strictly upper triangular matrices respectively.

After establishing the above notation we can now define various Block Iterative schemes as follows. The Block Jacobi method is given by the equation

\[
D^{(\pi)} x^{(n+1)} = C^{(\pi)} x^{(n)} + b,
\]

(5.3.6)

or

\[
x^{(n+1)} = B^{(\pi)} x^{(n)} + (D^{(\pi)})^{-1} b
\]

(5.3.7)

and the Block Gauss-Seidel method is given by
The Block successive overrelaxation (BSOR) method associated with the partition $\pi$ is defined by

$$
\mathbf{x}^{(n+1)} = \mathbf{L}^{(\pi)} \mathbf{x}^{(n)} + (I-L^{(\pi)})^{-1}D^{(\pi)}^{-1} \mathbf{b}, \quad (5.3.8)
$$

where

$$
\mathbf{L}^{(\pi)} = (I-L^{(\pi)})^{-1} \mathbf{U}^{(\pi)}. \quad (5.3.9)
$$

For subsequent analysis and convergence of the methods we note the following definitions, Lemmas and Theorems.

**Definition (5.3.1).** The matrix $A$ has property $(A^{(\pi)})$ for a given partition $\pi$ if there exists two disjoint subsets $s$ and $t$ of $\mathbb{W}$, the set of the first $N$ positive integers, such that $s \cup t = \mathbb{W}$ and such that if $A_{k\ell} \neq 0$. Then either $k = \ell$ or $k \in s$ and $\ell \in t$ or $k \in t$ and $\ell \in s$, [ARMS, GATES and ZONDAK, 1956].

**Lemma (5.3.1).** If $A$ is symmetric and $D^{(\pi)}$ is positive definite, then $B^{(\pi)}$ is similar to a symmetric matrix [YOUNG, 1954].

**Lemma (5.3.2).** If $A$ is symmetric and has property $(A^{\pi})$ and $D^{(\pi)}$
is positive definite, then $\rho(B^{(\pi)}) < 1$ if and only if $A$ is positive definite [ARMS, et al, 1956].

From Ostrowski [1954] and Varga [1962] we obtain:

**Theorem (5.3.1).** If $A$ is symmetric and $D^{(\pi)}$ is positive definite, then $\rho(L^{(\pi)}) < 1$ if and only if $A$ is positive definite and $0 < \omega < 2$.

For a symmetric matrix with $D^{(\pi)}$ positive definite Theorem (5.3.1) shows the (BSOR) method converges. The question arises as to whether there is an optimum $\omega$ in the sense $R(L^{(\pi)}) > R(L^{(\omega)})$, $\omega \neq \omega_b$.

The answer is the same as for the point iterative method. i.e. it is possible for matrices which possess property $(A^{(\pi)})$.

By methods used in [YOUNG, 1954], which we mentioned earlier in Chapter 3 for point iterative methods, it is easy to show that if a matrix has property $(A^{\pi})$, then by a suitable permutation of its rows and columns it can be written in the form

$$A = \begin{bmatrix} D_1 & F \\ E & D_2 \end{bmatrix},$$

where the elements of $D_1, D_2, E$ and $F$ are the $A_k, \ell$ of the partition $\pi$ of (5.3.1) and where $D_1$ and $D_2$ are block diagonal matrices. Hence, the $D_1 \subseteq$ could be full matrices for some partitions. We also have the following theorem the proof of which can be found in Forsythe and Wasow, [1960].
Theorem (5.3.2). A matrix $A$ has property $(A^{(\pi)})$ for a given partition $\pi$ if and only if there exists a permutation matrix $P$ such that $P A P^T$ has a block tridiagonal representation of the form

$$P A P^T = \begin{bmatrix}
D_1 & F_1 & 0 & 0 & \cdots & 0 & 0 & 0 \\
E_1 & D_2 & F_2 & 0 & \cdots & 0 & 0 & 0 \\
0 & E_2 & D_3 & F_3 & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \cdots & E_{N-2} & D_{N-1} & F_{N-1} \\
0 & 0 & 0 & 0 & \cdots & 0 & E_{N-1} & D_N
\end{bmatrix}, \quad (5.3.14)$$

where each submatrix has the blocks $A_{k,l}$ of (5.3.1) as elements and where the $D_i$ are block diagonal matrices with elements $A_{k,k}$ of (5.3.1).

It can be seen that the permutation matrix $P$ is such that the submatrices of (5.3.1) are to be permuted as blocks.

Definition (5.3.2) [YOUNG, 1971]. To any matrix $A$ with partition $\pi$ we correspond a matrix $Z$ defined as

$$z_{k,l} = 1, \quad \text{if} \quad A_{k,l} \neq 0$$

and

$$z_{k,l} = 0, \quad \text{if} \quad A_{k,l} = 0$$

then the matrix $A$ is said to be $\pi$-consistently ordered if $Z$ is consistently ordered. Related to the ordering vectors for general matrices, we have an ordering $N$-tuple for $A$ as an $N$-tuple

$$\gamma^{(\pi)} = (\gamma_1^{(\pi)}, \gamma_2^{(\pi)}, \ldots, \gamma_N^{(\pi)}),$$

where each $\gamma_k^{(\pi)}$ is an integer,
such that, if \( A_{k,l} \neq 0 \), and \( k \neq l \) then \( |\gamma_k^{(\pi)} - \gamma_l^{(\pi)}| = 1 \). It can be easily verified that the matrix \( A \) with partition \( \pi \) has property \( A^{(\pi)} \) if and only if there exists an ordering \( N \)-tuple for \( A \).

[ARMS, et al., 1956]

**Definition (5.3.3).** A matrix \( A \) will be said to be ordered by \( \gamma^{(\pi)} \) if, whenever \( A_{k,l} \neq 0 \) and \( \gamma_k^{(\pi)} > \gamma_l^{(\pi)} \), the \( k \)th row of the block follows the \( l \)th row, and, if \( \gamma_k^{(\pi)} > \gamma_l^{(\pi)} \), the \( l \)th row follows the \( k \)th row. This definition in fact is a generalization of the definition of a consistent ordering as given in Chapter 3.

**Lemma (5.3.3).** If \( A \) has property \( A^{(\pi)} \) and \( \mu \) is an eigenvalue of \( B^{(\pi)} \), then \(-\mu\) is an eigenvalue of \( B^{(\pi)} \).

**Theorem (5.3.3).** If \( A \) has property \( A^{(\pi)} \) and is consistently ordered, with \( 0 < \omega < 2 \), and if \( \lambda \) is a non-zero eigenvalue of \( \mathcal{B}_\omega^{(\pi)} \) and \( \mu \) satisfies

\[
(\lambda + \omega - 1)^2 = \omega^2 \lambda \mu^2 ,
\]

then \( \mu \) is an eigenvalue of \( B^{(\pi)} \). Also for the converse we have \( \mu \) as an eigenvalue of \( B^{(\pi)} \), and if \( \lambda \) satisfies (5.3.15), then \( \lambda \) is an eigenvalue of \( \mathcal{B}_\omega^{(\pi)} \).

**Theorem (5.3.4).** If \( A \) is positive definite with property \( (A^{(\pi)}) \) and is consistently ordered, with

\[
\omega_b = \frac{2}{1 + \sqrt{1 - \mu^2}}
\]

(5.3.16)
where \( \bar{\mu} = \rho(B(\pi)) \), then

\[ \rho(\mathcal{L}_{\omega_b}^{(\pi)}) < \rho(\mathcal{L}_{\omega}^{(\pi)}) , \quad \omega \neq \omega_b \]  \hspace{1cm} (5.3.17)

and

\[ \rho(\mathcal{L}_{\omega}^{(\pi)}) = \omega_b - 1 \quad \text{for} \quad \omega_b \leq \omega < 2 . \]  \hspace{1cm} (5.3.18)

**Theorem (5.3.5).** If \( A \) is positive definite, has property \( A(\pi) \) and is consistently ordered, then

\[ R_{\infty}(\mathcal{L}_{1}^{(\pi)}) = 2R_{\infty}(B(\pi)) . \]  \hspace{1cm} (5.3.19)

**Theorem (5.3.6).** If \( A \) satisfies the hypothesis of theorem (5.3.5), and \( \bar{\mu} = \rho(B(\pi)) \), then

\[ \lim_{\bar{\mu} \to 1} \frac{R_{\infty}(\mathcal{L}_{\omega_b}^{(\pi)})}{\bar{\mu}} = 2 . \]  \hspace{1cm} (5.3.20)

Theorems (5.3.5) and (5.3.6) show the effectiveness of the (BSOR) method. From Theorem (5.3.6), by applying (3.4.50) we obtain

\[ \ln n_{\omega_b} \approx \frac{\sqrt{-\ln \epsilon}}{2} \sqrt{n_1} , \]  \hspace{1cm} (5.3.21)

where \( n_{\omega} \) is the number of iterations necessary for the indicated degree of convergence, which illustrates the efficiency of the method.

However, it is of interest to know which partition \( \pi \) will yield the fastest convergence. This can be partially answered by considering the following theorem, see [ARMS et al., 1956] and [VARGA, 1960].
Theorem (5.3.7). Let A be a L-matrix, positive definite, have property $A^{(\pi)}$ for each $\pi$ and be consistently ordered, then

$$R(\mathcal{L}_\omega^{(\pi)}) > R(\mathcal{L}_\omega^{(\pi_0)})$$

(5.3.22)

where $\mathcal{L}_\omega^{(\pi_0)}$ denote the point SOR matrix.

Indeed, under the assumptions of the above theorem, Varga [1960] shows that the more non-zero elements $D^{(\pi)}$ contains, the faster the convergence. In the limiting case, when $D^{(\pi)} = A$, we would obtain the solution $\mathbf{x} = (D^{(\pi)})^{-1} \mathbf{b}$ in one iteration.

As pointed out earlier a faster rate of convergence can always be obtained by selecting a partitioning which results in blocks which have more non-zero elements. However, the amount of work involved in each iteration will have a reverse effect. That is, the amount of work will sharply increase in each iteration with such partitionings.

The choice of the partitioning used to solve a problem with their relative effects and efficiencies have been widely studied in recent years.

In the next section we shall describe briefly a few alternative partitionings and introduce two new methods of grouping the points and show that the amount of work can be dramatically decreased.

5.4 THE SOLUTION OF ELLIPTIC B.V. PROBLEMS BY BLOCK ITERATIVE METHODS.

In the previous sections, we presented group and block iterative methods and their comparisons and it was shown that any arbitrary grouping of equations is equivalent to a "partitioning" of the coefficient matrix. In this section we study alternative partitioning
strategies of a system of linear equations arising from the five-point finite difference equations as considered in Chapter 2.

For purposes of illustration, let us consider the following model problem. Given a function \( g(x,y) \) defined and continuous on the boundary \( \partial \Omega \) of the unit square \( \Omega, \ 0 \leq x, y \leq 1 \), and given the function \( f(x,y) \) defined and continuous in \( \Omega \), we seek a function \( u(x,y) \) which is continuous and twice differentiable in the interior \( R \) of \( \Omega \) and which satisfies in \( R \) the Poisson differential equation

\[
\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f(x,y) \ , \tag{5.4.1}
\]

the function \( u(x,y) \) is required to satisfy on \( \partial \Omega \) the Dirichlet condition

\[
u(x,y) = g(x,y) \ . \tag{5.4.2}
\]

To apply the method of finite differences as described in Chapter 2, we replace the differential equation (5.4.1) by their usual five-point difference approximations. With the values of \( u(x,y) \) at the interior points of \( R \) considered as unknowns and with the values on \( \partial \Omega \) determined by the boundary condition (5.4.2) we get a system of linear algebraic equations and by collecting all known terms to the right hand side, we obtain a linear system of the form

\[
A \ u = b \ . \tag{5.4.3}
\]

Consider the region in Figure (5.1):
By a numbering of the mesh points along the rows as shown above, the matrix $A$ is of the form given in (5.3.14) and where the diagonal $(N \times N)$ submatrices $D_k$, $k = 1, 2, \ldots, N$, are of the form

$$D_k = \begin{bmatrix}
4 & -1 \\
-1 & 4 & -1 \\
& -1 & 4 & -1 \\
& & -1 & 4
\end{bmatrix}. \quad (5.4.4)$$

As a simple example, let $h = \frac{1}{5}$ then, we have the following region
and the set of equations in the unknowns $u_1, u_2, \ldots, u_{16}$ is

$$
\begin{bmatrix}
4 & -1 & & & & & & & & & & & & & & \\
-1 & 4 & -1 & & & & & & & & & & & & & \\
& -1 & 4 & -1 & & & & & & & & & & & & & \\
\end{bmatrix}
= 
\begin{bmatrix}
u_1 \\
u_2 \\
u_3 \\
u_4 \\
u_5 \\
u_6 \\
u_7 \\
u_8 \\
u_9 \\
u_{10} \\
u_{11} \\
u_{12} \\
u_{13} \\
u_{14} \\
u_{15} \\
u_{16}
\end{bmatrix} = 
\begin{bmatrix}b_1 \\
b_2 \\
b_3 \\
b_4 \\
b_5 \\
b_6 \\
b_7 \\
b_8 \\
b_9 \\
b_{10} \\
b_{11} \\
b_{12} \\
b_{13} \\
b_{14} \\
b_{15} \\
b_{16}
\end{bmatrix}
$$

The unknowns $u_i, (i = 1, 2, \ldots, N^2)$ are evidently partitioned into $N$ blocks in the following order:

**Block 1** $u_i, \quad i = 1, 2, \ldots, N$

**Block 2** $u_i, \quad i = N+1, N+2, \ldots, 2N$

and in general **Block $k$** $u_i, \quad i = (k-1)N+1, (k-1)N+2, \ldots, kN$,
for \( k = 1,2,\ldots,N \), and where the submatrices \( A_{k,k} \) of (5.3.1), \( k = 1,2,\ldots,N \) are of the form shown in (5.4.4). By this partitioning, which henceforth we call the \( \pi_1 \)-partition, the corresponding iterative methods are known as 1-line iterative methods.

By using the method of Frankel [1950] it can be shown that [Arms et al., 1956] the eigenvalues \( \mu \) of \( B \) (Definition 5.3.5) are of the form

\[
\mu_{p,q} = \cos \frac{q\pi h}{2 - \cos p\pi h}, \quad p,q = 1,2,\ldots,h^{-1}-1,
\]

and then it can be verified that

\[
\frac{\mu}{\mu_{p,q}} = \rho(B^{-1}) = \cos \frac{\pi h}{2 - \cos \pi h} \approx 1 - \pi^2 h^2.
\]

By comparing (5.4.6) with

\[
\rho(B^{-1}) = \cos \pi h \approx 1 - \frac{1}{2} \pi^2 h^2
\]

for the point Jacobi method, it can be seen that the rate of convergence of the 1-line Jacobi method is approximately twice that of the point Jacobi method for small \( h \). Also to determine the optimum relaxation parameter \( \omega_b \) in (5.3.16), i.e.

\[
\omega_b = \frac{2}{1 + \sqrt{\frac{\mu}{\mu_1}}}
\]

and by (5.3.18) it can be shown [Parter, 1961] that the 1-line SOR method converges approximately \( (2)^{1/2} \) times as fast as the point SOR method.
The next partitioning strategy which is considered here is a 2-line grouping. In this partitioning we place the mesh points of the first two rows of Figure (5.1) in the first block and the mesh points of the third and fourth rows in the second block and so on. The matrix $A$ still retains the form of (5.3.14), but the $D_k$ are now of the following structure.

\[
\begin{bmatrix}
4 & -1 & 0 & . & . & . & -1 \\
-1 & 4 & -1 & 0 & . & . & -1 \\
0 & . & . & . & . & . & . \\
. & . & . & . & . & . & . \\
. & -1 & 4 & 0 & . & . & -1 \\
-1 & . & . & 0 & 4 & -1 & 0 & . & . \\
. & . & . & . & . & . & . \\
. & . & . & . & . & . & . \\
-1 & . & . & . & -1 & 4 & -1 \\
-1 & . & . & . & -1 & 4
\end{bmatrix}
\]

which is a $(2N+1)$ bandwidth matrix and can be inconvenient for direct computation, particularly when $N$ is large, this is due to the fill-ins which occur during the direct solution process and therefore, destroys the existing sparsity in $D_k$. To overcome this difficulty the mesh points are re-ordered so that the problem remains unchanged and $D_k$ have a more suitable form.

Consider now a re-ordering of the points as shown in (Fig. 5.3) so that each individual diagonal block of the coefficient matrix $A$, still consists of mesh points on two adjacent lines. Note that $N$ must be an even number in order to have an exact number of 2-line blocks.
The coefficient matrix $A$ is again of the form (5.3.14) where $D_k$ is a square matrix of order $(2N \times 2N)$ and is quindiaogal in structure, i.e. containing no more than 4 non-zero elements in each row of the matrix and each $F_k(E_k)$ is a sparse matrix of order $(2N \times 2N)$. For example, in the case of $N = 4$, $D_k$ has the form:

$$
D_k = \begin{bmatrix}
4 & -1 & -1 \\
-1 & 4 & 0 & -1 & & 0 \\
-1 & 0 & 4 & -1 & -1 \\
-1 & -1 & 4 & 0 & -1 \\
-1 & 0 & 4 & -1 & -1 \\
0 & -1 & 0 & 4 & -1 \\
-1 & -1 & 4 & -1 & -1
\end{bmatrix}
$$
and $F_k$ has the form:

$$
F_k = \begin{bmatrix}
0 & . & . & . & . & . & . \\
-1 & 0 & . & . & . & . & . \\
0 & 0 & 0 & . & . & . & . \\
0 & 0 & -1 & . & . & . & . \\
. & . & . & 0 & . & . & . \\
. & . & . & -1 & . & . & . \\
. & . & . & . & 0 & . & . \\
. & . & . & . & . & -1 & . \\
\end{bmatrix}
$$

Note that the direct solution of this form of the numbering of the mesh points can now be carried out more efficiently without creating any non-zero elements in the $D_k$ matrices.

The corresponding iterative methods based on this form of partitioning (called the $\pi_2$-partition) are known as 2-line iterative methods.

As mentioned in Chapter 2, the matrix $A$, derived from the five-point finite difference approximation on a square mesh to a self-adjoint elliptic partial differential equation for a bounded region is:

(i) real symmetric,

(ii) diagonally dominant,

(iii) irreducible.

Thus, $A$ is a positive definite matrix with non-positive off-diagonal entries. Therefore, $A$ is a Stieltjes matrix and $A^{-1} > 0$. 
For comparison of the rates of convergence of the point, 1-line and 2-line iterative methods, we consider the following splitting of the matrix A,

\[ A = D^{i_1} - C^{i_1} , \quad i = 0,1,2 \]  \hspace{1cm} (5.4.8)

where \( \pi_i , \ i = 0,1,2 \) refers, respectively, to the partitioning of the matrix A into single point, 1-line and 2-line blocks. In each case, the splitting can be seen to be a regular splitting (Def. 3.4.1).

For each value of \( i \), the corresponding Jacobi iterative method is given by

\[ D^{i_1} u^{(n+1)} = C^{i_1} u^{(n)} + k \]  \hspace{1cm} (5.4.9)

where the iteration process starts with an arbitrary initial guess \( u^{(0)} \).

Since \( D^{i_1} \) is non-singular, equation (5.4.9) can be re-written in the equivalent form of

\[ u^{(n+1)} = B^{i_1} u^{(n)} + (D^{i_1})^{-1} k \]  \hspace{1cm} (5.4.10)

and from the definition of regular splitting, it is evident that the matrix \( B^{i_1} \) given by

\[ B^{i_1} = (D^{i_1})^{-1} C^{i_1} , \quad i = 0,1,2 \]  \hspace{1cm} (5.4.11)

contains non-negative elements, i.e.

\[ B^{i_1} \geq 0 \]  \hspace{1cm} (5.4.12)
From (5.4.8) and (5.4.11) we have

$$B^{-1} = (A + C^{-1})^{-1} C^{-1} = (I + A^{-1} C^{-1})^{-1} A^{-1} C^{-1}. \quad (5.4.13)$$

Let

$$Q^{-1} = A^{-1} C^{-1} \quad (5.4.14)$$

then we obtain

$$B^{-1} = (I + Q^{-1})^{-1} Q^{-1}. \quad (5.4.15)$$

Since $A^{-1} > 0$ and $C^{-1} > 0$ then, the matrix $Q^{-1}$ is a non-negative matrix, i.e. $Q^{-1} > 0$.

Therefore, from Theorem (3.4.8) we get

$$\frac{\bar{\mu}(B^{-1})}{\bar{\mu}(Q^{-1})} < 1. \quad (5.4.16)$$

Hence, the iterative methods defined by (5.4.10) are convergent.

By definition of the three splittings in (5.4.8) it is evident that

$$C^0 > C^1 > C^2. \quad (5.4.17)$$

However, from (3.4.57) we obtain

$$0 < \bar{\mu}(B^2) < \bar{\mu}(B^1) < \bar{\mu}(B^0) < 1. \quad (5.4.18)$$

and from (3.4.58) we have

$$R_\infty(B^0) < R_\infty(B^1) < R_\infty(B^2). \quad (5.4.19)$$

i.e. the 2-line Jacobi iterative method is faster than the 1-line which in turn, is faster than the point Jacobi method.

We now obtain the relation between the rates of convergence
of Jacobi iterations based on the splittings (5.4.8) where the 
\( C^i \) are communitative with the matrix \( A \) [VARGA, 1960]. Note 
that for this problem, only \( C^0 \) and \( C^1 \) possess this property, 
i.e.

\[
AC^i = C^i A , \quad i = 0,1.
\]

For \( C^2 \), we have

\[
AC^2 = (C^2 A)^T.
\]

By Theorem (3.4.9) and (5.4.8) with \( C^0 > C^1 > C^2 > 0 \)
and the fact that each \( C^i \) is symmetric and \( C^0 \) and \( C^1 \) are 
commutative with respect to \( A \) we can see that, if \( \mu(Q^1) \rightarrow +\infty \)
then

\[
\frac{R_{\infty}(B^1)}{R_{\infty}(B^0)} \geq \frac{\mu(Q^1)}{\mu(C^0)} > 1 . \quad (5.4.20)
\]

and if \( \mu(Q^2) \rightarrow +\infty \) then

\[
\frac{R_{\infty}(B^2)}{R_{\infty}(B^1)} \geq \frac{\mu(Q^1)}{\mu(C^0)} > 1 . \quad (5.4.21)
\]

Consider now the application of over-relaxation techniques to 
these three regular splittings (5.4.8) of \( A \). The theory of 
successive Block over-relaxation is valid for the point, 1-line 
and 2-line partitionings of the mesh points, thus, the optimum 
values of the over-relaxation parameters are

\[
\omega_i = \frac{2}{1 + \sqrt{1 - \mu^2(B^i)}} , \quad i = 0,1,2 \quad , \quad (5.4.22)
\]
and from (5.3.18) we have

\[
\rho_b^{(\pi_1)} = \omega_{1}^{(\pi_1)} - 1 .
\]  

(5.4.23)

Therefore, as \( \bar{\mu}(B_1) \to 1^- \) and \( \bar{\mu}(B_2) \to 1^- \), then

\[
R_\infty \left[ \mathcal{L}_b^{(\pi_1)} \right] \sim 2 \sqrt{R_\infty \left[ \mathcal{L}_b^{(\pi_1)} \right]} = 2\sqrt{2} \sqrt{R_\infty (B_1)} .
\]  

(5.4.24)

and

\[
R_\infty \left[ \mathcal{L}_b^{(\pi_2)} \right] \sim 2 \sqrt{R_\infty \left[ \mathcal{L}_b^{(\pi_2)} \right]} = 2\sqrt{2} \sqrt{R_\infty (B_2)} .
\]  

(5.4.25)

Hence, by (5.4.24), (5.4.25) and (5.4.21) we obtain.

\[
\frac{R_\infty \left[ \mathcal{L}_b^{(\pi_2)} \right]}{R_\infty \left[ \mathcal{L}_b^{(\pi_1)} \right]} \sim \sqrt{\frac{R_\infty (B_2)}{R_\infty (B_1)}} \sqrt{\frac{\bar{\mu}(C_1)}{\bar{\mu}(C_2)}} > 1 .
\]  

(5.4.26)

Similarly, we have

\[
\frac{R_\infty \left[ \mathcal{L}_b^{(\pi_1)} \right]}{R_\infty \left[ \mathcal{L}_b^{(\pi_0)} \right]} \sim \sqrt{\frac{\bar{\mu}(C_0)}{\bar{\mu}(C_1)}} > 1 .
\]  

(5.4.27)
It can be estimated that \( \bar{\mu} (C_{1}) + 1, \bar{\mu} (C_{2}) + \frac{1}{4} \) as \( h \to 0 \), then from (5.4.26) and (5.4.27) we have

\[
\lim_{h \to 0} \frac{R_{\infty} \left[ L_{\pi_{2}} \right]}{\omega_{b}^{\pi_{2}}} > \sqrt{2}
\]

and

\[
\lim_{h \to 0} \frac{R_{\infty} \left[ L_{\pi_{1}} \right]}{\omega_{b}^{\pi_{1}}} > \sqrt{2}
\]

We now construct two novel groupings of mesh points in the following manner. Consider the net region depicted in (Fig. 5.1):

1) We place the 4 mesh points as shown in (Fig. 5.4) in each group

\[
\begin{array}{cccc}
2N+k & 2N+k+1 \\
N+k-1 & N+k & N+k+1 & N+k+2 \\
k-1 & k & k+1 & k+2 \\
k-N & k-N+1 & & \\
\end{array}
\]

(Fig. 5.4)
where \( k = (2N+1)(2) (\ell+1)N-1 \) and \( \ell = 0(2)N-2 \). In this case, where \( N \) is an even number we have the subset \( G_M \) of Definition (5.2.1), \( M = 1,2,\ldots,N^2/4 \), such that each \( G_M \) consists of four elements, \( \{k, k+1, N+k, N+k+1\} \) however, the matrix \( A_{M,M} \) is of order 4 and of the form

\[
A_{M,M} = \begin{bmatrix}
-1 & 1 & -1 & 0 \\
-1 & 4 & 0 & -1 \\
-1 & 0 & 4 & -1 \\
0 & -1 & -1 & 4
\end{bmatrix}, \quad M = 1,2,\ldots,N^2/4. \tag{5.4.30}
\]

Thus, for the model problem (5.4.1), we have the group Jacobi method given by

\[
\begin{align*}
4u_k^{(n+1)} & - (u_{k+1}^{(n+1)} + u_{N+k}^{(n+1)}) = u_k^{(n)} + u_{k-1}^{(n)} + f_k \\
4u_{k+1}^{(n+1)} & - (u_k^{(n+1)} + u_{N+k+1}^{(n+1)}) = u_k^{(n)} + u_{k-N+1}^{(n)} + f_{k+1} \\
4u_{N+k}^{(n+1)} & - (u_k^{(n+1)} + u_{N+k+1}^{(n+1)}) = u_k^{(n)} + u_{N+k-1}^{(n)} + f_{N+k} \\
4u_{N+k+1}^{(n+1)} & - (u_{k+1}^{(n+1)} + u_{N+k}^{(n+1)}) = u_{2N+k+1}^{(n)} + u_{N+k+2}^{(n)} + f_{N+k+1}
\end{align*} \tag{5.4.31}
\]

such that, each iteration involves the solution of \( N^2/4 \) systems of order 4, with the matrix given in (5.4.30). Evidently, it can be seen that the matrix \( A_{M,M} \) can be easily inverted. Therefore, in practice, group systems of the type (5.4.31) are solved beforehand and the iteration completed explicitly. As an example, the solution
of the model problem (5.4.1) in the case of the Laplace equation (i.e. \( f(x,y) = 0 \)), gives the system of equations (5.4.31) which may be rewritten explicitly as:

\[
\begin{align*}
    u_{k}^{(n+1)} &= \frac{1}{24} \left[ 7(u_{k-N}^{(n)} + u_{k-1}^{(n)}) + 2(u_{k-N+1}^{(n)} + u_{k+2}^{(n)} + u_{2N+k}^{(n)} + u_{N+k-1}^{(n)}) \\ &\quad + u_{N+k+2}^{(n)} + u_{2N+k+1}^{(n)} \right], \\
    u_{k+1}^{(n+1)} &= \frac{1}{24} \left[ 7(u_{k-N+1}^{(n)} + u_{k+2}^{(n)}) + 2(u_{k-N}^{(n)} + u_{k-1}^{(n)} + u_{N+k+2}^{(n)} + u_{2N+k+1}^{(n)}) \\ &\quad + u_{2N+k}^{(n)} + u_{N+k-1}^{(n)} \right], \\
    u_{N+k}^{(n+1)} &= \frac{1}{24} \left[ 7(u_{2N+k}^{(n)} + u_{N+k-1}^{(n)}) + 2(u_{k-N}^{(n)} + u_{k-1}^{(n)} + u_{N+k+2}^{(n)} + u_{2N+k+1}^{(n)}) \\ &\quad + u_{k-N+1}^{(n)} + u_{k+2}^{(n)} \right], \\
    u_{N+k+1}^{(n+4)} &= \frac{1}{24} \left[ 7(u_{N+k+2}^{(n)} + u_{2N+k+1}^{(n)}) + 2(u_{k-N+1}^{(n)} + u_{k+2}^{(n)} + u_{2N+k}^{(n)} + u_{N+k-1}^{(n)}) \\ &\quad + u_{k-N}^{(n)} + u_{k-1}^{(n)} \right],
\end{align*}
\]

where \( k = (\ell N+1)(2)(\ell+1)N-1 \) and \( \ell = 0(2)N-2 \). Note that, if any of \( u_{M}^{(n)} \) in the right hand sides of (5.4.32) is a point of the boundary \( \partial \Omega \) then we compute the appropriate value of \( u \) at this point by (5.4.2).

2) The second type of grouping of the mesh points which we are concerned with is as follows. We consider each group as being formed from 9 points of the net region (Fig. 5.1) in accordance with (Fig. 5.5).
where \( k = (\ell N+1)(3)(\ell+1)N-2 \) and \( \ell = O(3)N-3 \). For this scheme of grouping of the mesh points \( N \) must be divisible by 3. In this case, each subset \( G^M \), \( M = 1, 2, \ldots, N^2/9 \), of Definition (5.2.1) consists of 9 elements and the submatrices \( A^M \), \( M = 1, 2, \ldots, N^2/9 \), of the original matrix \( A \) are of order 9 of the form

\[
\begin{bmatrix}
4 & -1 & 0 & -1 & . & . & . & . & . \\
-1 & 4 & -1 & 0 & -1 & . & . & . & . \\
0 & -1 & 4 & 0 & 0 & -1 & . & . & . \\
-1 & 0 & 0 & 4 & -1 & 0 & -1 & . & . \\
. & -1 & 0 & -1 & 4 & -1 & 0 & -1 & . \\
. & . & -1 & 0 & -1 & 4 & 0 & 0 & -1 \\
. & . & . & -1 & 0 & 0 & 4 & -1 & . \\
. & . & . & . & -1 & 0 & -1 & 4 & -1 \\
. & . & . & . & . & -1 & 0 & -1 & 4
\end{bmatrix}
\]

\( A^M = \) \( M = 1, 2, \ldots, N^2/9 \) .

The matrix (5.4.33) again, can be easily inverted, and for the solution of the model problem (5.4.1) in the case \( f(x, y) = 0 \), (i.e.
Laplace equation) by the group Jacobi method where each group will be solved by the following formulae.

\[
\begin{align*}
\mathcal{u}_{k}^{(n+1)} & = \frac{1}{224} \left[ 67\mathcal{u}_{k-N}^{(n)} + \mathcal{u}_{k-1}^{(n)} + 22\mathcal{u}_{k-N+1}^{(n)} + \mathcal{u}_{k-N+2}^{(n)} + 7\mathcal{u}_{k-N+3}^{(n)} + \mathcal{u}_{k-3N+k}^{(n)} \\
& \quad \quad \quad + \mathcal{u}_{2N+k-1}^{(n)} + 6\mathcal{u}_{N+k+3}^{(n)} + \mathcal{u}_{3N+k+1}^{(n)} + 3\mathcal{u}_{2N+k+3}^{(n)} + \mathcal{u}_{3N+k+2}^{(n)} \right] , \\
\mathcal{u}_{k+1}^{(n+1)} & = \frac{1}{112} \left[ 37\mathcal{u}_{k-N+1}^{(n)} + 11\mathcal{u}_{k-N}^{(n)} + \mathcal{u}_{k-1}^{(n)} + \mathcal{u}_{k-N+2}^{(n)} + \mathcal{u}_{k+3}^{(n)} + 7\mathcal{u}_{k-N+3}^{(n)} + \mathcal{u}_{N+k-1}^{(n)} \\
& \quad \quad \quad + 5\mathcal{u}_{3N+k+1}^{(n)} + 3\mathcal{u}_{2N+k+3}^{(n)} + \mathcal{u}_{3N+k+2}^{(n)} + \mathcal{u}_{3N+k}^{(n)} + \mathcal{u}_{2N+k-1}^{(n)} \right] , \\
\mathcal{u}_{k+2}^{(n+1)} & = \frac{1}{224} \left[ 67\mathcal{u}_{k-N+2}^{(n)} + \mathcal{u}_{k-3}^{(n)} + 22\mathcal{u}_{k-N+1}^{(n)} + \mathcal{u}_{k-N+3}^{(n)} + 7\mathcal{u}_{k-N+2}^{(n)} + \mathcal{u}_{2N+k+3}^{(n)} \\
& \quad \quad \quad + \mathcal{u}_{3N+k+2}^{(n)} + \mathcal{u}_{k-1}^{(n)} + 6\mathcal{u}_{N+k+1}^{(n)} + \mathcal{u}_{N+k-1}^{(n)} + 3\mathcal{u}_{3N+k}^{(n)} + \mathcal{u}_{2N+k-1}^{(n)} \right] , \\
\mathcal{u}_{N+k+1}^{(n+1)} & = \frac{1}{112} \left[ 37\mathcal{u}_{N+k-1}^{(n)} + 11\mathcal{u}_{k-N}^{(n)} + \mathcal{u}_{N+k}^{(n)} + \mathcal{u}_{k-1}^{(n)} + \mathcal{u}_{k-N+1}^{(n)} + \mathcal{u}_{N+k+1}^{(n)} \\
& \quad \quad \quad + 3\mathcal{u}_{k-N+2}^{(n)} + \mathcal{u}_{k+3}^{(n)} + \mathcal{u}_{3N+k+2}^{(n)} + \mathcal{u}_{2N+k-1}^{(n)} \right] , \\
\mathcal{u}_{N+k+2}^{(n+1)} & = \frac{1}{16} \left[ 2\mathcal{u}_{k-N+1}^{(n)} + \mathcal{u}_{N+k+3}^{(n)} + \mathcal{u}_{3N+k+1}^{(n)} + \mathcal{u}_{N+k+3}^{(n)} + \mathcal{u}_{k-1}^{(n)} + \mathcal{u}_{k-N+2}^{(n)} + \mathcal{u}_{k+3}^{(n)} \\
& \quad \quad \quad + \mathcal{u}_{N+k+3}^{(n)} + \mathcal{u}_{3N+k+2}^{(n)} + \mathcal{u}_{N+k-1}^{(n)} + \mathcal{u}_{3N+k}^{(n)} + \mathcal{u}_{2N+k+1}^{(n)} \right] , \\
\mathcal{u}_{2N+k}^{(n+1)} & = \frac{1}{224} \left[ 67\mathcal{u}_{3N+k}^{(n)} + \mathcal{u}_{2N+k-1}^{(n)} + 22\mathcal{u}_{3N+k+1}^{(n)} + \mathcal{u}_{N+k-1}^{(n)} + 7\mathcal{u}_{k-N+1}^{(n)} + \mathcal{u}_{k-1}^{(n)} + \mathcal{u}_{2N+k+3}^{(n)} \\
& \quad \quad \quad + \mathcal{u}_{3N+k+2}^{(n)} + 6\mathcal{u}_{N+k+1}^{(n)} + \mathcal{u}_{N+k+3}^{(n)} + \mathcal{u}_{N+k+3}^{(n)} \right] .
\end{align*}
\]
\[
\begin{align*}
\frac{u^{(n+1)}}{2N+k+1} &= \frac{1}{112} \left[ 37 u^{(n)}_{3N+k+1} + 11 u^{(n)}_{2N+k+2} + u^{(n)}_{3N+k} + u^{(n)}_{2N+k-1} + 7(u^{(n)}_{N+k+3} + u^{(n)}_{N+k-1}) + 3(u^{(n)}_{k-N} + u^{(n)}_{k-N+2} + u^{(n)}_{k+3}) \right], \\
\frac{u^{(n+1)}}{2N+k+2} &= \frac{1}{224} \left[ 67 u^{(n)}_{2N+k+3} + u^{(n)}_{3N+k+2} + 22(u^{(n)}_{N+k+3} + u^{(n)}_{3N+k+1}) + 7(u^{(n)}_{k-N+2} + u^{(n)}_{k+3}) \\
&\quad + u^{(n)}_{3N+k} + u^{(n)}_{2N+k-1} + 6(u^{(n)}_{k-N+1} + u^{(n)}_{N+k-1}) + 3(u^{(n)}_{k-N} + u^{(n)}_{k-1}) \right],
\end{align*}
\]

where \( k = (\ell N + 1)(3)(\ell + 1)N - 2 \) and \( \ell = 0(3)N - 3 \).

The novelty of these types of grouping of the mesh points lie in the following facts. As mentioned before, in the solution of the problem by group iterative methods, each group of mesh points can be considered as a single point, thus if we consider here each group of 4 (or 9) mesh points as shown in (Fig. 5.4) or (Fig. 5.5) as a single point \( P_M \) then, we have different methods of labeling the point \( P_M \), according to the choice of solving the groups successively; for example, if we chose the groups of mesh points corresponding to the first two (or three for 9 points grouping) rows from left to right and then the groups corresponding to third and fourth (or fourth, fifth and sixth in the case of 9 points) row and so on. Therefore, the labeling points \( P_M \) is known as the "natural" ordering and it can be seen that the matrix \( Z = (z_{ki}) \) of Definition (5.3.2) has property (A) and is consistently ordered. Thus, the matrix \( A \) has property \( (A^{(\pi)}) \) and is \( \pi \)-consistently ordered. By this consideration we have a different choice of labeling the points \( P_M \) which could result in the matrix \( A \) having property \( (A^{(\pi)}) \) and be \( \pi \)-consistently ordered, such as "red-black" ordering or "diagonal"
ordering. Therefore, in each case the theory of group successive over-relaxation is valid.

Another advantage with these types of grouping of mesh points is that, the resulting matrices of the groups are easily invertable and since all the matrices are the same therefore, only one matrix needs to be inverted and used thereafter. The inverse of these matrices can be seen to be obtained only with few operations, especially when the group size is small.
5.4.1 Experimental Results for S.O.R. Methods.

We now describe some numerical experiments in order to compare the new block methods with the standard point and line block S.O.R. methods. All the results, unless otherwise indicated, were obtained on the "ICL 1904S computer located in the Computer Center of Loughborough University of Technology". All the programs were written in FORTRAN.

Problem 1. The solution of the model problem, i.e. the Laplace's equation

\[
\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \quad \text{in } \Omega \quad (5.4.35)
\]

and

\[
u = 0 \quad \text{on } \partial \Omega \quad ,\quad (5.4.36)
\]

where \(\Omega\) is a unit square with boundary \(\partial \Omega\).

We use a red-black ordering of points of the mesh and also of blocks of 4 points and 9 points as shown in Figure 5.6.

(Fig. 5.6)
where each point on the mesh can represent a single point, a block of four points or a block of nine points depending on the size of the grouping scheme. Therefore, the resulting matrices are \( \pi \)-consistently ordered, thus the optimum \( \omega \), \( \omega_b \) (say) are calculated by the formula

\[
\omega_b = 2/(1 + (1 - \rho(G))^{1/2})
\]

(5.4.37)

where \( \rho(G) \) is the spectral radius of the Gauss-Seidel iteration matrix and is estimated by use of the Power Method from

\[
\rho(G) = \lim_{n \to \infty} \frac{\|d^{(n)}\|}{\|d^{(n-1)}\|},
\]

(5.4.38)

where \( d^{(n)} = u^{(n)} - u^{(n-1)} \) and the norm \( \|d^{(n)}\| \) can be defined as

\[
\|d^{(n)}\| = \left[ \sum_{i=1}^{N} (x_i^{(n)} - x_i^{(n-1)})^2 \right]^{1/2}.
\]

(5.4.39)

An experimental optimum value of \( \omega \) is also found by solving the problem using different values of \( \omega \). The condition for convergence of the S.O.R. methods is

\[
|u_{i,j}^{(n+1)}| < \varepsilon, \quad \text{for all } i,j
\]

(5.4.40)

where \( \varepsilon = 5 \times 10^{-6} \), and similarly for the Power method where the difference between successive estimates of \( \rho(G) \) is chosen to be less than the same value of \( \varepsilon \). Different mesh sizes are used which produce \((20 \times 20)\), \((26 \times 26)\), \((32 \times 32)\), \((38 \times 38)\), \((44 \times 44)\), \((50 \times 50)\), ... and \((98 \times 98)\) networks. The results obtained from these experiments
are recorded in Tables (5.4.1), (5.4.2), (5.4.3), (5.4.4) and (5.4.5), where \( n_e \) is the minimum number of iterations required for convergence. From equations (3.4.46), (3.4.50) and (5.3.18) it is not difficult to show that

\[
n \approx \frac{\ln \epsilon}{\ln(\omega - 1)} \quad \text{(5.4.40)}
\]

In each case we have estimated the number of iterations \( n_T \) (say) from equation (5.4.40).
<table>
<thead>
<tr>
<th>$\frac{1}{h}$</th>
<th>$\rho(B)$</th>
<th>$\omega_b$</th>
<th>$\omega_e$</th>
<th>$n_T$</th>
<th>$n_E$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mesh Spacing</td>
<td>Spectral Radius</td>
<td>Optimal Over-relaxation Parameter</td>
<td>Experimental Over-relaxation Parameter</td>
<td>No. of Iterations</td>
<td>Exp. No. of Iterations</td>
</tr>
<tr>
<td>19</td>
<td>0.986361</td>
<td>1.717336</td>
<td>1.72359</td>
<td>38</td>
<td>40</td>
</tr>
<tr>
<td>25</td>
<td>0.992115</td>
<td>1.777251</td>
<td>1.782251</td>
<td>50</td>
<td>52</td>
</tr>
<tr>
<td>31</td>
<td>0.994869</td>
<td>1.816253</td>
<td>1.826253</td>
<td>62</td>
<td>64</td>
</tr>
<tr>
<td>37</td>
<td>0.996397</td>
<td>1.843648</td>
<td>1.84740</td>
<td>74</td>
<td>75</td>
</tr>
<tr>
<td>43</td>
<td>0.997382</td>
<td>1.86706</td>
<td>1.86706</td>
<td>86</td>
<td>88</td>
</tr>
<tr>
<td>49</td>
<td>0.997945</td>
<td>1.879575</td>
<td>1.882705</td>
<td>95</td>
<td>99</td>
</tr>
<tr>
<td>55</td>
<td>0.998369</td>
<td>1.89449</td>
<td>1.89449</td>
<td>107</td>
<td>111</td>
</tr>
<tr>
<td>61</td>
<td>0.998674</td>
<td>1.904583</td>
<td>1.904583</td>
<td>118</td>
<td>123</td>
</tr>
<tr>
<td>67</td>
<td>0.998901</td>
<td>1.910453</td>
<td>1.910453</td>
<td>130</td>
<td>141</td>
</tr>
<tr>
<td>73</td>
<td>0.999074</td>
<td>1.917505</td>
<td>1.917505</td>
<td>142</td>
<td>152</td>
</tr>
<tr>
<td>79</td>
<td>0.999209</td>
<td>1.923527</td>
<td>1.923527</td>
<td>153</td>
<td>163</td>
</tr>
<tr>
<td>85</td>
<td>0.999317</td>
<td>1.928731</td>
<td>1.928731</td>
<td>165</td>
<td>175</td>
</tr>
<tr>
<td>91</td>
<td>0.999404</td>
<td>1.933271</td>
<td>1.933271</td>
<td>177</td>
<td>187</td>
</tr>
<tr>
<td>97</td>
<td>0.999476</td>
<td>1.937268</td>
<td>1.937268</td>
<td>188</td>
<td>198</td>
</tr>
</tbody>
</table>

**TABLE (5.4.1) POINT S.O.R.**

**PROBLEM 1.**
<table>
<thead>
<tr>
<th>Mesh Spacing</th>
<th>$\rho(B)$ Spectral Radius</th>
<th>$\omega_b$ Optimal Over-relaxation Parameter</th>
<th>$\omega_e$ Experimental Over-relaxation Parameter</th>
<th>$n_T$ No. of Iterations</th>
<th>$n_E$ Exp. No. of Iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>0.973090</td>
<td>1.625452</td>
<td>1.635452</td>
<td>26</td>
<td>30</td>
</tr>
<tr>
<td>25</td>
<td>0.984353</td>
<td>1.700378</td>
<td>1.710378</td>
<td>34</td>
<td>40</td>
</tr>
<tr>
<td>31</td>
<td>0.989791</td>
<td>1.750507</td>
<td>1.756757</td>
<td>44</td>
<td>47</td>
</tr>
<tr>
<td>37</td>
<td>0.992821</td>
<td>1.786335</td>
<td>1.79071</td>
<td>51</td>
<td>55</td>
</tr>
<tr>
<td>43</td>
<td>0.994679</td>
<td>1.813196</td>
<td>1.818196</td>
<td>62</td>
<td>63</td>
</tr>
<tr>
<td>49</td>
<td>0.995899</td>
<td>1.834072</td>
<td>1.837822</td>
<td>69</td>
<td>72</td>
</tr>
<tr>
<td>55</td>
<td>0.996744</td>
<td>1.85076</td>
<td>1.85451</td>
<td>78</td>
<td>81</td>
</tr>
<tr>
<td>61</td>
<td>0.997352</td>
<td>1.864403</td>
<td>1.868153</td>
<td>84</td>
<td>90</td>
</tr>
<tr>
<td>67</td>
<td>0.997804</td>
<td>1.875763</td>
<td>1.880763</td>
<td>92</td>
<td>99</td>
</tr>
<tr>
<td>73</td>
<td>0.998150</td>
<td>1.885369</td>
<td>1.890369</td>
<td>100</td>
<td>109</td>
</tr>
<tr>
<td>79</td>
<td>0.998420</td>
<td>1.893597</td>
<td>1.893597</td>
<td>109</td>
<td>118</td>
</tr>
<tr>
<td>85</td>
<td>0.996835</td>
<td>1.900724</td>
<td>1.905724</td>
<td>117</td>
<td>126</td>
</tr>
<tr>
<td>91</td>
<td>0.998809</td>
<td>1.906957</td>
<td>1.906957</td>
<td>125</td>
<td>134</td>
</tr>
<tr>
<td>97</td>
<td>0.998952</td>
<td>1.912454</td>
<td>1.912454</td>
<td>133</td>
<td>143</td>
</tr>
</tbody>
</table>

**TABLE (5.4.2) 1-LINE S.O.R. PROBLEM 1.**
<table>
<thead>
<tr>
<th>$h^{-1}$ Mesh Spacing</th>
<th>$\rho(B)$ Spectral Radius</th>
<th>$\omega_b$ Optimal Over-relaxation Parameter</th>
<th>$\omega_e$ Experimental Over-relaxation Parameter</th>
<th>$n_T$ No. of Iterations</th>
<th>$n_E$ Exp. No. of Iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>0.947535</td>
<td>1.515552</td>
<td>1.524927</td>
<td>18</td>
<td>21</td>
</tr>
<tr>
<td>25</td>
<td>0.969172</td>
<td>1.604642</td>
<td>1.613392</td>
<td>24</td>
<td>27</td>
</tr>
<tr>
<td>31</td>
<td>0.979783</td>
<td>1.666578</td>
<td>1.674087</td>
<td>30</td>
<td>33</td>
</tr>
<tr>
<td>37</td>
<td>0.985742</td>
<td>1.711939</td>
<td>1.719439</td>
<td>36</td>
<td>39</td>
</tr>
<tr>
<td>43</td>
<td>0.989413</td>
<td>1.746528</td>
<td>1.751528</td>
<td>42</td>
<td>46</td>
</tr>
<tr>
<td>49</td>
<td>0.991831</td>
<td>1.773747</td>
<td>1.788747</td>
<td>48</td>
<td>51</td>
</tr>
<tr>
<td>55</td>
<td>0.993508</td>
<td>1.795713</td>
<td>1.800713</td>
<td>53</td>
<td>57</td>
</tr>
<tr>
<td>61</td>
<td>0.994717</td>
<td>1.813806</td>
<td>1.818806</td>
<td>59</td>
<td>64</td>
</tr>
<tr>
<td>67</td>
<td>0.995618</td>
<td>1.828965</td>
<td>1.8239650</td>
<td>65</td>
<td>69</td>
</tr>
<tr>
<td>73</td>
<td>0.996307</td>
<td>1.841847</td>
<td>1.8468470</td>
<td>71</td>
<td>78</td>
</tr>
<tr>
<td>79</td>
<td>0.996845</td>
<td>1.852928</td>
<td>1.857928</td>
<td>77</td>
<td>83</td>
</tr>
<tr>
<td>85</td>
<td>0.997274</td>
<td>1.862561</td>
<td>1.867561</td>
<td>83</td>
<td>89</td>
</tr>
<tr>
<td>91</td>
<td>0.997621</td>
<td>1.871012</td>
<td>1.876012</td>
<td>88</td>
<td>97</td>
</tr>
<tr>
<td>97</td>
<td>0.997906</td>
<td>1.878485</td>
<td>1.878485</td>
<td>94</td>
<td>103</td>
</tr>
</tbody>
</table>

**TABLE (5.4.3) 2-LINES S.O.R.**

**PROBLEM 1.**
<table>
<thead>
<tr>
<th>$h^{-1}$ Mesh Spacing</th>
<th>$\rho(B)$ Spectral Radius</th>
<th>$\omega_b$ Optimal Over-relaxation Parameter</th>
<th>$\omega_e$ Experimental Over-relaxation Parameter</th>
<th>$n_T$ No. of Iterations</th>
<th>$n_E$ Exp. No. of Iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>0.973056</td>
<td>1.625265</td>
<td>1.635265</td>
<td>26</td>
<td>30</td>
</tr>
<tr>
<td>25</td>
<td>0.984344</td>
<td>1.700306</td>
<td>1.707806</td>
<td>35</td>
<td>37</td>
</tr>
<tr>
<td>31</td>
<td>0.989788</td>
<td>1.750474</td>
<td>1.755474</td>
<td>44</td>
<td>46</td>
</tr>
<tr>
<td>37</td>
<td>0.992820</td>
<td>1.786318</td>
<td>1.793818</td>
<td>54</td>
<td>56</td>
</tr>
<tr>
<td>43</td>
<td>0.994678</td>
<td>1.813186</td>
<td>1.817565</td>
<td>61</td>
<td>63</td>
</tr>
<tr>
<td>49</td>
<td>0.995899</td>
<td>1.834066</td>
<td>1.836570</td>
<td>67</td>
<td>72</td>
</tr>
<tr>
<td>55</td>
<td>0.996743</td>
<td>1.850756</td>
<td>1.85326</td>
<td>76</td>
<td>80</td>
</tr>
<tr>
<td>61</td>
<td>0.997352</td>
<td>1.86440</td>
<td>1.86690</td>
<td>84</td>
<td>88</td>
</tr>
<tr>
<td>67</td>
<td>0.997804</td>
<td>1.875761</td>
<td>1.878261</td>
<td>92</td>
<td>97</td>
</tr>
<tr>
<td>73</td>
<td>0.998150</td>
<td>1.885368</td>
<td>1.887868</td>
<td>100</td>
<td>107</td>
</tr>
<tr>
<td>79</td>
<td>0.998420</td>
<td>1.893596</td>
<td>1.898596</td>
<td>109</td>
<td>116</td>
</tr>
<tr>
<td>85</td>
<td>0.998635</td>
<td>1.900723</td>
<td>1.900723</td>
<td>117</td>
<td>123</td>
</tr>
<tr>
<td>91</td>
<td>0.998809</td>
<td>1.906956</td>
<td>1.909456</td>
<td>125</td>
<td>132</td>
</tr>
<tr>
<td>97</td>
<td>0.998952</td>
<td>1.912453</td>
<td>1.914953</td>
<td>133</td>
<td>141</td>
</tr>
</tbody>
</table>

**TABLE (5.4.4) 4-POINTS S.O.R.**

**PROBLEM 1.**
<table>
<thead>
<tr>
<th>( h^{-1} ) Mesh Spacing</th>
<th>( \rho(B) ) Spectral Radius</th>
<th>( \omega_b ) Optimal Over-relaxation Parameter</th>
<th>( \omega_e ) Experimental Over-relaxation Parameter</th>
<th>( n_T ) No. of Iterations</th>
<th>( n_E ) Exp. No. of Iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>0.960222</td>
<td>1.563431</td>
<td>1.573431</td>
<td>22</td>
<td>24</td>
</tr>
<tr>
<td>25</td>
<td>0.976738</td>
<td>1.646853</td>
<td>1.654353</td>
<td>29</td>
<td>31</td>
</tr>
<tr>
<td>31</td>
<td>0.984778</td>
<td>1.703844</td>
<td>1.708844</td>
<td>36</td>
<td>38</td>
</tr>
<tr>
<td>37</td>
<td>0.989278</td>
<td>1.745128</td>
<td>1.750128</td>
<td>43</td>
<td>45</td>
</tr>
<tr>
<td>43</td>
<td>0.992044</td>
<td>1.776369</td>
<td>1.781369</td>
<td>50</td>
<td>52</td>
</tr>
<tr>
<td>49</td>
<td>0.993864</td>
<td>1.800818</td>
<td>1.805818</td>
<td>55</td>
<td>59</td>
</tr>
<tr>
<td>55</td>
<td>0.995125</td>
<td>1.820465</td>
<td>1.825465</td>
<td>62</td>
<td>67</td>
</tr>
<tr>
<td>61</td>
<td>0.996034</td>
<td>1.836594</td>
<td>1.839094</td>
<td>68</td>
<td>73</td>
</tr>
<tr>
<td>67</td>
<td>0.996711</td>
<td>1.85007</td>
<td>1.85507</td>
<td>75</td>
<td>81</td>
</tr>
<tr>
<td>73</td>
<td>0.997228</td>
<td>1.861496</td>
<td>1.863996</td>
<td>82</td>
<td>88</td>
</tr>
<tr>
<td>79</td>
<td>0.997632</td>
<td>1.871307</td>
<td>1.873808</td>
<td>89</td>
<td>95</td>
</tr>
<tr>
<td>85</td>
<td>0.997954</td>
<td>1.879822</td>
<td>1.882322</td>
<td>95</td>
<td>102</td>
</tr>
<tr>
<td>91</td>
<td>0.998215</td>
<td>1.887281</td>
<td>1.889781</td>
<td>102</td>
<td>109</td>
</tr>
<tr>
<td>97</td>
<td>0.998429</td>
<td>1.893870</td>
<td>1.896370</td>
<td>109</td>
<td>117</td>
</tr>
</tbody>
</table>

**TABLE (5.4.5) 9-POINTS S.O.R. PROBLEM 1.**
The timing information for execution of the different schemes are recorded in Table (5.4.6).

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>61</td>
<td>370</td>
<td>367</td>
<td>272</td>
<td>303</td>
<td>202</td>
</tr>
<tr>
<td>67</td>
<td>494</td>
<td>489</td>
<td>354</td>
<td>333</td>
<td>279</td>
</tr>
<tr>
<td>73</td>
<td>634</td>
<td>639</td>
<td>476</td>
<td>438</td>
<td>361</td>
</tr>
<tr>
<td>79</td>
<td>798</td>
<td>808</td>
<td>595</td>
<td>557</td>
<td>445</td>
</tr>
<tr>
<td>85</td>
<td>993</td>
<td>1005</td>
<td>739</td>
<td>686</td>
<td>570</td>
</tr>
<tr>
<td>91</td>
<td>1218</td>
<td>1220</td>
<td>925</td>
<td>847</td>
<td>699</td>
</tr>
<tr>
<td>97</td>
<td>1468</td>
<td>1480</td>
<td>1118</td>
<td>1026</td>
<td>855</td>
</tr>
</tbody>
</table>

**TABLE (5.4.6)**

TIME IN MILL UNITS FOR EXECUTION WITH THE BEST EXPERIMENTAL VALUE OF \( \omega \).

**PROBLEM 1.**
To compare the efficiency of each of the iterative methods it is necessary to consider the computational complexity for each iteration as well as the number of iterations required for convergence. One way of estimating this computational complexity is to count the number of arithmetic operations performed per iteration.

In counting these operations we include subtraction with additions, since they take about the same time to execute, but list multiplications and divisions separately.

We now investigate the computational requirements of the point, line, 2-line, 4 point and 9 point block S.O.R. methods.

POINT S.O.R. METHOD.

From equation (3.4.17) it is evident that for the solution of our problem, using the 5-point finite difference formula, by the point S.O.R. method we have:

\[ U_{i,j}^{(n+1)} = U_{i,j}^{(n)} + \omega \ast (U_{i,j}^{*} - U_{i,j}^{(n)}) \]  

(5.4.41)

where

\[ U_{i,j}^{*} = 0.25 \ast (U_{i-1,j}^{(n+1)} + U_{i+1,j}^{(n)} + U_{i,j-1}^{(n+1)} + U_{i,j+1}^{(n)}) \] , for all \( i,j \)

(5.4.42)

represents the Gauss-Seidel solution of the problem.

Therefore, if we have \( m^2 \) internal points on our mesh region, then the total number of operations will be

\[ 5m^2 \text{ additions} + 2m^2 \text{ multiplications}. \]  

(5.4.43)
1-LINE S.O.R. METHOD.

By (5.4.4) it can be shown that the 1-line G.S. Method involves the solution of \( m \) system of equations of the form;

\[
\begin{bmatrix}
4 & -1 & & & \\
-1 & 4 & -1 & & \\
& -1 & 4 & -1 & \\
& & -1 & 4 & \\
& & & -1 & 4
\end{bmatrix}
\begin{bmatrix}
U_{i,1}^{(n+1)} \\
U_{i,2}^{(n+1)} \\
\vdots \\
\vdots \\
U_{i,m}^{(n+1)}
\end{bmatrix}
= 
\begin{bmatrix}
\rho_{i,1}^{(n)} \\
\rho_{i,2}^{(n)} \\
\vdots \\
\vdots \\
\rho_{i,m}^{(n)}
\end{bmatrix}, \quad i = 1, \ldots, m
\]

(5.4.44)

where, the right hand side, \( \rho_{i,j}^{(n)} \) \( (j = 1, 2, \ldots, m) \) consists of values at neighbouring mesh points not lying along the row \( i \), is defined as

\[
\rho_{i,1}^{(n)} = U_{i-1,1}^{(n+1)} + U_{i,0}^{(n+1)} + U_{i+1,1}^{(n)}
\]

(5.4.45)

\[
\rho_{i,j}^{(n)} = U_{i-1,j}^{(n+1)} + U_{i,j-1}^{(n)} + U_{i,j+1}^{(n)} \quad j = 2, 3, \ldots, (m-1)
\]

(5.4.46)

and

\[
\rho_{i,m}^{(n)} = U_{i-1,m}^{(n+1)} + U_{i,m-1}^{(n)} + U_{i,m+1}^{(n)}
\]

(5.4.47)

Efficient algorithms exists for solving the system of equation (5.4.44). A well-known algorithm for this problem is the Gaussian Elimination Method and is defined as follows:
g_1 = \frac{1}{4} \quad h_1 = r_{i,1} * g_1 \quad (5.4.48)

\begin{align*}
g_j &= \frac{1}{(4 - g_{j-1})} \\
h_j &= (r_{i,j} + h_{j-1}) * g_j
\end{align*}

\text{for } j = 2(1)m \quad (5.4.49)

then,

\begin{align*}
U_{i,m}^{(n+1)} &= h_m
\end{align*}

and

\begin{align*}
U_{i,j}^{(n+1)} &= h_j + g_j * U_{i,j+1}^{(n+1)} \quad \text{for } j = m-1(-1)1. \quad (5.4.50)
\end{align*}

The \((n+1)\)th iterates of the \(i\)th row are now re-defined by the extrapolation equation

\begin{align*}
U_{i,j}^{(n+1)} &= U_{i,j}^{(n)} + \omega * (U_{i,j}^{(n)} - U_{i,j}^{(n)}) \quad \text{for } j = 1(1)m, \quad (5.4.51)
\end{align*}

where \(U_{i,j}^{(n)} \quad (j = 1,2,\ldots,m)\) represents the Gauss-Seidel solution to the system of equations (5.4.44) which is defined in (5.4.50) as \(U_{i,j}^{(n+1)}\). It is important to remember that the \(g_j \quad (j = 1,2,\ldots,m)\) need only be calculated once since they remain constant for each system of equations and each iteration for the model problem. Thus the average work for \(m^2\) interior points for this implementation of the S.L.O.R. method is;

\begin{align*}
m(3m-1) \text{ multiplication } + 5m^2 \text{ additions}, \quad (5.4.52)
\end{align*}

with an additional \(m\) multiplications + \((m-1)\) additions before the
first iteration to evaluate the $g_j$ ($j = 1, 2, \ldots, m$).

**2-LINE S.O.R. METHOD.**

As mentioned before, it can be seen that the 2-line G-S method involves $\frac{m}{2}$ systems of equations of the form:

$$
\begin{bmatrix}
4 & -1 & -1 \\
-1 & 4 & 0 & -1 \\
-1 & 0 & 4 & -1 & -1 \\
-1 & -1 & 4 & 0 & -1 \\
-1 & -1 & 4 & 0 & -1 \\
-1 & -1 & 4 & 0 & -1 \\
\end{bmatrix}
\begin{bmatrix}
U^{(n+1)}_1 \\
U^{(n+1)}_2 \\
U^{(n+1)}_3 \\
U^{(n+1)}_4 \\
\vdots \\
U^{(n+1)}_m \\
\end{bmatrix}
= 
\begin{bmatrix}
\兼任^{(n)}_1 \\
\兼任^{(n)}_2 \\
\兼任^{(n)}_3 \\
\兼任^{(n)}_4 \\
\vdots \\
\兼任^{(n)}_m \\
\end{bmatrix}
$$

(5.4.53)

where the solution vector $\兼任^{(n+1)}$ is defined as:

$$
兼任^{(n+1)} = \begin{cases}
兼任^{(n+1)}_i, & \text{for } k = 1, 3, \ldots, (2m-1) \\
兼任^{(n+1)}_{i+1, k/2}, & \text{for } k = 2, 4, \ldots, 2m
\end{cases}
$$

(5.4.54)

and the right hand side vector $\兼任^{(n)}$ is defined as:

$$
兼任^{(n)}_1 =兼任^{(n+1)}_{i-1, 1} +兼任^{(n)}_{i, 0} \\
兼任^{(n)}_2 =兼任^{(n)}_{i+1, 0} +兼任^{(n)}_{i+2, 1}
$$

(5.4.55)
\[ r_k^{(n)} = \begin{cases} U_{i-1,(k+1)/2}^{(n+1)} & , \text{ for } k = 3, 5, \ldots, (2m-3) \\ U_{i+2,k/2}^{(n)} & , \text{ for } k = 4, 6, \ldots, (2m-2) \end{cases} \]

\[ r_{2m-1}^{(n)} = U_{i-1,m}^{(n+1)} + U_{i,m+1}^{(n)} \]

\[ r_{2m}^{(n)} = U_{i+1,m+1}^{(n)} + U_{i+2,m}^{(n)} \]

The coefficient matrix is quindiaonal and so can be solved using the Gaussian Elimination algorithm which can be expressed as follows;

\[ g_1 = 1/4 \quad a_1 = -g_1 \quad h_1 = r_1g_1 \quad (5.4.56) \]

\[ g_2 = 1/(4+a_1), \quad a_2 = -g_1g_2, \quad h_2 = (r_2+h_1)g_2, \]

and for \( k = 3(1)2m \)

\[ b_k^* = \begin{cases} a_{k-2} & , \text{ when } k \text{ is odd} \\ (a_{k-2}^{-1}) & , \text{ when } k \text{ is even} \end{cases} \quad (5.4.57) \]

\[ g_k = 1/(4-g_{k-2} - b_k a_{k-1}) \quad (5.4.58) \]

\[ a_k^* = \begin{cases} (b_{k-1} g_{k-1} - 1)g_k & , \text{ when } k \text{ is odd} \\ b_k g_k \quad (5.4.59) \quad \text{ when } k \text{ is even} \end{cases} \]

and
\[ h_k = (r_k + h_{k-2} - b_k h_{k-1}) g_k \]  

(5.4.60)

then

\[ u_{2m}^{(n+1)} = h_{2m} \]  

(5.4.61)

\[ u_{2m-1}^{(n+1)} = h_{2m-1} - a_{2m-1} u_{2m}^{(n+1)} \]  

(5.4.62)

and

\[ u_k^{(n+1)} = h_k - a_k u_{k+1}^{(n+1)} + g_k u_{k+2}^{(n+1)} \]  

for \( k = (2m-2)(-1)^1 \)

(5.4.63)

Therefore, as with the single line S.O.R. method, the application of the relaxation technique leads to the \((n+1)\)th iterates of the \(i\)th and \((i+1)\)th rows being redefined to give the S.O.R. formulae as

\[ u_{i,j}^{(n+1)} = u_{i,j}^{(n)} + \omega \ast (u_{2j-1,i}^* - u_{i,j}^{(n)}) \]  

for \( j = 1, 2, \ldots, m \)

\[ u_{i+1,j}^{(n+1)} = u_{i+1,j}^{(n)} + \omega \ast (u_{2j,i+1}^* - u_{i+1,j}^{(n)}) \]  

(5.4.64)

where \( u^* \) represents the Gauss-Seidel solution \( u_i^{(n+1)} \) defined in (5.4.61) - (5.4.63).

Again it is only necessary to calculate \( g_k, a_k \) and \( b_k \) \((k = 1(1)2m)\) once. Therefore the total of work per \( m^2 \) interior points per iteration is

\[ \frac{m}{2} (10m - 5) \text{ multiplication } + m(6m-1) \text{ additions} \]  

(5.4.65)

with an additional \((8m-5) \text{ multiplication } + (6m-5) \text{ additions before} \)
the first iteration to evaluate \( g_k, a_k \text{ and } b_k \) \((k = 1, 2, \ldots, 2m)\).

**4-POINT EXPLICIT BLOCK S.O.R. METHOD.**

We shall now calculate the number of operations per iteration when the grouping technique of 4 points is employed to solve our model problem. From equation (5.4.32) it can be seen that for the group of 4 points the Gauss-Seidel iterative method involves \( \frac{m^2}{4} \) systems of equations of the form:

\[
\begin{bmatrix}
4 & -1 & -1 & 0 \\
-1 & 4 & 0 & -1 \\
-1 & 0 & 4 & -1 \\
0 & -1 & -1 & 4
\end{bmatrix}
\begin{bmatrix}
U^{(n+1)}_{i,j} \\
U^{(n+1)}_{i,j+1} \\
U^{(n+1)}_{i+1,j} \\
U^{(n+1)}_{i+1,j+1}
\end{bmatrix}
= 
\begin{bmatrix}
r_{i,j} \\
r_{i,j+1} \\
r_{i+1,j} \\
r_{i+1,j+1}
\end{bmatrix}
\text{ for } i,j = 1(2)m
\]

(5.4.66)

where

\[
r_{i,j}^{(n)} = U^{(n+1)}_{i-1,j} + U^{(n+1)}_{i,j-1},
\]

(5.4.67)

\[
r_{i,j+1}^{(n)} = U^{(n+1)}_{i-1,j+1} + U^{(n)}_{i,j+2},
\]

\[
r_{i+1,j}^{(n)} = U^{(n+1)}_{i+1,j-1} + U^{(n)}_{i+2,j},
\]

and

\[
r_{i+1,j+1}^{(n)} = U^{(n)}_{i+1,j+2} + U^{(n)}_{i+2,j+1}.
\]

Next, we set
\[ S_1 = r_{i,j}^{(n)} + r_{i,j}^{(n)} + r_{i+1,j+1}^{(n)} + r_{i+1,j+1}^{(n)} \]  \hspace{1cm} (5.4.68)

and

\[ S_2 = r_{i,j+1}^{(n)} + r_{i,j+1}^{(n)} + r_{i+1,j}^{(n)} + r_{i+1,j}^{(n)} \]  \hspace{1cm} (5.4.69)

Therefore, the solution of the system (5.4.66) can be found from the formulae;

\[ u_{i,j}^{(n+1)} = \frac{1}{24} (7 r_{i,j}^{(n)} + S_2 + r_{i+1,j+1}^{(n)}) \]

\[ u_{i,j+1}^{(n+1)} = \frac{1}{24} (7 r_{i,j+1}^{(n)} + S_1 + r_{i,j+1}^{(n)}) \]  \hspace{1cm} (5.4.70)

\[ u_{i+1,j}^{(n+1)} = \frac{1}{24} (7 r_{i+1,j}^{(n)} + S_1 + r_{i+1,j}^{(n)}) \]

\[ u_{i+1,j+1}^{(n+1)} = \frac{1}{24} (7 r_{i+1,j+1}^{(n)} + S_2 + r_{i,j}^{(n)}) \]

Thus, the application of the over-relaxation technique leads to the (n+1)th iterates of the group of 4 points being redefined to give the S.O.R. formulae as;

\[ u_{i,j}^{(n+1)} = u_{i,j}^{(n)} + \omega (U^*_{i,j} - U_{i,j}^{(n)}) \]

\[ u_{i,j+1}^{(n+1)} = u_{i,j+1}^{(n)} + \omega (U^*_{i,j+1} - U_{i,j+1}^{(n)}) \]

\[ u_{i+1,j}^{(n+1)} = u_{i+1,j}^{(n)} + \omega (U^*_{i+1,j} - U_{i+1,j}^{(n)}) \]

\[ u_{i+1,j+1}^{(n+1)} = u_{i+1,j+1}^{(n)} + \omega (U^*_{i+1,j+1} - U_{i+1,j+1}^{(n)}) \]  \hspace{1cm} (5.4.71)
where again \( U^* \) represents the Gauss-Seidel solution \( U^{(n+1)} \) defined in (5.4.70) so this process requires

\[
3m^2 \text{ multiplications } + \frac{13}{2} m^2 \text{ additions} \quad (5.4.72)
\]

for \( m^2 \) internal mesh points per iteration, assuming that the constant \( \frac{1}{24} \) is stored beforehand.

We can reduce the number of operations by solving \( U^{(n+1)}_{i,j} \) and \( U^{(n+1)}_{i+1,j+1} \), and using these values to determine \( U^{(n+1)}_{i,j+1} \) and \( U^{(n+1)}_{i+1,j} \).

We calculate the values of \( r_{i,j}^{(n)} \), \( r_{i+1,j}^{(n)} \), \( r_{i,j+1}^{(n)} \), \( r_{i+1,j+1}^{(n)} \) and \( S_2 \), then \( U^{(n+1)}_{i,j} \) and \( U^{(n+1)}_{i+1,j+1} \) as before, and define

\[
S_3 = U^{(n+1)}_{i,j} + U^{(n+1)}_{i+1,j+1} \quad (5.4.73)
\]

Now we have;

\[
U^{(n+1)}_{i,j+1} = \frac{1}{24} \left( 7 r_{i,j+1}^{(n)} + S_1 + r_{i+1,j}^{(n)} \right)
\]

\[
= \frac{1}{12} r_{i,j}^{(n)} + \frac{7}{24} r_{i,j+1}^{(n)} + \frac{1}{12} r_{i+1,j}^{(n)} + \frac{1}{24} r_{i+1,j+1}^{(n)}
\]

\[
= \frac{1}{4} \left\{ r_{i,j+1}^{(n)} + \left\{ \frac{7}{24} r_{i,j}^{(n)} + \frac{1}{12} r_{i,j+1}^{(n)} + \frac{1}{12} r_{i+1,j}^{(n)} + \frac{1}{24} r_{i+1,j+1}^{(n)} \right\} \right\}
\]

\[
+ \left\{ \frac{7}{24} r_{i+1,j+1}^{(n)} + \frac{1}{24} r_{i,j}^{(n)} + \frac{1}{12} r_{i,j+1}^{(n)} + \frac{1}{12} r_{i+1,j}^{(n)} \right\}
\]

\[
= \frac{1}{4} \left\{ r_{i,j+1}^{(n)} + \frac{1}{24} \left\{ 7 r_{i,j}^{(n)} + S_2 + r_{i+1,j+1}^{(n)} \right\} \right\}
\]

\[
+ \frac{1}{24} \left\{ 7 r_{i+1,j+1}^{(n)} + S_2 + r_{i,j}^{(n)} \right\}
\]

\[
= \frac{1}{4} \left( r_{i,j+1}^{(n)} + U^{(n+1)}_{i,j} + U^{(n+1)}_{i+1,j+1} \right)
\]

\[
= \frac{1}{4} \left( r_{i,j+1}^{(n)} + S_3 \right) \quad (5.4.74)
\]
and similarly it can be shown that

$$U^{(n+1)}_{i+1,j} = \frac{1}{4} (r^{(n)}_{i+1,j} + S_j) \quad (5.4.75)$$

Therefore, the average work per iteration for $m^2$ internal mesh points including the over-relaxation process, is

$$\frac{5m^2}{2} \text{ multiplications} + \frac{11m^2}{2} \text{ additions} \quad (5.4.76)$$

**9-POINT EXPLICIT BLOCK S.O.R. METHOD.**

Finally, we consider the group of 9 points and calculate the amount of work to obtain the solution of the system of equations per iteration. From the formulae of (5.4.34) it can be seen that the 9-points group Gauss-Seidel iterative method for the solution of our model problem involves $\frac{m^2}{9}$ system of equations of the form:

$$
\begin{bmatrix}
4 & -1 & 0 & -1 \\
-1 & 4 & -1 & 0 & -1 \\
0 & -1 & 4 & 0 & 0 & -1 \\
-1 & 0 & 0 & 4 & -1 & 0 & -1 \\
-1 & 0 & -1 & 4 & -1 & 0 & -1 \\
-1 & 0 & -1 & 4 & 0 & 0 & -1 \\
-1 & 0 & 0 & 4 & -1 & 0 & -1 \\
-1 & 0 & -1 & 4 & -1 & 0 & -1 \\
-1 & 0 & -1 & 4 & -1 & 0 & -1 \\
\end{bmatrix}
\begin{bmatrix}
U^{(n+1)}_{i,j} \\
U^{(n+1)}_{i,j+1} \\
U^{(n+1)}_{i,j+2} \\
U^{(n+1)}_{i+1,j} \\
U^{(n+1)}_{i+1,j+1} \\
U^{(n+1)}_{i+1,j+2} \\
U^{(n+1)}_{i+2,j} \\
U^{(n+1)}_{i+2,j+1} \\
U^{(n+1)}_{i+2,j+2} \\
\end{bmatrix}
=
\begin{bmatrix}
\hat{r}^{(n)}_{i,j} \\
\hat{r}^{(n)}_{i,j+1} \\
\hat{r}^{(n)}_{i,j+2} \\
\hat{r}^{(n)}_{i+1,j} \\
\hat{r}^{(n)}_{i+1,j+1} \\
\hat{r}^{(n)}_{i+1,j+2} \\
\hat{r}^{(n)}_{i+2,j} \\
\hat{r}^{(n)}_{i+2,j+1} \\
\hat{r}^{(n)}_{i+2,j+2} \\
\end{bmatrix}
$$

for $i,j = 1(3)m \quad (5.4.77)$
where

\[ r_{i,j}^{(n)} = u_{i-1,j}^{(n+1)} + u_{i,j-1}^{(n+1)} \]

\[ r_{i,j+1}^{(n)} = u_{i-1,j+1}^{(n+1)} \]

\[ r_{i,j+2}^{(n)} = u_{i-1,j+2}^{(n+1)} + u_{i,j+3}^{(n)} \]  \hspace{1cm} (5.4.78)

\[ r_{i+1,j}^{(n)} = u_{i+1,j-1}^{(n+1)} \]

\[ r_{i+1,j+1}^{(n)} = 0 \]

\[ r_{i+1,j+2}^{(n)} = u_{i+1,j+3}^{(n)} \]

\[ r_{i+2,j}^{(n)} = u_{i+2,j-1}^{(n+1)} + u_{i+3,j}^{(n)} \]

\[ r_{i+2,j+1}^{(n)} = u_{i+3,j+1}^{(n)} \]

and

\[ r_{i+2,j+2}^{(n)} = u_{i+3,j+2}^{(n)} + u_{i+2,j+3}^{(n)} \]

Therefore, we have the "G-S" solution given by
\[
U_{i,j}^{(n+1)} = (67r_{i,j}^{(n)} + 22(r_{i,j+1}^{(n)} + r_{i+1,j}^{(n)}) + 7(r_{i,j+2}^{(n)} + r_{i+2,j}^{(n)}) \\
\hspace{2cm} + 6(r_{i+1,j+2}^{(n)} + r_{i+2,j+1}^{(n)}) + 3r_{i+2,j+2}^{(n)})/224
\]

\[
U_{i,j+1}^{(n+1)} = (37r_{i,j+1}^{(n)} + 11(r_{i,j}^{(n)} + r_{i,j+2}^{(n)}) + 7(r_{i+1,j+2}^{(n)} + r_{i+2,j+1}^{(n)}) \\
\hspace{2cm} + 5r_{i+2,j+1}^{(n)} + 3(r_{i+2,j+2}^{(n)})/112
\]

\[
U_{i,j+2}^{(n+1)} = (67r_{i,j+2}^{(n)} + 22(r_{i+1,j+1}^{(n)} + r_{i,j+2}^{(n)}) + 7(r_{i,j+1}^{(n)} + r_{i+2,j+2}^{(n)}) \\
\hspace{2cm} + 6(r_{i+2,j+1}^{(n)} + r_{i+1,j+2}^{(n)}) + 3r_{i+2,j+2}^{(n)})/224
\]

\[
U_{i+1,j}^{(n+1)} = (37r_{i+1,j}^{(n)} + 11(r_{i,j}^{(n)} + r_{i+2,j}^{(n)}) + 7(r_{i,j+1}^{(n)} + r_{i+2,j+1}^{(n)}) \\
\hspace{2cm} + 5r_{i+1,j+1}^{(n)} + 3(r_{i+2,j+2}^{(n)})/112
\]

\[
U_{i+1,j+1}^{(n+1)} = (2(r_{i,j}^{(n)} + r_{i,j+1}^{(n)} + r_{i+1,j}^{(n)} + r_{i+1,j+1}^{(n)}) \\
\hspace{2cm} + r_{i,j}^{(n)} + r_{i+1,j}^{(n)} + r_{i+2,j}^{(n)} + r_{i+2,j+1}^{(n)})/16
\]

\[
U_{i+1,j+2}^{(n+1)} = (37r_{i+1,j+2}^{(n)} + 11(r_{i,j+2}^{(n)} + r_{i+2,j+2}^{(n)}) + 7(r_{i,j+1}^{(n)} + r_{i+2,j+1}^{(n)}) \\
\hspace{2cm} + 5r_{i+1,j}^{(n)} + 3(r_{i,j+2}^{(n)} + r_{i+2,j}^{(n)})/112
\]

\[
U_{i+2,j}^{(n+1)} = (67r_{i+2,j}^{(n)} + 22(r_{i+2,j+1}^{(n)} + r_{i+1,j}^{(n)}) + 7(r_{i,j}^{(n)} + r_{i+2,j+2}^{(n)}) \\
\hspace{2cm} + 6(r_{i,j+1}^{(n)} + r_{i+1,j+2}^{(n)}) + 3r_{i,j+2}^{(n)})/224
\]
\[
\begin{align*}
U_{1+2, j+1}^{(n+1)} &= (37r_{i+2,j+1}^{(n)} + 11r_{1+2,j+2}^{(n)} + r_{i+2,j}^{(n)}) + 7(r_{1+1,j}^{(n)} + r_{1+1,j+2}^{(n)}) \\
&
\quad + 5r_{i,j+1}^{(n)} + 3(r_{i,j}^{(n)} + r_{i,j+2}^{(n)})/112 \\
U_{1+2, j+2}^{(n+1)} &= (67r_{i+2,j+2}^{(n)} + 22r_{1+1,j+2}^{(n)} + r_{i+2,j+1}^{(n)}) + 7(r_{1,j+2}^{(n)} + r_{1+2,j}^{(n)}) \\
&
\quad + 6(r_{i,j+1}^{(n)} + r_{i+1,j}^{(n)}) + 3r_{i,j}^{(n)}/224 \\
& \quad \text{(5.4.79)}
\end{align*}
\]

By application of over-relaxation technique on the \((n+1)\)th iteration of the group of 9 points we have the following formulae;

\[
\begin{align*}
U_{i+1, j+k}^{(n+1)} &= U_i^{(n)} + \omega(U_i^{*} - U_i^{(n)}) \\
& \quad \text{for } i,j = 1(3)n \& k,l = 0(1)2, \\
& \quad \text{(5.4.80)}
\end{align*}
\]

where \(U_i^{*}\) represents the Gauss-Seidel solution \(U_i^{(n+1)}\) defined by (5.4.79). Therefore, this scheme requires a total amount of computational work given by

\[
\begin{align*}
\frac{59}{9} m^2 \text{ multiplications} + \frac{85}{9} m^2 \text{ additions}. \\
& \quad \text{(5.4.81)}
\end{align*}
\]

Once again we can reduce the number of operations as follows. First we calculate the 4 points \(U_{i,j+1}^{(n+1)}, U_{i+1,j}^{(n+1)}, U_{i+1,j+2}^{(n+1)}\) and \(U_{i+2,j+1}^{(n+1)}\) from (5.4.79) then we set

\[
S_i = U_{i,j+1}^{(n+1)} + U_{i+1,j}^{(n+1)} \\
& \quad \text{(5.4.82)}
\]
\[ S_2 = U_{i,j+1}^{(n+1)} + U_{i+1,j+2}^{(n+1)} \]  
(5.4.83)

\[ S_3 = U_{i+1,j}^{(n+1)} + U_{i+2,j+1}^{(n+1)} \]  
(5.4.84)

and

\[ S_4 = U_{i+2,j+1}^{(n+1)} + U_{i+1,j+2}^{(n+1)} \]  
(5.4.85)

Then, it can be shown that

\[ U_{i,j}^{(n+1)} = \frac{1}{4} (r_{i,j}^{(n)} + S_1) \]

\[ U_{i,j+2}^{(n+1)} = \frac{1}{4} (r_{i,j+2}^{(n)} + S_2) \]

\[ U_{i+1,j+1}^{(n+1)} = \frac{1}{4} (S_1 + S_4) \]  
(5.4.86)

\[ U_{i+2,j}^{(n+1)} = \frac{1}{4} (r_{i+2,j}^{(n)} + S_3) \]

\[ U_{i+2,j+2}^{(n+1)} = \frac{1}{4} (r_{i+2,j+2}^{(n)} + S_4) \]

so the amount of work per iteration for this method including the over-relaxation process is

\[
\frac{34}{9} \, m \text{ multiplications} + \frac{53}{9} \, m \text{ additions} . \quad (5.4.87)
\]

The results shown in Tables (5.4.1) - (5.4.5) can be combined with the number of operations per iteration required by each method given in (5.4.43), (5.4.52), (5.4.65), (5.4.76) and (5.4.87), to give the total number of arithmetic operations required by each of the point and block S.O.R. methods, and these are recorded in Table (5.4.7).
<table>
<thead>
<tr>
<th>Method</th>
<th>h⁻¹</th>
<th>19</th>
<th>31</th>
<th>43</th>
<th>55</th>
<th>61</th>
<th>73</th>
<th>85</th>
<th>97</th>
</tr>
</thead>
<tbody>
<tr>
<td>POINT S.O.R. A</td>
<td>200m²</td>
<td>320m²</td>
<td>440m²</td>
<td>555m²</td>
<td>615m²</td>
<td>760m²</td>
<td>875m²</td>
<td>990m²</td>
<td></td>
</tr>
<tr>
<td>M</td>
<td>80m²</td>
<td>128m²</td>
<td>176m²</td>
<td>222m²</td>
<td>246m²</td>
<td>304m²</td>
<td>350m²</td>
<td>396m²</td>
<td></td>
</tr>
<tr>
<td>1-LINE S.O.R. A</td>
<td>150m²</td>
<td>235m²</td>
<td>315m²</td>
<td>405m²</td>
<td>450m²</td>
<td>545m²</td>
<td>630m²</td>
<td>670m²</td>
<td></td>
</tr>
<tr>
<td>M</td>
<td>90m²-30m</td>
<td>141m²-47m</td>
<td>189m²-63m</td>
<td>243m²-81m</td>
<td>270m²-90m</td>
<td>327m²-109m</td>
<td>378m²-126m</td>
<td>429m²-143m</td>
<td></td>
</tr>
<tr>
<td>2-LINE S.O.R. A</td>
<td>126m²-21m</td>
<td>198m²-33m</td>
<td>276m²-46m</td>
<td>342m²-57m</td>
<td>384m²-64m</td>
<td>468m²-78m</td>
<td>534m²-89m</td>
<td>618m²-103m</td>
<td></td>
</tr>
<tr>
<td>M</td>
<td>105m²-52.5m</td>
<td>165m²-82.5m</td>
<td>230m²-115m</td>
<td>285m²-142.5m</td>
<td>320m²-160m</td>
<td>390m²-195m</td>
<td>445m²-222.5m</td>
<td>515m²-257.5m</td>
<td></td>
</tr>
<tr>
<td>4-POINT S.O.R. A</td>
<td>165m²</td>
<td>253m²</td>
<td>346.5m²</td>
<td>440m²</td>
<td>484m²</td>
<td>588.5m²</td>
<td>676.5m²</td>
<td>775.5m²</td>
<td></td>
</tr>
<tr>
<td>M</td>
<td>75m²</td>
<td>115m²</td>
<td>157.5m²</td>
<td>200m²</td>
<td>220m²</td>
<td>267.5m²</td>
<td>307.5m²</td>
<td>352.5m²</td>
<td></td>
</tr>
<tr>
<td>9-POINT S.O.R. A</td>
<td>141 1/3 m²</td>
<td>223 7/9 m²</td>
<td>306 2/9 m²</td>
<td>394 5/9 m²</td>
<td>429 8/9 m²</td>
<td>518 2/9 m²</td>
<td>600 2/3 m²</td>
<td>689m²</td>
<td></td>
</tr>
<tr>
<td>M</td>
<td>90 2/3 m²</td>
<td>143 5/9 m²</td>
<td>196 4/9 m²</td>
<td>253 1/9 m²</td>
<td>275 7/9 m²</td>
<td>332 4/9 m²</td>
<td>385 1/3 m²</td>
<td>442m²</td>
<td></td>
</tr>
</tbody>
</table>

TABLE (5.4.7)
Problem 2. The solution of generalised Dirichlet problem

\[
\frac{\partial}{\partial x} \left( e^{10(x+y)} \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left( e^{10(x+y)} \frac{\partial u}{\partial y} \right) = 0 , \quad (5.4.88)
\]
in the unit square with the boundary condition \( u = 0 \) on all boundaries.

Using formula (2.4.92) the equation (5.4.41) was replaced by

\[
\begin{align*}
2 e^{10(x+y)} \left( e^{5h} + e^{-5h} \right) U_{i,j} &= e^{10(x+y)} e^{5h} U_{i+1,j} + e^{10(x+y)} e^{-5h} U_{i-1,j} \\
&+ e^{10(x+y)} e^{-5h} U_{i,j-1} + e^{10(x+y)} e^{5h} U_{i,j+1} . \quad (5.4.89)
\end{align*}
\]

By multiplying the system (5.4.89), i.e. \( A \mathbf{U} = \mathbf{b} \), by the diagonal matrix \( D_1 \), where the diagonal elements of \( D_1 \) are of the form \( e^{-10(x+y)} \) we obtain the system of equations

\[
\begin{align*}
2 \left( e^{5h} + e^{-5h} \right) U_{i,j} &= e^{5h} U_{i+1,j} + e^{-5h} U_{i-1,j} + e^{5h} U_{i,j+1} + e^{-5h} U_{i,j-1} \\
&= \mathbf{D}_1 \mathbf{A} \mathbf{D}_1 \quad (5.4.90)
\end{align*}
\]

which leads to an unsymmetric matrix. Further, by choosing the diagonal matrix \( T = D_1^{-\frac{1}{2}} \) and applying a similarity transformation we obtain

\[
T(D_1 \mathbf{A})T^{-1} = D_1^{-\frac{1}{2}}(D_1 \mathbf{A})D_1^{\frac{1}{2}}
\]

\[
= D_1^{\frac{1}{2}} \mathbf{A} D_1^{\frac{1}{2}} \quad (5.4.91)
\]

which gives the system of equations
\[ 2 \left( e^{5h} + e^{-5h} \right) U_{i,j} = U_{i+1,j} + U_{i-1,j} + U_{i,j+1} + U_{i,j-1}, \quad (5.4.92) \]

and the corresponding coefficient matrix is now a symmetric positive definite matrix, see [WARLICK, 1955].

We choose the labelling of the mesh points and groups of 4 points and 9 points as before in a red-black ordering. For this case, the initial guess was chosen to be \( U_{i,j}^{(0)} = 1 \) in the interior of the region, with zero values on the boundary. The iteration was continued until the criteria

\[ \max_{i,j} \left| U_{i,j}^{(n)} \right| = \max_{i,j} \left| e_{i,j}^{(n)} \right| \leq 5 \times 10^{-6}, \]

was satisfied.

In this case as in Problem 1, the submatrices \( A_{k,k} \), \( k = 1, \ldots, \frac{m^2}{4} \), of the 4 point groups are all the same and can be easily inverted beforehand and we have

\[
A_{k,k}^{-1} = \begin{bmatrix}
\frac{\alpha^2-2}{\alpha(\alpha^2-4)} & \frac{\alpha}{\alpha(\alpha^2-4)} & \frac{\alpha}{\alpha(\alpha^2-4)} & \frac{2}{\alpha(\alpha^2-4)} \\
\frac{\alpha}{\alpha(\alpha^2-4)} & \frac{\alpha^2-2}{\alpha(\alpha^2-4)} & \frac{2}{\alpha(\alpha^2-4)} & \frac{\alpha}{\alpha(\alpha^2-4)} \\
\frac{\alpha}{\alpha(\alpha^2-4)} & \frac{2}{\alpha(\alpha^2-4)} & \frac{\alpha^2-2}{\alpha(\alpha^2-4)} & \frac{\alpha}{\alpha(\alpha^2-4)} \\
\frac{2}{\alpha(\alpha^2-4)} & \frac{\alpha}{\alpha(\alpha^2-4)} & \frac{\alpha}{\alpha(\alpha^2-4)} & \frac{\alpha^2-2}{\alpha(\alpha^2-4)} \\
\end{bmatrix}; \quad (5.4.93)\]

where \( \alpha = 2 \cdot (e^{5h} + e^{-5h}) \).
Once again we can calculate first the values of $u_{i,j}^{(n+1)}$ and $u_{i+1,j+1}^{(n+1)}$. Then using these values to calculate the $u_{i,j+1}^{(n+1)}$ and $u_{i+1,j}^{(n+1)}$ using equation (5.4.92) as described in Problem 1.

The inverse of the submatrices $A_{k,k}$, $k = 1, \ldots, m^2/9$ where the group of 9 points are employed to obtain the solution of the problem is of the form shown in (5.4.94) to reduce the number of operations per iteration. Once again we first calculate the values $u_{i,j+1}^{(n+1)}$, $u_{i+1,j}^{(n+1)}$, $u_{i+1,j+2}^{(n+1)}$ and $u_{i+2,j+1}^{(n+1)}$ by multiplying the rows 2, 4, 6 and 8 of the matrix (5.4.94) in the right hand side vector and then using these values we calculate the remaining 9 points in a similar way as described in Problem 1. The results are shown in Tables (5.4.8), (5.4.9), (5.4.10), (5.4.11) and (5.4.12) and the time information is recorded in Table (5.4.13) and the number of operations in each case is similar to Problem 1.
\[
A_{-1}^{M,M} = \frac{1}{\alpha(a^2-2)(a^2-8)}
\]

\[
\begin{bmatrix}
(a^2-5)(a^2-3)-9 & a(a^2-5) & a^2-2 & a(a^2-5) & 2(a^2-2) & 3a & a^2-2 & 3a & 6 \\
 a(a^2-5) & (a^2-5)(a^2-2)-6 & a(a^2-5) & 2(a^2-2) & a(a^2-2) & 2(a^2-2) & 3a & a^2+4 & 3a \\
 a^2-2 & a(a^2-5) & (a^2-5)(a^2-3)-9 & 3a & 2(a^2-2) & a(a^2-5) & 6 & 3a & a^2-2 \\
 a(a^2-5) & 2(a^2-2) & 3a & (a^2-2)(a^2-5)-6 & a(a^2-2) & a^2+4 & a(a^2-5) & 2(a^2-2) & 3a \\
 2(a^2-2) & a(a^2-2) & 2(a^2-2) & a(a^2-2) & (a^2-4)(a^2-2) & a(a^2-2) & 2(a^2-2) & a(a^2-2) & 2(a^2-2) \\
 3a & 2(a^2-2) & a(a^2-5) & a^2+4 & a(a^2-2) & (a^2-2)(a^2-5)-6 & 3a & 2(a^2-2) & a(a^2-5) \\
 a^2-2 & 3a & 6 & a(a^2-5) & 2(a^2-2) & 3a & (a^2-5)(a^2-3)-9 & a(a^2-5) & a^2-2 \\
 3a & a^2+4 & 3a & 2(a^2-2) & a(a^2-2) & 2(a^2-2) & a(a^2-5) & (a^2-2)(a^2-5)-6 & a(a^2-5) \\
 6 & 3a & a^2-2 & 3a & 2(a^2-2) & a(a^2-5) & a^2-2 & a(a^2-5) & (a^2-5)(a^2-3)-9 \\
\end{bmatrix}
\]

(5.4.94)
<table>
<thead>
<tr>
<th>Mesh Spacing</th>
<th>( \rho(B) )</th>
<th>( \omega_b )</th>
<th>( \omega_e )</th>
<th>( n_T )</th>
<th>( n_E )</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>0.953166</td>
<td>1.535570</td>
<td>1.545570</td>
<td>20</td>
<td>23</td>
</tr>
<tr>
<td>31</td>
<td>0.982068</td>
<td>1.682752</td>
<td>1.687752</td>
<td>32</td>
<td>36</td>
</tr>
<tr>
<td>43</td>
<td>0.990628</td>
<td>1.759650</td>
<td>1.764690</td>
<td>44</td>
<td>49</td>
</tr>
<tr>
<td>55</td>
<td>0.994258</td>
<td>1.806666</td>
<td>1.809166</td>
<td>57</td>
<td>62</td>
</tr>
<tr>
<td>61</td>
<td>0.995329</td>
<td>1.823910</td>
<td>1.828910</td>
<td>65</td>
<td>69</td>
</tr>
<tr>
<td>73</td>
<td>0.996735</td>
<td>1.850584</td>
<td>1.853084</td>
<td>75</td>
<td>81</td>
</tr>
<tr>
<td>85</td>
<td>0.997591</td>
<td>1.870251</td>
<td>1.872751</td>
<td>88</td>
<td>93</td>
</tr>
</tbody>
</table>

**TABLE (5.4.8) POINT S.O.R. METHOD. PROBLEM 2.**
<table>
<thead>
<tr>
<th>( h^{-1} )</th>
<th>( p(B) )</th>
<th>( \omega_b )</th>
<th>( \omega_e )</th>
<th>( n_T )</th>
<th>( n_E )</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>0.910523</td>
<td>1.414968</td>
<td>1.424968</td>
<td>14</td>
<td>16</td>
</tr>
<tr>
<td>31</td>
<td>0.964767</td>
<td>1.583398</td>
<td>1.590898</td>
<td>23</td>
<td>26</td>
</tr>
<tr>
<td>43</td>
<td>0.981429</td>
<td>1.678101</td>
<td>1.685601</td>
<td>31</td>
<td>35</td>
</tr>
<tr>
<td>55</td>
<td>0.988581</td>
<td>1.738088</td>
<td>1.743088</td>
<td>40</td>
<td>44</td>
</tr>
<tr>
<td>61</td>
<td>0.990701</td>
<td>1.760471</td>
<td>1.765471</td>
<td>45</td>
<td>48</td>
</tr>
<tr>
<td>73</td>
<td>0.993492</td>
<td>1.795485</td>
<td>1.800485</td>
<td>53</td>
<td>58</td>
</tr>
<tr>
<td>85</td>
<td>0.995193</td>
<td>1.821602</td>
<td>1.824102</td>
<td>62</td>
<td>68</td>
</tr>
</tbody>
</table>

**TABLE (5.4.9) 1-LINE S.O.R. METHOD. PROBLEM 2.**
<table>
<thead>
<tr>
<th>$h^{-1}$ Mesh Spacing</th>
<th>$\rho(B)$ Spectral Radius</th>
<th>$\omega_b$ Optimal Over-relaxation Parameter</th>
<th>$\omega_e$ Experimental Over-relaxation Parameter</th>
<th>$n_T$ No. of Iterations</th>
<th>$n_E$ Exp. No. of Iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>0.835713</td>
<td>1.291016</td>
<td>1.301016</td>
<td>10</td>
<td>12</td>
</tr>
<tr>
<td>31</td>
<td>0.931923</td>
<td>1.467720</td>
<td>1.477720</td>
<td>16</td>
<td>19</td>
</tr>
<tr>
<td>43</td>
<td>0.963533</td>
<td>1.577799</td>
<td>1.582799</td>
<td>22</td>
<td>26</td>
</tr>
<tr>
<td>55</td>
<td>0.977419</td>
<td>1.651105</td>
<td>1.656105</td>
<td>29</td>
<td>33</td>
</tr>
<tr>
<td>61</td>
<td>0.981572</td>
<td>1.679131</td>
<td>1.684131</td>
<td>32</td>
<td>36</td>
</tr>
<tr>
<td>73</td>
<td>0.987067</td>
<td>1.723682</td>
<td>1.728682</td>
<td>38</td>
<td>42</td>
</tr>
<tr>
<td>85</td>
<td>0.990432</td>
<td>1.757462</td>
<td>1.762462</td>
<td>44</td>
<td>48</td>
</tr>
</tbody>
</table>

TABLE (5.4.10) 2-LINE S.O.R. METHOD. PROBLEM 2.
<table>
<thead>
<tr>
<th>$h^{-1}$ Mesh Spacing</th>
<th>$\rho(B)$ Spectral Radius</th>
<th>$\omega_b$ Optimal Over-relaxation Parameter</th>
<th>$\omega_e$ Experimental Over-relaxation Parameter</th>
<th>$n_T$ No. of Iterations</th>
<th>$n_E$ Exp. No. of Iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>0.910456</td>
<td>1.414822</td>
<td>1.424822</td>
<td>14</td>
<td>16</td>
</tr>
<tr>
<td>31</td>
<td>0.964759</td>
<td>1.583361</td>
<td>1.590861</td>
<td>23</td>
<td>26</td>
</tr>
<tr>
<td>43</td>
<td>0.981428</td>
<td>1.678088</td>
<td>1.683088</td>
<td>31</td>
<td>35</td>
</tr>
<tr>
<td>55</td>
<td>0.988581</td>
<td>1.738082</td>
<td>1.743082</td>
<td>40</td>
<td>44</td>
</tr>
<tr>
<td>61</td>
<td>0.990700</td>
<td>1.760467</td>
<td>1.765467</td>
<td>45</td>
<td>48</td>
</tr>
<tr>
<td>73</td>
<td>0.993492</td>
<td>1.795483</td>
<td>1.800483</td>
<td>53</td>
<td>58</td>
</tr>
<tr>
<td>85</td>
<td>0.995193</td>
<td>1.821601</td>
<td>1.824101</td>
<td>62</td>
<td>67</td>
</tr>
</tbody>
</table>

TABLE (5.4.11) 4-POINT GROUP S.O.R. METHOD. PROBLEM 2.
<table>
<thead>
<tr>
<th>$h^{-1}$ Mesh Spacing</th>
<th>$\rho(B)$ Spectral Radius</th>
<th>$\omega_b$ Optimal Over-relaxation Parameter</th>
<th>$\omega_e$ Experimental Over-relaxation Parameter</th>
<th>$n_T$ No. of Iterations</th>
<th>$n_E$ Exp. No. of Iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>0.873118</td>
<td>1.344531</td>
<td>1.352031</td>
<td>12</td>
<td>14</td>
</tr>
<tr>
<td>31</td>
<td>0.948333</td>
<td>1.518284</td>
<td>1.528284</td>
<td>19</td>
<td>21</td>
</tr>
<tr>
<td>43</td>
<td>0.972478</td>
<td>1.622066</td>
<td>1.629566</td>
<td>26</td>
<td>29</td>
</tr>
<tr>
<td>55</td>
<td>0.982999</td>
<td>1.689744</td>
<td>1.694744</td>
<td>33</td>
<td>37</td>
</tr>
<tr>
<td>61</td>
<td>0.986136</td>
<td>1.715352</td>
<td>1.720352</td>
<td>36</td>
<td>40</td>
</tr>
<tr>
<td>73</td>
<td>0.990279</td>
<td>1.755780</td>
<td>1.760780</td>
<td>44</td>
<td>47</td>
</tr>
<tr>
<td>85</td>
<td>0.992812</td>
<td>1.786218</td>
<td>1.791218</td>
<td>51</td>
<td>55</td>
</tr>
</tbody>
</table>

TABLE (5.4.12) 9-POINT GROUP S.O.R. METHOD. PROBLEM 2.
TABLE (5.4.13) TIME IN MILL-UNITS FOR EXECUTION WITH THE BEST EXPERIMENTAL VALUE OF $\omega$. PROBLEM 2.

<table>
<thead>
<tr>
<th>Method $h^{-1}$</th>
<th>POINT S.O.R.</th>
<th>1-LINE S.O.R.</th>
<th>2-LINE S.O.R.</th>
<th>4-POINT GROUP S.O.R.</th>
<th>9-POINT GROUP S.O.R.</th>
</tr>
</thead>
<tbody>
<tr>
<td>43</td>
<td>70</td>
<td>70</td>
<td>54</td>
<td>48</td>
<td>40</td>
</tr>
<tr>
<td>55</td>
<td>146</td>
<td>145</td>
<td>113</td>
<td>100</td>
<td>85</td>
</tr>
<tr>
<td>61</td>
<td>200</td>
<td>195</td>
<td>152</td>
<td>134</td>
<td>111</td>
</tr>
<tr>
<td>73</td>
<td>337</td>
<td>338</td>
<td>255</td>
<td>234</td>
<td>193</td>
</tr>
<tr>
<td>85</td>
<td>526</td>
<td>539</td>
<td>397</td>
<td>367</td>
<td>307</td>
</tr>
</tbody>
</table>
Problem 3. The solution of the equation

\[ \frac{\partial}{\partial x} \left( e^{xy} \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left( e^{xy} \frac{\partial u}{\partial y} \right) - u = 0, \quad (5.4.95) \]

in the unit square with the boundary condition \( u = 0 \) on all boundaries.

Equation (5.4.95) was replaced by the finite difference form;

\[ (e^{xy} \left( e^{hy/2} + e^{-hy/2} + e^{hx/2} + e^{-hx/2} \right) + h^2) u_{i,j} = e^{xy} e^{hy/2} u_{i+1,j} \]

\[ + e^{xy} e^{-hy/2} u_{i-1,j} + e^{xy} e^{hx/2} u_{i,j+1} + e^{xy} e^{-hx/2} u_{i,j-1} \]

or

\[ (e^{hy/2} + e^{-hy/2} + e^{hx/2} + e^{-hx/2} + h^2 e^{-xy}) u_{i,j} = e^{hy/2} u_{i+1,j} + e^{-hy/2} u_{i-1,j} \]

\[ + e^{hx/2} u_{i,j+1} + e^{-hx/2} u_{i,j-1} \]. \quad (5.4.96) \]

Equation (5.4.97) was obtained from (5.4.96) by multiplying the system of (5.4.96), i.e. \( A U = b \), by the diagonal matrix \( D_1 \), with diagonal elements of the form \( e^{-xy} \). Again similar to Problem 2, if we choose \( T = D_1^{-1} \) then the matrix

\[ T(D_1 A)T^{-1} = D_1^{1/2} A D_1^{1/2} \]

is symmetric and positive definite. Therefore, finally we have the system of equations.
In this case, as in Problem 2, the initial guess was chosen to be \( U_{i,j}^{(0)} = 1 \) on the interior mesh points of the region, with zero values on the points of boundary and we continue the iteration process until

\[
\max_{i,j} \left| U_{i,j}^{(n)} \right| \leq 5 \times 10^{-6} ,
\]

the labelling of the mesh points were chosen as before.

It can be easily seen that the diagonal elements of the blocks of \((2 \times 2)\) or \((3 \times 3)\) mesh points, for the 4-point groups or 9-point groups respectively, are different and therefore employing the previous strategy of finding the inverse beforehand would be inefficient and uneconomic. Therefore, for the solution of the groups we use the Gaussian elimination procedure as described in Chapter 3. In order to minimise the amount of fill-ins occurring during the elimination process we first re-order the columns and rows such that the resulting submatrices in each group of 4 and 9 points becomes as follows;

\[
\begin{bmatrix}
d_{i,j+1} & 0 & -1 & -1 \\
0 & d_{i+1,j} & -1 & -1 \\
-1 & -1 & d_{i,j} & 0 \\
-1 & -1 & 0 & d_{i+1,j+1}
\end{bmatrix}, \quad i,j = 1(2)m
\]

\[ (5.4.99) \]
and

\[
\begin{bmatrix}
   d_{i,j} & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 \\
   0 & d_{i,j+2} & 0 & 0 & 0 & -1 & 0 & -1 & 0 \\
   0 & 0 & d_{i+1,j+1} & 0 & 0 & -1 & -1 & -1 & -1 \\
   0 & 0 & 0 & d_{i+2,j} & 0 & 0 & -1 & 0 & -1 \\
   0 & 0 & 0 & 0 & d_{i+2,j+2} & 0 & 0 & -1 & 0 & -1 \\
   -1 & -1 & -1 & 0 & 0 & d_{i,j+1} & 0 & 0 & 0 \\
   -1 & 0 & -1 & -1 & 0 & 0 & d_{i+1,j} & 0 & 0 \\
   0 & -1 & -1 & 0 & 0 & 0 & 0 & d_{i+1,j+2} & 0 \\
   0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & d_{i+2,j+1}
\end{bmatrix},
\]

\[i,j = 1(3)m, \quad (5.4.100)\]

where

\[d_{k,\ell} = e^{(kh^2/2)} + e^{(-kh^2/2)} + e^{(\ell h^2/2)} + e^{(-\ell h^2/2)} + \frac{h^2}{2} e^{-k\ell h^2}\]

and \(h = 1/(m+1)\). Therefore, the number of operations for the solution of the system of equations by;

1) the point S.O.R. method is

\[2m^2 \text{ multiplications} + 5m^2 \text{ additions} + m^2 \text{ divisions},\]

2) the 1-line S.O.R. method is

\[m(3m-1) \text{ multiplications} + m(6m-1) \text{ additions} + m^2 \text{ divisions},\]

3) the 2-line S.O.R. method is

\[m(8m-5) \text{ multiplications} + \frac{m}{2}(18m-7) \text{ additions} + m^2 \text{ divisions},\]
4) the group of 4-point S.O.R. method is

\[ \frac{17}{4} m^2 \text{ multiplications} + \frac{25}{4} m^2 \text{ additions} + m^2 \text{ divisions}, \]

5) the group of 9-point S.O.R. method is

\[ \frac{62}{9} m^2 \text{ multiplications} + \frac{74}{9} m^2 \text{ additions} + m^2 \text{ divisions}. \]

The numerical results for various choices of mesh regions are shown in Tables (5.4.14), (5.4.15), (5.4.16), (5.4.17) and (5.4.18) and the timing information for execution with the best experimental values of \( \omega \) (i.e. \( \omega_e \)) is recorded in Table (5.4.19).
<table>
<thead>
<tr>
<th>( h^{-1} ) Mesh Spacing</th>
<th>( \rho(B) ) Spectral Radius</th>
<th>( \omega_b ) Optimal Over-relaxation Parameter</th>
<th>( \omega_e ) Experimental Over-relaxation Parameter</th>
<th>( n_T ) No. of Iterations</th>
<th>( n_E ) Exp. No. of Iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>0.985729</td>
<td>1.711828</td>
<td>1.716828</td>
<td>36</td>
<td>38</td>
</tr>
<tr>
<td>31</td>
<td>0.994630</td>
<td>1.812416</td>
<td>1.814916</td>
<td>59</td>
<td>63</td>
</tr>
<tr>
<td>43</td>
<td>0.997207</td>
<td>1.861014</td>
<td>1.863514</td>
<td>82</td>
<td>86</td>
</tr>
<tr>
<td>55</td>
<td>0.998293</td>
<td>1.889626</td>
<td>1.892126</td>
<td>104</td>
<td>110</td>
</tr>
</tbody>
</table>

**TABLE (5.4.14)** POINT S.O.R. METHOD.  PROBLEM 3.

<table>
<thead>
<tr>
<th>( h^{-1} ) Mesh Spacing</th>
<th>( \rho(B) ) Spectral Radius</th>
<th>( \omega_b ) Optimal Over-relaxation Parameter</th>
<th>( \omega_e ) Experimental Over-relaxation Parameter</th>
<th>( n_T ) No. of Iterations</th>
<th>( n_E ) Exp. No. of Iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>0.971859</td>
<td>1.618695</td>
<td>1.628695</td>
<td>25</td>
<td>28</td>
</tr>
<tr>
<td>31</td>
<td>0.989316</td>
<td>1.745530</td>
<td>1.750530</td>
<td>42</td>
<td>45</td>
</tr>
<tr>
<td>43</td>
<td>0.994430</td>
<td>1.809305</td>
<td>1.814305</td>
<td>58</td>
<td>62</td>
</tr>
<tr>
<td>55</td>
<td>0.996591</td>
<td>1.847576</td>
<td>1.852576</td>
<td>74</td>
<td>79</td>
</tr>
</tbody>
</table>

**TABLE (5.4.15)** 1-LINE S.O.R. METHOD.  PROBLEM 3.
<table>
<thead>
<tr>
<th>$h^{-1}$ Mesh Spacing</th>
<th>$\rho(B)$ Spectral Radius</th>
<th>$\omega_b$ Optimal Over-relaxation Parameter</th>
<th>$\omega_e$ Experimental Over-relaxation Parameter</th>
<th>$n_T$ No. of Iterations</th>
<th>$n_E$ Exp. No. of Iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>0.945203</td>
<td>1.507746</td>
<td>1.517746</td>
<td>18</td>
<td>21</td>
</tr>
<tr>
<td>31</td>
<td>0.978853</td>
<td>1.660351</td>
<td>1.667851</td>
<td>29</td>
<td>32</td>
</tr>
<tr>
<td>43</td>
<td>0.988921</td>
<td>1.741488</td>
<td>1.746488</td>
<td>41</td>
<td>45</td>
</tr>
<tr>
<td>55</td>
<td>0.993205</td>
<td>1.791508</td>
<td>1.796508</td>
<td>52</td>
<td>56</td>
</tr>
<tr>
<td>61</td>
<td>0.994470</td>
<td>1.809928</td>
<td>1.814928</td>
<td>58</td>
<td>62</td>
</tr>
</tbody>
</table>

**TABLE (5.4.16) 2-LINE S.O.R. METHOD. PROBLEM 3.**

<table>
<thead>
<tr>
<th>$h^{-1}$ Mesh Spacing</th>
<th>$\rho(B)$ Spectral Radius</th>
<th>$\omega_b$ Optimal Over-relaxation Parameter</th>
<th>$\omega_e$ Experimental Over-relaxation Parameter</th>
<th>$n_T$ No. of Iterations</th>
<th>$n_E$ Exp. No. of Iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>0.971824</td>
<td>1.618508</td>
<td>1.626008</td>
<td>25</td>
<td>28</td>
</tr>
<tr>
<td>31</td>
<td>0.989313</td>
<td>1.745496</td>
<td>1.752996</td>
<td>42</td>
<td>46</td>
</tr>
<tr>
<td>43</td>
<td>0.994430</td>
<td>1.809295</td>
<td>1.814295</td>
<td>58</td>
<td>62</td>
</tr>
<tr>
<td>55</td>
<td>0.996591</td>
<td>1.847572</td>
<td>1.852572</td>
<td>74</td>
<td>79</td>
</tr>
<tr>
<td>61</td>
<td>0.997228</td>
<td>1.861483</td>
<td>1.863983</td>
<td>82</td>
<td>88</td>
</tr>
</tbody>
</table>

**TABLE (5.4.17) 4-POINT GROUP S.O.R. METHOD. PROBLEM 3.**
<table>
<thead>
<tr>
<th>$h^{-1}$ Mesh Spacing</th>
<th>$p(8)$ Spectral Radius</th>
<th>$\omega_b$ Optimal Over-relaxation Parameter</th>
<th>$\omega_e$ Experimental Over-relaxation Parameter</th>
<th>$n_T$ No. of Iterations</th>
<th>$n_E$ Exp. No. of Iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>0.958438</td>
<td>1.556056</td>
<td>1.563556</td>
<td>21</td>
<td>24</td>
</tr>
<tr>
<td>31</td>
<td>0.984076</td>
<td>1.698153</td>
<td>1.705653</td>
<td>34</td>
<td>37</td>
</tr>
<tr>
<td>43</td>
<td>0.991674</td>
<td>1.771831</td>
<td>1.776831</td>
<td>47</td>
<td>51</td>
</tr>
<tr>
<td>55</td>
<td>0.994898</td>
<td>1.816711</td>
<td>1.819211</td>
<td>60</td>
<td>65</td>
</tr>
<tr>
<td>61</td>
<td>0.995849</td>
<td>1.833141</td>
<td>1.835641</td>
<td>67</td>
<td>72</td>
</tr>
</tbody>
</table>

**TABLE (5.4.18) 9-POINT GROUP S.O.R. METHOD. PROBLEM 3.**

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>17</td>
<td>11</td>
<td>11</td>
<td>12</td>
<td>9</td>
</tr>
<tr>
<td>31</td>
<td>67</td>
<td>60</td>
<td>49</td>
<td>42</td>
<td>35</td>
</tr>
<tr>
<td>43</td>
<td>163</td>
<td>160</td>
<td>138</td>
<td>110</td>
<td>94</td>
</tr>
<tr>
<td>55</td>
<td>307</td>
<td>289</td>
<td>279</td>
<td>215</td>
<td>212</td>
</tr>
<tr>
<td>61</td>
<td>Time prohibited.</td>
<td>Time prohibited.</td>
<td>386</td>
<td>309</td>
<td>281</td>
</tr>
</tbody>
</table>

**TABLE (5.4.19) TIME IN MILL UNITS FOR EXECUTION WITH THE BEST EXPERIMENTAL VALUE OF $\omega$. PROBLEM 3.**
In this chapter, a variety of group S.O.R. methods have been investigated to obtain some impressive results on three different typical problems. During the investigation it became clear that the most efficient way by which the system of equations is partitioned to give the best result depends on the nature of the problem in hand. It can be seen from Tables (5.4.6), (5.4.7) and (5.4.13) that if the resulting blocks on the diagonal of the coefficient matrix are the same (e.g. Problem 1 and 2) then comparing the number of iterations, the amount of arithmetic operations required per iteration and the total time incurred to obtain the solution shows that the group of nine points appears to be the most efficient algorithm.

In Problem 3, where the submatrices of groups are different and the Gaussian-elimination method is employed to obtain the direct solution of the groups of 4 and 9 points respectively as well as the 1 and 2-line S.O.R. methods, then Table (5.4.19) shows that the group of nine points is again the best amongst all other groupings.

However, the most important conclusion we can derive from this exercise is that contrary to the tradition of choosing the 2-line S.O.R. method as the most efficient iterative method, the smaller sized blocks, i.e. 9-point and 4-point seem to work better under many circumstances.
CHAPTER 6

EXTERIOR PROBLEMS
6.1 INTRODUCTION.

In the previous chapters the problem obtaining the numerical solution of elliptic partial differential equation on bounded regions was fully discussed and alternative strategies for their solutions were examined. However, the numerical solution of partial differential equation on infinite region, i.e. exterior problems, presents a formidable task, even with present day computer architecture and the recent improvement in convergence rates of iterative methods which are considered to be suitable for the solution of partial differential equations.

In this chapter we concern ourselves with the solution by iterative methods of exterior problems and in particular elliptic partial differential equations on infinite regions. For this we consider the solution to the problem exterior to special geometrical shaped domains such as, circular, square, and ellipsoid.

Our model problem for this will be the exterior Dirichlet problem, for the Laplace equation, and it is formulated as follows:

Let $\Omega$ be the interior and $\partial \Omega$ be the boundary of a simply connected open bounded domain in the $xy$-plane. Let $\Omega^*$ be the exterior of the domain. If $f(x,y)$ is given and continuous on $\partial \Omega$, then the exterior Dirichlet problem for the Laplace equation is that of determining a function $u(x,y)$ on $\Omega^* \cup \partial \Omega$ which is

(a) defined and continuous on $\Omega^* \cup \partial \Omega$
(b) harmonic on $\Omega^*$
(c) identical with $f(x,y)$ on $\partial \Omega$, and
(d) bounded on $\Omega^* \cup \partial \Omega$. 


It is known that the exterior Direchlet problem has a unique solution, but in general no analytical technique is available at present for the construction of the solution [GREENSPAN, 1968].

A suitable numerical strategy for tackling the above problem is the successive peripheral block over-relaxation method (S.P.O.R.) which has been applied successfully to a variety of boundary value problems in annular and circular regions. Studies by BENSON & EVANS, [1972].

Moreover, we study the efficiency of the methods on individual mesh structures depending on the regularity or irregularity of the mesh sizes.

In Section 6.2 we shall present the peripheral ordering in general for Networks, and we shall study the properties of resulting systems of linear equations and in particular the coefficient matrix involved. Some suitable algorithms for solving the subsystems incurred using a five-point finite difference approximation are also presented in this section. In Section 6.2 we solve the model problem outside a given square and the solution of the model problem exterior to an ellipse using the finite differences in elliptical coordinates. Finally, the solution of exterior Dirichlet problem for Laplace's equation will be sought where the inside boundary is a circle using the finite differences in polar coordinates.

6.2 PERIPHERAL ORDERINGS OF POINTS.

We now consider a serial ordering of the grid points around successive peripherals of a two-dimensional area.

Thus, if a general five-point difference operator typified by Fig. 6.1.
at the point $p$ of its mesh points adjacent to $p$, mnemonically denoted by $L, T, R$ and $B$, (as in Fig. 6.2).

is applied to a second order elliptic partial differential equation involving no mixed derivative terms, the resulting finite difference equation is

$$-l_p U_L - b_p U_B + U_p - t_p U_T - r_p U_R = S_p .$$

Equation (6.2.1) holds for each internal point $P$ within the region of integration. If no mixed-derivative term exists in the second order, linear elliptic, partial differential equation and the equation itself is self adjoint, then the coefficient matrix is non-singular and positive definite.
As an example, we consider a five-point finite difference operator applied to the elliptic partial differential equation described in the previous section. The region of integration to be considered is the outside of a unit square where the boundary conditions for the boundary of the unit square is given and the boundary condition at infinity is assumed to have reached its terminal (steady state) value after \( k+1 \) (say) peripherals. It is also assumed that the mesh sizes on the peripherals and between the peripherals are taken to be equal to \( \Delta x = \Delta y = 1/N \). That is, the external region between the boundary of the unit square and the assumed boundary at infinity is divided into \( k \) peripherals with \( \Delta x = \Delta y = 1/N \) and the distances between the mesh points on the peripherals themselves have been chosen to be of equal size of \( 1/N \). This analogy is taken up in this example for the sake of simplicity in understanding the concept of dividing a region into a number of peripherals. However, in practice one would assume that the whole region bounded by the imaginary boundary at infinity to be considered as an interior problem whereby the region is divided into a net region as before but the labelling of the mesh points is carried out in such a way as to form individual peripherals.

According to the above arrangement, the number of mesh points on the boundary of the unit square (inside boundary) is equal to \( M(0) = 4*N \) and the number of mesh points on the first peripheral (next to the inside boundary) can be easily seen to be \( M(1) = M(0) + 8 \). Following the above arguments it is easy to see that the \( i^{th} \) peripheral will have \( M(i) = M(i-1) + 8 \) number of mesh points, see Fig. 6.3.
Since the five-point finite difference scheme only refers to at most 3 peripherals at any time (every point on a peripheral is expressed in terms of the corresponding points on the neighbouring peripherals) we can evidently see that the coefficient matrix $A$ must
then be block Tridiagonal in structure, as

\[
\begin{bmatrix}
  A_{1,1} & -A_{1,2} \\
  -A_{2,1} & A_{2,2} & -A_{2,3} \\
  & & & & 0 \\
  & & & & -A_{k-1,k} \\
  & & & & -A_{k,k-1} & A_{k,k}
\end{bmatrix}, \quad (6.2.2)
\]

where the diagonal submatrices \( A_{i,i} \) are square, diagonally dominant and of order \( M(i) \), \( i = 1,2,\ldots,k \). The matrices \( A_{i,i-1} \) and \( A_{i,i+1} \) are sparse, rectangular and non-negative.

The ordering of the points on the \( i \)th peripherals starts at

\[
1 + \sum_{j=1}^{i-1} M(j)
\]

and goes up to

\[
\sum_{j=1}^{i} M(j).
\]

Thus, for the \( i \)th peripheral, if we let

\[
P = 1 + \sum_{j=1}^{i-1} M(j)
\]

then the points are ordered as in Fig. 6.4.
Consequently, $A_{i,j}$ has the following form

\[
\begin{bmatrix}
1 & -t_p & -t_{p+1} & -t_{p+2} & \cdots & -t_{p+M(i)-1} \\
-1 & 1 & -\frac{t}{p+M(i)-1} & \cdots & \cdots & \cdots \\
-\frac{t}{p} & 1 & -\frac{t}{p+M(i)-1} & \cdots & \cdots & \cdots \\
\vdots & \vdots & \ddots & \ddots & \ddots & \ddots \\
-\frac{t}{p} & 1 & -\frac{t}{p+M(i)-1} & \cdots & \cdots & \cdots \\
-t & 1 & -\frac{t}{p+M(i)-1} & \cdots & \cdots & \cdots \\
-t & 1 & -\frac{t}{p+M(i)-1} & \cdots & \cdots & \cdots \\
\vdots & \vdots & \ddots & \ddots & \ddots & \ddots \\
-t & 1 & -\frac{t}{p+M(i)-1} & \cdots & \cdots & \cdots \\
-t & 1 & -\frac{t}{p+M(i)-1} & \cdots & \cdots & \cdots \\
\end{bmatrix}
\]

(6.2.3)
and the matrix $A_{i,i+1}$ of order $M(i)M(i+1)$ is of the form given in equation (6.2.4).
(Fig. 6.5)
and

\[ A_{i+1,i} = A_{i,1+i} \]  \hspace{1cm} (6.2.5)

to illustrate the structure of the coefficient matrix in the above example we consider the case where \( k = 2 \) and \( N = 2 \). It can be easily seen that the coefficient matrix in this case is of the form given in (6.2.6) where the region is shown in Fig. 6.5.

Using the peripheral ordering, as shown above, the problem now is to solve the system of linear equations

\[ AU = S \]  \hspace{1cm} (6.2.7)

by block iterative methods as described in Chapter 5.

We consider a splitting of the coefficient matrix \( A \) of the form,

\[ A = B - C = B - L - U \]  \hspace{1cm} (6.2.8)

where

\[
B = \begin{bmatrix}
A_{1,1} & 0 \\
& \ddots & \ddots \\
& 0 & A_{k,k}
\end{bmatrix}
\]  \hspace{1cm} (6.2.9)

and

\[
C = \begin{bmatrix}
0 & A_{1,2} \\
A_{2,1} & 0 & A_{2,3} & 0 \\
& \ddots & \ddots & \ddots \\
& 0 & A_{k-1,k-2} & 0 & A_{k-1,k} \\
& & 0 & A_{k,k-1} & 0
\end{bmatrix}
\]  \hspace{1cm} (6.2.10)
and \(L, U\) are respectively, the lower and upper Triangular components of \(C\). It is clear that (6.2.8) defines a regular splitting of \(A\) corresponding to a peripheral ordering of the points, but where all the points on the peripheral \(i, (i = 1, 2, ..., k)\) have been arranged to lie within the submatrix block \(A_{i, i}, i = 1, ..., k\).

The block iterative method which we consider here is the successive peripheral over-relaxation method and is defined by the equation

\[
A_{i, i} U_{i}^{(r+1)} = A_{i, i} U_{i}^{(r)} + \frac{1}{\omega} \left[ A_{i, i-1} U_{i-1}^{(r+1)} + A_{i, i+1} U_{i+1}^{(r)} + S_{i} - A_{i, i} U_{i}^{(r)} \right],
\]

\[
i = 1, 2, ..., k
\]

or

\[
U_{i}^{(r+1)} = \frac{L}{\omega p} U_{i}^{(r)} + b
\]

where

\[
L = (B - \frac{\omega}{p} L)^{-1} \left[ \frac{\omega}{p} U + (1-\frac{\omega}{p})B \right], \quad b = \frac{\omega}{p} (B - \frac{\omega}{p} L)^{-1} S.
\]

(6.2.11)

(6.2.12)

The parameter \(\omega\) is the optimum peripheral block over-relaxation factor chosen to accelerate convergence, the \(u^{(o)}\) is a chosen initial guess vector and \(r\), the iteration index. [EVANS, 1972].

The coefficient matrix \(A\) possesses property \((A^\pi)\) and is \(\pi\)-consistently ordered.

Thus if \(\lambda, \mu\) are non-zero eigenvalues of the successive block over-relaxation and peripheral block Jacobi operators respectively then,
(\lambda + \frac{\omega}{\rho} - 1)^2 = \frac{-2}{\omega^2} \mu^2 \lambda \quad , \quad (6.2.13)

where \( \frac{\omega}{\rho} \) is the over-relaxation parameters for the single peripheral ordering. [BENSON, 1968].

However, since \( \mu \) the spectral radius of the peripheral block Jacobi may not be real, and its determination difficult due to the increasing order of the diagonal submatrices \( A_{i,i} \) the use of this equation to determine the optimum over-relaxation factor may not be possible. A tentative analysis suggests an estimated value of \( \frac{\omega}{\rho} \) to be given by

\[
\frac{\omega}{\rho} = 2/\left[1 + \sin(\pi/(k+1))\right] \quad , \quad (6.2.14)
\]

for \( k \) peripherals. [EVANS, 1977].

As it was shown, using the peripheral ordering, that the diagonal blocks of the coefficient matrix (6.2.2) is a circular tridiagonal matrix of the form

\[
\begin{pmatrix}
b_1 & -c_1 & & -a_1 \\
-\alpha_2 & b_2 & -c_2 & \\
& -\alpha_3 & b_3 & -c_3 & \\
& & \ddots & \ddots & \ddots \\
& & & -\alpha_{n-1} & b_{n-1} & -c_{n-1}
\end{pmatrix} \quad (6.2.15)
\]

and since the solution of our elliptic problem by iterative method as given by (6.2.11) usually involves solving smaller linear
systems of the form (6.2.15) by direct method, we therefore, examine the existing direct methods and corresponding algorithms for solving such systems.

Algorithm 1.

The problem is to solve the system

\[ A \mathbf{u} = \mathbf{k} \]

where A is of the form (6.2.15).

Divide the first equation by \( b_1 \)

\[
\begin{bmatrix}
1 & -g_1 & -h_1 \\
-a_2 & b_2 & c_2 \\
-a_3 & b_3 & -c_3 \\
& & & \ddots \\
-a_{n-1} & b_{n-1} & c_{n-1} \\
-G & a_n & H_1 \\
\end{bmatrix}
\begin{bmatrix}
u_1 \\
u_2 \\
u_3 \\
\vdots \\
u_{n-1} \\
u_n
\end{bmatrix}
=
\begin{bmatrix}
f_1 \\
k_2 \\
k_3 \\
\vdots \\
k_{n-1} \\
F_1
\end{bmatrix}
\]  \hspace{1cm} (6.2.16)

where

\[ g_1 = c_1 / b_1 , \quad h_1 = a_1 / b_1 , \quad f_1 = k_1 / b_1 \]

\[ G_1 = c_n , \quad H_1 = b_n , \quad F_1 = k_n . \]  \hspace{1cm} (6.2.17)

Adding a multiple \( a_2 \) of the first equation of (6.2.16) to the second equation and a multiple \( G_1 \) of the first equation to the last are yields,
Dividing the second equation of (6.2.18) by diagonal elements \((b_2 - a_2 g_1)\) then gives the system

\[
\begin{bmatrix}
1 & -g_1 & & & & -h_1 \\
& 1 & -g_2 & & & -h_2 \\
& & -a_3 & b_3 & -c_3 \\
& & -a_{n-1} & b_{n-1} & -c_{n-1} \\
-g_1 & & -a_n & H_1 - G_1 h_1
\end{bmatrix}
\begin{bmatrix}
u_1 \\
u_2 \\
u_3 \\
u_{n-1} \\
u_n
\end{bmatrix}
= \begin{bmatrix}
f_1 \\
f_2 \\
k_3 \\
k_{n-1} \\
F_2
\end{bmatrix}
\tag{6.2.19}
\]

where

\[
g_2 = c_2/(b_2 - a_2 g_1), \quad h_2 = a_2 h_1/(b_2 - a_2 g_1), \quad f_2 = (k_2 + a_2 f_1)/(b_2 - a_2 g_1)
\]

\[
G_2 = g_1 G_1, \quad H_2 = H_1 - G_1 h_1, \quad F_2 = F_1 + G_1 f_1
\tag{6.2.20}
\]

and the system is reduced to one of order \((n-1)\). Consequently for

\[i = 2, 3, \ldots, (n-1)\]
\[ g_i = \frac{c_i}{b_i - a_i g_{i-1}}, \quad h_i = \frac{a_i h_{i-1}}{b_i - a_i g_{i-1}}, \quad f_i = \frac{k_i + a_i f_{i-1}}{b_i - a_i g_{i-1}} \]

(6.2.21)

\[ G_i = g_{i-1} G_{i-1}, \quad H_i = H_{i-1} - G_{i-1} h_{i-1}, \quad F_i = F_{i-1} + G_{i-1} f_{i-1} \]

and after \((n-1)\) applications of the above process, the original system is replaced by

\[
\begin{bmatrix}
1 & -g_1 & & & & \\
0 & 1 & -g_2 & & & \\
0 & 0 & 1 & -g_3 & & \\
0 & 0 & 0 & \ddots & \ddots & \\
0 & 0 & 0 & \ddots & \ddots & -g_{n-2} \\
0 & 0 & 0 & \ddots & \ddots & \ddots \\
0 & 0 & 0 & \ddots & \ddots & \ddots \\
0 & 0 & 0 & \ddots & \ddots & \ddots \\
0 & 0 & 0 & \ddots & \ddots & \ddots \\
\end{bmatrix}
\begin{bmatrix}
1 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
\end{bmatrix}
\begin{bmatrix}
f_1 \\
f_2 \\
f_3 \\
\vdots \\
f_{n-2} \\
f_{n-1} \\
f_n \\
F_{n-1} \\
\end{bmatrix}
\]

(6.2.22)

Adding \((G_{n-1} + a_n)\) times the \((n-1)\)th equation to the \(n\)th yields the final upper triangular form:
The solution can then be obtained by usual back-substitution process so that

\[ u_n = \frac{f_n}{H_n} \quad (6.2.25) \]

\[ u_i = f_i + g_i u_{i+1} + h_i u_n \quad \text{for} \quad i = (n-1), \ldots, 2, 1 \]

where the \( f \)'s, \( g \)'s and \( h \)'s are given by (6.2.17), (6.2.21) and (6.2.24).

If the matrix is symmetric and positive definite then a normalized form of algorithm 1 can be developed as shown below.
Algorithm 2.

Suppose the matrix $A$ is positive definite of the form

$$
\begin{bmatrix}
  b_1 & c_1 & & & c_n \\
  c_1 & b_2 & c_2 & & \\
  & c_2 & b_3 & c_3 & \\
  & & & \ddots & \\
  c_n & & & & c_{n-1} & b_n
\end{bmatrix}
$$

(6.2.26)

where the solution of the system $A \mathbf{u} = \mathbf{k}$ is sought.

Then it can be shown that $A$ has the unique factorization

$$
A = D T' T D
$$

where $D$ is a diagonal matrix of the form

$$
D =
\begin{bmatrix}
  d_1 \\
  & d_2 \\
  & & d_3 \\
  & & & \ddots \\
  & & & & d_n
\end{bmatrix}
$$

(6.2.27)

and $T$ is of the form

$$
T =
\begin{bmatrix}
  1 & e_1 & & & f_1 \\
  1 & e_2 & & & f_2 \\
  & 1 & e_3 & & f_3 \\
  & & \ddots & \ddots & \ddots \\
  & & & 1 & e_{n-2} & f_{n-2} \\
  & & & & 1 & (e_{n-1} + f_{n-1}) \\
  & & & & & 1
\end{bmatrix}
$$

(6.2.28)
and $T'$ denotes the transpose of $T$.

The elements of $D$ and $T$ are given by

$$d_1 = \sqrt{b_1}$$
$$d_j = \left\{ b_j - \frac{(c_j - 1)}{d_{j-1}} \right\}^{1/2}, \quad j = 2, 3, \ldots, n-1$$

$$e_j = \frac{c_j}{d_j \cdot d_{j+1}}, \quad j = 1, 2, \ldots, n-2$$

and

$$d_n = \left\{ b_n - \frac{c_n^2}{d_1} \right\} \left[ 1 + e_1^2 + e_1^2 e_2^2 + \ldots + e_1^2 e_2^2 e_3^2 \ldots e_{n-3}^2 \right]
- \left[ \frac{c_{n-1}}{d_n} + (-1)^n \frac{e_1 e_2 \ldots e_n c_n}{d_1} \right]^{1/2}$$

$$e_{n-1} = \frac{c_{n-1}}{d_{n-1} \cdot d_n}$$

$$f_1 = \frac{c_n}{d_1 \cdot d_n}$$

$$f_j = -e_{j-1} f_{j-1}, \quad j = 2, 3, \ldots, n-1$$

(6.2.29)

The system $A \mathbf{u} = \mathbf{k}$ is solved by rewriting in the form

$$D T' T D \mathbf{u} = \mathbf{k}$$

and putting

$$D \mathbf{u} = \mathbf{y}$$

$$D^{-1} \mathbf{k} = \mathbf{g}$$

so that the system
is solved for $\nu$. This system is solved directly for $\nu$ in terms of the auxiliary vector $h$ whose components are given by

$$h_1 = g_1$$

$$h_j = g_j - e_{j-1} h_{j-1}, \quad j = 2, 3, \ldots, n-1 \quad (6.2.30)$$

$$h_n = g_n - e_{n-1} h_{n-1} - \sum_{i=1}^{n-1} f_{j} h_{i}.$$  

The solution, $\nu$, is then obtained by back substitution so that

$$\nu_n = h_n \quad (6.2.31)$$

$$\nu_j = h_j - e_j \nu_{j+1} - f_j \nu_n, \quad j = n-1, \ldots, 3, 2, 1.$$  

6.3 NUMERICAL SOLUTION OF A MODEL PROBLEM.

Problem 1.

Let $\Omega$ be the unit square with vertices $(\frac{1}{2}, \frac{1}{2}), (-\frac{1}{2}, \frac{1}{2}), (-\frac{1}{2}, -\frac{1}{2})$ and let $\Omega^*$ be the exterior of $\Omega$. Let

$$u(x, y) = f(x, y) = \frac{x^2 - y^2}{(x^2 + y^2)^2}, \quad (x, y) \in \Omega. \quad (6.3.1)$$

Then, the exterior problem is to find the function $u(x, y)$ such that

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \quad \text{in} \quad \Omega^* \quad (6.3.2)$$
under boundary conditions (6.3.1) and \( u \to 0 \) as \( (x,y) \to \infty \), and so the problem is as typified in Fig. 6.6.

As mentioned before we assume the boundary condition at infinity to have reached its terminal value after \((k+l)\) peripherals. Thus, the problem now is the solution of Laplace's equation in the square

\[-(k+1)h-0.5 \leq x,y \leq (k+1)h+0.5\]

with a unit square hole removed from the centre of the square, where \( h \) indicates the regular mesh size within peripherals.

For the problem to have a reasonable accurate numerical solution the value of \( k \) should be chosen large enough to ensure the accuracy. On the other hand to reduce the truncation error the value of \( h \) (mesh size) should also be small and if this small value of \( h \) were maintained throughout all \( \Omega^* \) then this may result in far more numbers of equations than the computer could handle.

There are usually parts of the region \( \Omega^* \) in which the solution \( u \) varies slowly and parts in which it varies rapidly. It is possible
that the entire interest in the solution may be in the behaviour of \( u \) in certain parts of the region (e.g. near one corner, around the inside boundary or around the outside boundary etc.). Therefore, the usual approach is to have a small size \( h \) for those parts where the accuracy of the solution is of more interest than the other parts. For example, in our model problem we use a very fine net near the inside boundary while a coarse net is used elsewhere, with increasing values of \( h \) as we get further from the inside boundary.

The process which we employ to obtain the finite difference equations at the mesh points on interfaces between different nets is as follows:

To keep adjoining nets reasonably computible we double the mesh size \( h \) at each interface, therefore, two nets might meet at the interface WW-EE of Fig. 6.7.

![Diagram](Fig. 6.7)

The points like \( N \) are in the interior of the fine net, and corresponding finite difference equations are set up according to the formula (2.4.42) of Chapter 2 with a mesh size \( h \). Similarly,
points like SS are in the coarse net, and their finite difference equations are easily found, with a mesh size of 2h. But for points like P and E we used the following technique to define the corresponding finite difference equations.

We introduce new mesh points at the centre points of the large squares, (see Fig. 6.7), we treat E as an ordinary point of the fine net, with neighbours EE, NE, P, SE, while at point SE, we use the following formula obtained by rotating the network through 45°,

\[
A_h[u] = (\sqrt{2} h)^{-2} \left[ u(\text{EE}) + u(\text{SS}) + u(\text{SSEE}) + u(\text{P}) - 4u(\text{SE}) \right].
\]

(6.3.3)

And at point P we can use one of the following methods.

1. We treat P as an ordinary point of the coarse net, with the four neighbours EE, WW, NN, SS [FORSYTHE & WASOW, 1960], then the corresponding block matrix \(A_{i,i}\), where \(i\) indicates the interface peripheral between two different nets, is a quindiaagonal matrix in the form,

\[
\begin{bmatrix}
    c_1 & -d_1 & -e_1 & & & \\
    -b_2 & c_2 & -d_2 & -e_2 & & \\
    -a_3 & -b_3 & c_3 & -d_3 & -e_3 & \\
    & -a_4 & -b_4 & c_4 & -d_4 & -e_4 \\
    & & -a_5 & -b_5 & c_5 & -d_5 & -e_5 \\
    & & & -a_{n-2} & -b_{n-2} & c_{n-2} & -d_{n-2} & -e_{n-2} \\
    & & & & -a_{n-1} & -b_{n-1} & c_{n-1} & -d_{n-1} \\
    -e_{n-1} & & & & & -a_n & -b_n & c_n \\
    -d_n & -e_n & & & & & & \\
\end{bmatrix}
\]

(6.3.4)
Therefore, the suitable algorithm for solving the system of equation with coefficient matrix of (6.3.4) is as follows.

**Algorithm 3.**

Consider the system of equation of the form

\[ A \mathbf{u} = \mathbf{k} \]

where the matrix \( A \) is given by (6.3.4) expressing \( A \) in the form

\[ A = L U \]

where

\[
L = \begin{bmatrix}
\omega_1 \\
\beta_1 & \omega_2 \\
\alpha_1 & \beta_2 & \omega_3 \\
& \alpha_{n-4} & \beta_{n-3} & \omega_{n-2} \\
& \gamma_1 & \gamma_2 & \cdots & \gamma_{n-4} & (\gamma_{n-2} + \alpha_{n-3}) & (\beta_{n-2} + \gamma_{n-2}) & \omega_{n-1} \\
& \delta_1 & \delta_2 & \cdots & \delta_{n-4} & \delta_{n-3} & (\alpha_{n-2} + \delta_{n-2}) & \times & \omega_n
\end{bmatrix}
\]

and

\[
U = \begin{bmatrix}
1 & f_1 & g_1 & h_1 & m_1 \\
1 & f_2 & g_2 & h_2 & m_2 \\
& 1 & f_{n-4} & g_{n-4} & h_{n-4} & m_{n-4} \\
& 1 & f_{n-3} & (g_{n-3} + h_{n-3}) & m_{n-3} \\
& 1 & (f_{n-2} + h_{n-2}) & (g_{n-2} + m_{n-2}) & \gamma \\
& 1 & & & 1
\end{bmatrix}
\]
The elements $L$ and $U$ calculated by the following equations,

$$
\omega_1 = c_1, \quad \beta_1 = -b_2, \quad \alpha_1 = -a_3, \quad \gamma_1 = -e_n-1, \quad \delta_1 = -d_n
$$

$$
f_1 = -d_1/\omega_1, \quad g_1 = -e_1/\omega_1, \quad h_1 = -a_1/\omega_1, \quad m_1 = -b_1/\omega_1
$$

$$
\omega_2 = c_2 - \beta_1 f_1, \quad \beta_2 = -(b_3+\alpha_1 f_1), \quad \alpha_2 = -a_4, \quad \gamma_2 = -\gamma_1 f_1, \quad \delta_2 = -(e_n+\delta_1 f_1)
$$

$$
f_2 = \frac{-\left(d_2+\beta_1 g_1\right)}{\omega_2}, \quad g_2 = \frac{-e_2}{\omega_2}, \quad h_2 = \frac{-\beta_1 h_1}{\omega_2}, \quad m_2 = \frac{-\left(\alpha_2+\beta_1 m_1\right)}{\omega_2}
$$

(6.3.7)

Then, for $i = 3,4,\ldots,n-2$, we have

$$
\omega_i = c_i - \beta_{i-1} f_{i-1} - \alpha_{i-2} g_{i-2}
$$

$$
\beta_i = -(b_{i+1} + \alpha_{i-1} f_{i-1}), \quad \alpha_i = -a_{i+2}
$$

$$
\gamma_i = -(\gamma_{i-1} f_{i-1} + \gamma_{i-2} g_{i-2}), \quad \delta_i = -(\delta_{i-1} f_{i-1} + \delta_{i-2} g_{i-2})
$$

$$
f_i = -(d_i + \beta_{i-1} g_{i-1})/\omega_i, \quad g_i = -e_i/\omega_i
$$

$$
h_i = -(\beta_{i-1} h_{i-1} + \alpha_{i-2} h_{i-2})/\omega_i, \quad m_i = -(\beta_{i-1} m_{i-1} + \alpha_{i-2} m_{i-2})
$$

(6.3.8)

Finally,
\[
\begin{align*}
\omega_{n-1} &= c_{n-1} - (\beta_{n-2} + \gamma_{n-2})(f_{n-2} + h_{n-2}) - (\gamma_{n-3} + \alpha_{n-3})(g_{n-3} + h_{n-3}) - \sum_{k=1}^{n-4} \gamma_k h_k \\
x &= -b_n + (\alpha_{n-2} + \delta_{n-2})(f_{n-2} + h_{n-2}) + \delta_{n-3}(g_{n-3} + h_{n-3}) + \sum_{k=1}^{n-4} \delta_k h_k \\
y &= \frac{-1}{\omega_{n-1}} \left[ d_{n-1} + (\gamma_{n-2} + \beta_{n-2})(g_{n-2} + m_{n-2}) + (\gamma_{n-3} + \alpha_{n-3})m_{n-3} + \sum_{k=1}^{n-4} \gamma_k m_k \right] \\
\omega_n &= c_n - XY - (\delta_{n-2} + \alpha_{n-2})(g_{n-2} + m_{n-2}) - \sum_{k=1}^{n-3} \delta_k m_k \quad \text{(6.3.9)}
\end{align*}
\]

Therefore, the system of equation \( Au = k \) can be rewritten in the form

\[
L U u = k \quad \text{(6.3.10)}
\]

and let

\[
U u = \chi \quad \text{(6.3.11)}
\]

\[
L \chi = k \quad \text{(6.3.12)}
\]

then we have

\[
y_1 = k_1/\omega_1 \quad , \quad y_2 = (k_2 - \beta_1 y_1)/\omega_2 \quad \text{(6.3.13)}
\]

and

\[
y_i = (k_i - \beta_{i-1} y_{i-1} - \alpha_{i-2} y_{i-2})/\omega_1 \quad \text{for} \quad i = 3, 4, \ldots, n-2. \quad \text{(6.3.14)}
\]

Finally,

\[
y_{n-1} = \frac{1}{\omega_{n-1}} \left( k_{n-1} - (\beta_{n-2} + \gamma_{n-2})y_{n-2} - (\alpha_{n-3} + \gamma_{n-3})y_{n-3} - \sum_{k=1}^{n-4} \gamma_k y_k \right) \quad \text{(6.3.15)}
\]

and
\[ y_n = \frac{1}{\omega_n} (k_n - \delta_n y_{n-1} - (\alpha_{n-2} + \delta_{n-2})y_{n-2} - \sum_{k=1}^{n-3} \delta_k y_k) \]  

(6.3.16)

Thus, for the solution

\[ u_n = y_n, \quad u_{n-1} = y_{n-1} - \delta y_n \]  

(6.3.17)

and

\[ u_i = y_i - f_i u_{i+1} - g_i u_{i+2} - h_i u_{i+1} - m_i u_i \text{ for } i = n-2, n-3, \ldots, 2, 1 \]  

(6.3.18)

2. We treat \( P \) as a point of a non-square mesh net, with neighbours, \( W, N, E, SS \) and use the finite difference approximation formula

(2.4.36) along \( SS-N \) with \( S_1 = 1 \) and \( S_3 = 2 \), i.e.

\[ \frac{\partial^2 u}{\partial x^2} = \frac{2}{h^2} \left[ \frac{1}{3} u(N) + \frac{1}{6} u(SS) - \frac{1}{2} u(P) \right] \]  

(6.3.19)

Now the corresponding block matrix \( A_{i,i} \) is of the form (6.2.26), using Algorithm 2 for the solution of the system of equation

\[ A_{i,i} u_{i-1} = k_{i-1} \text{, where again } i \text{ indicates the interface peripheral of two different nets.} \]

**Numerical results.**

We solve our problem by assuming the boundary at infinity to be a square with vertices (100,100), (-100,100), (-100,-100) and (100,100). Therefore the domain \( \Omega^* \) is inside a square of side 200 with a unit square hole removed from the centre of the square.

We divide the \( \Omega^* \) into 7 subregions given by \( S_{2p} \cap S_p \), for \( p = \frac{1}{2}, 1, 2, 4, 8, 16 \), and \( S_{100} \cap S_{32} \) where \( S_x \) is the set of inside
points of a square with vertices \((\alpha, \alpha), (-\alpha, \alpha), (-\alpha, -\alpha)\) and \((\alpha, -\alpha)\). We commence by choosing the mesh spacing \(h = 0.0125\) to cover the first subdomain and then double the size of \(h\) for the next subregion. Therefore, in each subdomain we have 40 peripherals and in the last subdomain we have 85 peripherals where the 85th peripheral is assumed to be boundary at infinity. Hence, the total number of peripherals is 326 and at peripheral 41, 81, 121, 161, 201 and 241 we have irregular mesh sizes, where the two subdomains meet. At these peripherals we examine the methods 1 and 2 described in this section and the experimental results indicate that the accuracy and simplicity of method 2, i.e. using irregular finite difference formulae, is better than method 1. Another example shows that when we order the peripherals from the inside boundary to the outside boundary requires less iterations compared with procedure of solving from the outside boundary to the inside.

The resulting 171,600 linear equations in as many unknowns was reduced by symmetry to 43,226 equations.

As mentioned before, these systems of equations were solved by the block peripheral method on the "CDC 7600" with a zero initial guess at all the mesh points and with \(\omega\) over-relaxation parameter in the range 1.92085 (0.005) 1.95085. The iteration process was continued until the following criteria was satisfied:

\[
\left| u^{(n)}_{i,j} - u^{(n-1)}_{i,j} \right| \leq 5 \times 10^{-6} \quad \text{for all } i,j.
\]

The number of iterations recorded in Table (6.3.1) which shows the optimum \(\omega\) for the minimum number of iterations is \(\bar{\omega}_p - 0.05\) where \(\bar{\omega}_p\) is calculated by formula (6.2.14)
In Table (6.3.2) we have recorded the average of the maximum errors on each subdomain and this illustrates the fact that the solution is correct to at least 3 decimal places because this problem has the exact solution, namely

$$u(x,y) = \frac{x^2 - y^2}{(x^2 + y^2)^2}.$$ 

<table>
<thead>
<tr>
<th>Over-Relaxation Parameter $\omega$</th>
<th>Number of Iterations required to achieve a relative accuracy of $5 \times 10^{-6}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.95085</td>
<td>216</td>
</tr>
<tr>
<td>1.94585</td>
<td>192</td>
</tr>
<tr>
<td>1.94085</td>
<td>172</td>
</tr>
<tr>
<td>1.93585</td>
<td>165</td>
</tr>
<tr>
<td>1.93085</td>
<td>150</td>
</tr>
<tr>
<td>1.92585</td>
<td>155</td>
</tr>
<tr>
<td>1.92085</td>
<td>175</td>
</tr>
</tbody>
</table>

TABLE (6.3.1) PROBLEM 1. BLOCK PERIPHERAL S.O.R. METHOD.
### TABLE (6.3.2) PROBLEM 1. AVERAGE ERRORS IN DIFFERENT SUBDOMAIN WITH BEST \( w \).

The notation \((-\alpha)\) means \( * 10^{-\alpha} \).

<table>
<thead>
<tr>
<th>SUBDOMAIN</th>
<th>No.</th>
<th>Mesh Spacing</th>
<th>Average of the Max. Errors.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>0.0125</td>
<td>0.26491(-3)</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>0.025</td>
<td>0.8357(-4)</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>0.05</td>
<td>0.18432(-4)</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>0.1</td>
<td>0.67268(-5)</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>0.2</td>
<td>0.36358(-5)</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>0.4</td>
<td>0.60180(-5)</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>0.8</td>
<td>0.44960(-4)</td>
</tr>
</tbody>
</table>

Problem 2.

The numerical solution of the exterior Dirichlet problem for the Laplace equation, where the value of \( \phi \) is known on \( \partial \Omega \), the boundary of a given ellipse and is zero at infinity, i.e.

\[
\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = 0 \quad \text{in} \quad \Omega^* \tag{6.3.20}
\]
with
\[ \phi = f(x, y), \quad \text{on } \partial \Omega \] (6.3.21)
\[ \phi = 0, \quad \text{at infinity}. \] (6.3.22)

For convenience we use elliptical coordinates which are defined as follows: each point in the uv-plane is assigned a pair of numbers such as \((u, v)\), where \(u > 0\) and \(0 \leq v \leq 2\pi\). These numbers show the location of the points in uv-plane.

\( u = \text{constant} \) and \( v = \text{constant} \) are a confocal ellipse and hyperbola respectively, therefore the points in the uv-plane are the points of intersection of confocal ellipses and hyperbolas, (see Fig. 6.8). So the point \( p(\frac{3}{2}, \frac{\pi}{6}) \) is the intersection of ellipse \( u = \frac{3}{2} \) and hyperbola \( v = \frac{\pi}{6} \), and the point \( p' \) in (Fig. 6.8) has coordinates \( (\frac{3}{2}, \frac{11\pi}{6}) \), the coordinates of point \( A \) is \((0, 0)\) while the point \( A' \) is assigned \((0, \pi)\).
It can be shown that the relation of Cartesian coordinates and Elliptical coordinates are as follows;

\[ x = a \cosh u \cos v, \quad y = a \sinh u \sin v \quad (6.3.23) \]

So, \( u = 0 \) is represented by a straight line \( x = t, \ y = 0 \) where \(-a \leq t \leq a\),

\( v = 0 \) is the portion of the \( x \)-axis defined by \( x \geq a, \ y = 0 \),

\( u = 1 \) represents an ellipse with the Cartesian equation

\[ \frac{x^2}{a^2 \cosh(1)} + \frac{y^2}{a^2 \sinh(1)} = 1 \]

and \( v = \frac{\pi}{4} \) is the hyperbola with the Cartesian equation

\[ \frac{x^2}{a^2/2} - \frac{y^2}{a^2/2} = 1 \]

From equations (6.3.23) it can be easily seen that the equation (6.3.20) can be replaced in the \( uv \)-plane by

\[ \frac{1}{a^2 (\sinh u + \sin v)} \left( \frac{\partial^2 \phi}{\partial u^2} + \frac{\partial^2 \phi}{\partial v^2} \right) = 0 \quad (6.3.24) \]

Therefore, the problem (6.3.20) with boundary conditions (6.3.21) and (6.3.22) can be replaced by

\[ \frac{\partial^2 \phi}{\partial u^2} + \frac{\partial^2 \phi}{\partial v^2} = 0 \text{ in } \Omega^* \quad (6.3.25) \]

\[ \phi = F(u,v), \text{ on } u = \alpha \quad (6.3.26) \]

\[ \phi = 0, \text{ on } u = \beta \quad (6.3.27) \]
where \( u = \beta \) represents the assumed boundary at infinity and \( \Omega^* \) is the interior of the elliptic ring \( \alpha \leq u \leq \beta \).

The mesh points are taken to be the intersection of confocal ellipses \( u = \alpha + i\delta u \) for \( i = 1, 2, \ldots, N-1 \), where \( \delta u = \frac{2 - \alpha}{N} \) and hyperbolas \( v = j\delta v \), \( j = 0, 1, \ldots, M-1 \), where \( \delta v = \frac{2\pi}{M} \). Therefore, equation (6.3.25) at the point \((i, j)\) for the unknown \( \phi(i\delta u, j\delta v) \), i.e. \( \phi_{i,j} \), can be approximated by

\[
\frac{\phi_{i+1,j} - 2\phi_{i,j} + \phi_{i-1,j}}{(\delta u)^2} + \frac{\phi_{i,j+1} - 2\phi_{i,j} + \phi_{i,j-1}}{(\delta v)^2} = 0 \quad (6.3.28)
\]

which on rearrangement yields

\[
\phi_{i,j} = \theta_u \phi_{i+1,j} + \theta_u \phi_{i-1,j} + \theta_v \phi_{i,j+1} + \theta_v \phi_{i,j-1} \quad (6.3.29)
\]

for \( i = 1, 2, \ldots, N-1 \) and \( j = 0, 1, 2, \ldots, M-1 \),

where

\[
\theta_u = \frac{(\delta v)^2}{2[(\delta u)^2 + (\delta v)^2]} \quad \text{and} \quad \theta_v = \frac{(\delta u)^2}{2[(\delta u)^2 + (\delta v)^2]} \quad (6.3.30)
\]

Hence, we have the usual five-point molecule.

(Fig. 6.9)
We now consider a peripheral ordering of the unknown mesh points along successive ellipses, shown in (Fig. 6.10), for the case \( N = 3 \) and \( M = 8 \). Then the coefficient matrix \( A \) is of the form (6.3.31).
The application of (6.3.8) at each interior mesh point in turn, in the order already specified yields a system of linear, simultaneous equations in the \((N-1)\times M\) unknowns \(\phi_1, \phi_2, \ldots, \phi_{(N-1)\times M}\), expressible in the form

\[
\mathbf{A} \vec{\phi} = \mathbf{s}
\]  

(6.3.32)

where \(\mathbf{A}\) is a square matrix of order \((N-1)\times M\).
If A is partitioned so that its diagonal submatrices consist of mesh points on successive ellipses \( i = 1, 2, \ldots, N-1 \), then A can be written in the form:

\[
A = \begin{bmatrix}
D & -E & 0 \\
-E & D & -E \\
0 & -E & D \\
\end{bmatrix}
\]

where \( D \) and \( E \) are \((M \times M)\) matrices of the form

\[
D = \begin{bmatrix}
1 & -\theta_v & -\theta_v \\
-\theta_v & 1 & -\theta_v \\
-\theta_v & -\theta_v & 1 \\
\end{bmatrix}, \quad E = \begin{bmatrix}
\theta_u & \theta_u \\
0 & \theta_u \\
\theta_u & 0 \\
\end{bmatrix}
\]

and where \( \theta_v \) and \( \theta_u \) defined by (6.3.30).

Since the matrix A is a block Tridiagonal matrix and therefore possesses property \( (A^m) \) and can be shown to be block consistently ordered, so satisfying the requirement of the theory of block over-relaxation.
Problem 3.

The numerical solution of the exterior boundary value problem for Laplace's equation where \( \Omega \) is the interior of a circle with radius \( r = a \) and the value of \( u \) is given at the boundary of the circle and is known when \( r \) tends to infinity. As mentioned earlier, if we assume the boundary at infinity to be a circle with radius \( r = b \) then we have the following interior boundary values problem;

\[
\frac{\partial^2 u}{\partial r^2} + \frac{1}{r} \frac{\partial u}{\partial r} + \frac{1}{r^2} \frac{\partial^2 u}{\partial \theta^2} = 0 \quad \text{in } \Omega^* \tag{6.3.35}
\]

and

\[
u(a,\theta) = f_1(a,\theta) \quad \text{on } \partial \Omega_1 \tag{6.3.36}
\]

\[
u(b,\theta) = f_2(b,\theta) \quad \text{on } \partial \Omega_2 \tag{6.3.37}
\]

where \( \Omega^* \) is the interior of the ring \( a \leq r \leq b \) and where \( \partial \Omega_1 \) and \( \partial \Omega_2 \) are the boundaries of the inside and outside of the ring respectively. The mesh points in the \( r-\theta \) plane are the points of intersection of the circles \( r = i\delta r \) for \( i = R, R+1, \ldots, N+1 \) where;

\[
R = \frac{a}{\delta r}
\]

and

\[(N+1)\delta r = b\]

and the straight lines \( \theta = j\delta \theta, \ j = 0,1, \ldots, M-1 \) where

\[
\delta \theta = \frac{2\pi}{M}.
\]

Equation (6.3.35) at the point \((i,j)\) for the unknown \( u(i\delta r, j\delta \theta) \) i.e. \( u_{i,j} \) may then be approximated by
\[
\begin{align*}
\frac{U_{i+1,j} - 2U_{i,j} + U_{i-1,j}}{(\delta r)^2} + \frac{1}{i \delta r} \left( \frac{U_{i+1,j} - U_{i-1,j}}{2 \delta r} \right) \\
+ \frac{1}{(i \delta r)^2} \frac{U_{i,j+1} - 2U_{i,j} + U_{i,j-1}}{(\delta \theta)^2} & = 0
\end{align*}
\]

(6.3.38)

which on re-arrangement yields;

\[
2 \left[ 1 + \frac{1}{i^2 \delta \theta^2} \right] U_{i,j} = \left( 1 - \frac{1}{2i} \right) U_{i-1,j} + \left( 1 + \frac{1}{2i} \right) U_{i+1,j} + \frac{1}{i^2 \delta \theta^2} U_{i,j-1} + \frac{1}{i^2 \delta \theta^2} U_{i,j+1}
\]

(6.3.39)

for \( i = R+1, R+2, \ldots, N-1 \) and \( j = 0, 1, 2, \ldots, N-1 \).

The 'stencil' to be applied at each unknown mesh point \((i \delta r, j \delta \theta)\) is therefore as given in (Fig. 6.11).
Consider a peripheral ordering of the unknown mesh points along successive circumferences, as shown in (Fig. 6.12) for the case \( M = 8 \), \( R = 3 \) and \( N = 6 \).

(Fig. 6.12)

Let \( Q = R + 1 \) and as before \( N = \frac{b}{\partial r} - 1 \), so that \( K = N - Q + 1 \) is the number of points on each radius for which a solution is required.

The application of equation (6.3.39) of each interior mesh point in turn, in the order already specified, yields a system of linear, simultaneous equations in the \( MK \) unknown \( U_1, U_2, \ldots, U_{MK} \), which we can express in the following form

\[
A \underline{U} = \underline{b},
\]

where \( A \) is a square matrix of order \( MK \), and \( \underline{U} \) is the column
vector
\[
\begin{pmatrix}
U_Q,1^\top & U_Q,2^\top & \cdots & U_Q,M^\top & U_{Q+1},1^\top & U_{Q+1},2^\top & \cdots & U_{Q+1},M^\top & \cdots & U_N,1^\top & U_N,2^\top & \cdots & U_N,M^\top
\end{pmatrix}^\top
\]

where \( U_{i,j} = U(i\delta r, j\delta \theta) \).

If \( \mathbf{A} \) is partitioned so that its diagonal submatrices consist of mesh points on successive circumferences \( i = Q, Q+1, \ldots, N \) then \( \mathbf{A} \) can be written in the form

\[
\mathbf{A} =
\begin{bmatrix}
A_Q & -\left(1 + \frac{1}{2Q}\right)I & 0 & & & & & \\
-\left(1 - \frac{1}{2(Q+1)}\right)A_Q & A_{Q+1} & -\left(1 + \frac{1}{2(Q+1)}\right)I & & & & & \\
0 & -\left(1 - \frac{1}{2(N-1)}\right)I & A_{N-1} & -\left(1 + \frac{1}{2(N-1)}\right)I & & & & \\
& & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \end{bmatrix}
\]

(6.3.40)

where the \( A_i, \ i = Q, Q+1, \ldots, N \) are \( M\times M \) matrices of the form

\[
A_i =
\begin{bmatrix}
2(1+\alpha_i) & -\alpha_i & & & & & -\alpha_i \\
-\alpha_i & 2(1+\alpha_i) & -\alpha_i & & & & & & \\
& & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \end{bmatrix}
\]

(6.3.41)
where

\[ \alpha_i = \frac{1}{1 + \theta^2} \quad (6.3.42) \]

It can be shown that the matrix A possesses property \( (A^n) \) as well as being \( \pi \)-consistently ordered. Therefore, the theory of block over-relaxation method is applicable for this case.

**Numerical Results.**

We solve the exterior problem (6.3.35) with boundary conditions (6.3.36) and (6.3.37) of the form

\[ f_1(1, \theta) = \frac{1}{4} + \cos \theta \text{ and as } b \to \infty, f_2(b, \theta) \to 0.5. \]

The strategy chosen to solve this problem uses the similar scheme as given for Problem 1; i.e. we divided the region \( \Omega^* \) into \( k \) subdomains, such that the subdomain \( \ell \) is the set of all mesh points \( S_{2^\ell} \cap S_k \), \( \ell = 1, 2, 4, \ldots, 2^{k-1} \) where \( S_\alpha \) is the set of interior points of a circle with radius \( \alpha \).

In this problem, we commence by choosing the mesh spacing \( \delta r = 0.025 \) to cover the first subdomain by 40 circles and then double the size of \( \delta r \) for the next subregion, and in each subdomain we take 40 circles until the last subregion is chosen with a mesh spacing of size \( \delta r = 0.8 \) with a total number of 85 circles to cover the final subdomain. Therefore, the total number of peripherals is 286 and at peripheral 41, 81, 121, 161, 201 and 241 where the two subdomains meet we use irregular finite difference formulae along \( r \) to find the corresponding finite difference equations. Hence, the assumed boundary at infinity is a circle.
of radius 100. Finally, we choose the constant value of $\delta \theta$ equal to $\pi/32$ in all regions.

The resulting set of 18112 linear equations is partitioned into the form (6.3.40) where each block $A_Q$ has the form of (6.3.41) which is a square matrix of order (64x64) respectively. We solve these block systems using the algorithm 1 on the "CDC 7600" at U.M.R.C.C. with a zero initial guess at all interior mesh points and with the over-relaxation parameter $\omega$ chosen to be in the range $0.01(0.0025) \omega + 0.005$, where the $\omega$ is calculated by formula (6.2.14) and the iteration process being continued until the following criteria was satisfied for all points,

$$|U_{i,j}^{(n)} - U_{i,j}^{(n-1)}| \leq 5 \times 10^{-6}$$

The number of iterations are recorded in Table (6.3.3) which shows the excellent agreement with $\overline{\omega} = 1.97819$, while in Table (6.3.4) we have recorded the average of the maximum errors on each subdomain.
<table>
<thead>
<tr>
<th>Over-Relaxation Parameter ( \omega )</th>
<th>Number of Iterations required to achieve a relative accuracy of ( 5 \times 10^{-6} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.98319</td>
<td>731</td>
</tr>
<tr>
<td>1.98069</td>
<td>696</td>
</tr>
<tr>
<td>1.97819</td>
<td>611</td>
</tr>
<tr>
<td>1.97569</td>
<td>587</td>
</tr>
<tr>
<td>1.97319</td>
<td>658</td>
</tr>
<tr>
<td>1.96819</td>
<td>800</td>
</tr>
</tbody>
</table>

**TABLE (6.3.3) PROBLEM 3. BLOCK PERIPHERAL S.O.R. METHOD.**

<table>
<thead>
<tr>
<th>SUBDOMAIN</th>
<th>Average of the Max. Errors.</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.</td>
<td>( \delta r ) Mesh Spacing</td>
</tr>
<tr>
<td>1</td>
<td>0.025</td>
</tr>
<tr>
<td>2</td>
<td>0.05</td>
</tr>
<tr>
<td>3</td>
<td>0.1</td>
</tr>
<tr>
<td>4</td>
<td>0.2</td>
</tr>
<tr>
<td>5</td>
<td>0.4</td>
</tr>
<tr>
<td>6</td>
<td>0.8</td>
</tr>
</tbody>
</table>

**TABLE (6.3.4) PROBLEM 3. AVERAGE ERRORS IN DIFFERENT SUBDOMAIN WITH BEST \( \omega \).**
CHAPTER 7

Conclusions and Further Work.
The main theme of this thesis is the numerical solution of elliptic partial differential equations by iterative methods. The work presents a collection of new ideas and strategies which when applied to boundary value problems produces efficiency and simplicity for deriving the solution to certain practical problems in engineering and science.

In Chapter 4, the subject of parallel computations was initiated and investigated to design parallel algorithms suitable for existing concurrent architectures in order to improve computational efficiency and produce a speed up over and above that of the serial computer. The results of the investigation is the production of a new splitting of the coefficient matrix for a linear system of equations. The algorithms which are called the Quadrant Interlocking Iterative methods are applicable to systems of equations arising from a five-point difference approximation to the solution of partial differential equations of elliptic type. All the algorithms presented were shown to be convergent from any arbitrary initial vector $x^{(0)}$. Moreover, it was shown how these new algorithms can be implemented on two parallel architectures (i.e. MIMD systems and SIMD systems) together with the number of time steps and processors required for each individual algorithm. In particular, the implementation of these methods on the Neptune system at Loughborough which is a multi-processor of the type MIMD with 4 processors was considered. Further work in this category can be carried out for the performance analysis of these parallel algorithms on specific existing systems such as the I.C.L. DAP which is of SIMD type and the CRAY-1
computer which is a vector machine. It is only after these analyses have been carried out can the presented algorithms be fully compared competitively with other existing procedures.

In Chapter 5, we turned our attention to a group of algorithms denoted as group iterative methods. A novel grouping of the equations of a linear system was adopted to solve the Laplace equation and the generalised Dirichlet problem with the successive over-relaxation group iterative method. The numerical experiments carried on for these problems indicate the fact that smaller size groups such as 9 or 4 points groups may require less storage and computational costs when compared with the usual grouping of equations such as 1-line or 2-line. Further work is required to evaluate a theoretical value of the spectral radius of the iteration matrix corresponding to the 4 and 9 point groups and hence determine the rate of convergence theoretically.

Finally, in Chapter 6 we applied the peripheral ordering of the points suggested by BENSON [1968] for the interior problems to the solution of exterior Dirichlet boundary value problem. Perhaps one of the most important advantages shown for this type of ordering of points is the flexibility of the procedure to adopt irregular mesh sizes which is a vital point when solving numerically a problem in infinite domains.

* Recently PARTER [1971] has shown that the spectral radius $\rho$ of the $k$ line and $(kkk)$ block Jacobi iterative methods can be evaluated from the formulae $\rho(kk) \approx 1 - K \pi h^2$ and $\rho(kk) \approx 1 - \frac{K}{2} \pi^2 h^2$ respectively. Hence it can be seen that the spectral radius of the 4-point group method is $\rho(3B) \approx 1 - \frac{3}{2} \pi^2 h^2$ and the spectral radius of the 9-point group method is $\rho(3B) \approx 1 - \frac{3}{2} \pi^2 h^2$. 
REFERENCES


(26) KUCK, D.J. [1973]. Multioperation machine computational. in TRAUB (44).


\frac{\partial^2 u}{\partial x^2} + \frac{k}{p} \frac{\partial u}{\partial \rho} + \frac{\partial^2 u}{\partial \rho^2} = 0.
\]
*M.A. thesis, Univ. of Maryland, College Park, Maryland.*


APPENDIX
APPENDIX A

THE FORTRAN PROGRAM ON

THE SOLUTION OF LAPLACE'S EQUATION BY

9-POINT GROUP ITERATIVE METHOD.

THE RED-BLACK ORDERING IS USED TO
ORDER THE GROUPS OF 9-POINT.

M - THE NUMBER OF MESH POINTS ON EACH ROW
AND COLUMN

MARK=0 DENOTES THE EVALUATION OF GROUPS
Labeled as RED.

MARK=1 DENOTES THE EVALUATION OF GROUPS
Labeled as BLACK.

IFLAG=1 DENOTES THAT THE ITERATIVE PROCESS
HAS NOT REACHED THE REQUIRED
RELATIVE ACCURACY;

IFLAG=0 DENOTES THAT THE ITERATIVE PROCESS
HAS CONVERGED TO THE REQUIRED RELATIVE
ACCURACY.

U - THE OVER-RELAXATION PARAMETER GIVING
BY USER.

DIMENSION U(92,92)

N=91
M=N+1
EPS=0.000005
W=1.889781
DO 10 I=1,N
U(I,1)=0.
U(1,2)=0.
U(I,N)=0.
U(M+1)=0.
10 CONTINUE
DO 15 I=2,N
DO 15 J=2,N
U(I,J)=1.
15 CONTINUE
A=37./112
A1=11./112
A2=7./112
A3=3./112
A4=5./112
B=0.25
ITER=1
20 IFLAG=0
MARK=0
IK=2
IL=3
25 DO 35 I=2,N,3
K = I/2
J K = I K
IF (I .NE. 2*K) JK = I L
DO 30 J = JK, N, 6

C THE EVALUATION OF RIGHT HAND SIDE VECTOR.

R1 = U(I-1, J) + U(I, J-1)
R2 = U(I-1, J+1)
R3 = U(I-1, J+2) + U(I, J+3)
R4 = U(I+1, J-1)
R5 = 0.
R6 = U(I+1, J+3)
R7 = U(I+2, J-1) + U(I+3, J)
R8 = U(I+3, J+1)
R9 = U(I+3, J+2) + U(I+2, J+3)

C THE GAUSS-SEIDEL SOLUTION OF 4 EQUATIONS
U(I, J+1), U(I+1, J); U(I+1, J+2) AND U(I+2, J+1).

Q1 = R1 + R3
Q2 = R7 + R9
Q3 = R1 + R7
Q4 = R3 + R9
Q5 = R4 + R6
Q6 = R2 + R8
D1 = A1 + A2 * Q1 + A2 * Q5 + A3 + A4 + R8
D2 = A1 + A2 * Q3 + A2 * Q6 + A3 + A4 + R6
D4 = A1 + A2 * Q2 + A2 * Q5 + A1 + A4 + R2

C THE SOLUTION OF REMAINING EQUATIONS OF A
C BLOCK OF 9-POINT.

S1 = D1 + D2
S2 = D1 + D3
S3 = D2 + D4
S4 = D3 + D4
S5 = B*(R1+S1)
S6 = B*(R3+S2)
S7 = B*(S1+S4)
S8 = B*(R7+S3)
S9 = B*(R9+S4)

C THE EVALUATION OF SOLUTION VECTOR BY OVER-
C RELAXATION FORMULA.

D = U(I, J+1) + W*(D1 - U(I, J+1))
IF (ABS(U(I, J+1) - D) .GT. EPS) IFLAG = 1
U(I, J+1) = D
D = U(I+1, J) + W*(D2 - U(I+1, J))
IF (ABS(U(I+1, J) - D) .GT. EPS) IFLAG = 1
U(I+1, J) = D
D = U(I+1, J+2) + W*(D3 - U(I+1, J+2))
IF (ABS(U(I+1, J+2) - D) .GT. EPS) IFLAG = 1
U(I+1, J+2) = D
D = U(I+2, J+1) + W*(D4 - U(I+2, J+1))
IF (ABS(U(I+2, J+1) - D) .GT. EPS) IFLAG = 1
U(I+2, J+1) = D
D = U(I, J) + W*(D5 - U(I, J))
IF (ABS(U(I, J) - D) .GT. EPS) IFLAG = 1
U(I, J) = D
D = U(I, J+2) + W*(D6 - U(I, J+2))
IF(ABS(U(I,J+2)=D),GT,EPS) IFLAG=1
U(I,J+2)=D
D=U(I+1,J+1)+W*(D7=U(I+1,J+1))
IF(ABS(U(I+1,J+1)=D),GT,EPS) IFLAG=1
U(I+1,J+1)=D
D=U(I+2,J)+W*(D8=U(I+2,J))
IF(ABS(U(I+2,J)=D),GT,EPS) IFLAG=1
U(I+2,J)=D
D=U(I+2,J+2)+W*(D9=U(I+2,J+2))
IF(ABS(U(I+2,J+2)=D),GT,EPS) IFLAG=1
U(I+2,J+2)=D
30 CONTINUE
35 CONTINUE
MARK=MARK+1
IF(MARK,GT,1) GO TO 40
IK=5
IL=2
GO TO 25
40 IF(IFLAG.EQ.0) GO TO 45
ITER=ITER+1
GO TO 20
45 WRITE(2,50) ITER
50 FORMAT(1H20X,'THE NUMBER OF ITERATION=',I8)
STOP
END
FINISH
APPENDIX B

THE FORTRAN PROGRAM ON

THE NUMERICAL SOLUTION OF THE EXTERIOR

BOUNDARY VALUE PROBLEM FOR LAPLACE

EQUATION WHERE THE DOMAIN OF INTEGRATION

IS OUTSIDE A CIRCLE WITH RADIUS R=1.

THE SOLUTION SCHEME IS THE BLOCK PERIPHERAL

S.O.R. ITERATIVE METHOD.

EACH BLOCK OF SYSTEM OF EQUATIONS IS SOLVED

BY SUBROUTINE 'SOLVE'.

NS - DENOTES THE NUMBER OF SUBDOMAIN.

L - DENOTES THE PERIPHERAL INTERSECTION OF
TWO SUBDOMAINS.

M - DENOTES THE NUMBER OF MESH POINTS ON
EACH CIRCUMFERENCE PERIPHERAL.

H - DENOTES THE MESH SPACING ALONG R IN
SUBDOMAIN 1.

WP - DENOTES THE THEORETICAL OPTIMUM VALUE
OF OVER-RELAXATION PARAMETER.

W - DENOTES THE BEST EXPERIMENTAL VALUE
OF OVER-RELAXATION PARAMETER.

IFLAG=1 INDICATED THAT THE ITERATIVE PROCESS
HAS NOT REACHED TO THE REQUIRED
RELATIVE ACCURACY.

IFLAG=0 INDICATED THAT THE ITERATIVE PROCESS
HAS CONVERGED TO THE REQUIRED RELATIVE
ACCURACY.

TITA - DENOTES THE MESH SIZE ON CIRCUMFERENCE
PERIPHERAL.
DIMENSION U(64,286), L(7), A(64), R(64)
NS=6
M=64
H=0.025
EPS=0.000005
PI=4.*ATAN(1.)
TITA=2.*PI/M
K=40
L(1)=1
DO 5 I=2,49
L(I)=L(I-1)+40
5 CONTINUE
L(7)=28A
WP=2./((1.+SIN(PI/(L(7)-1))))
W=W*0.0025
N=L(7)-1
DO 10 J=1,M
U(J,1)=0.5*COS(J*TITA)
U(J,L(7))=0.5
10 CONTINUE
DO 15 J=2,N
DO 15 J=1,M
U(J,1)=0.
15 CONTINUE
ITER=1
IFLAG=0
DO 25 IEX=1,NS
J=L(IEX)+1
I=1
DO 25 I=I,J,1
OMEGA=1./(I-L(IEX)+K)*(I-L(IEX)+K)*TITA*TITA
ALFA=1.+1./(2.*(I-L(IEX)+K))
BITA=1.-1./(2.*(I-L(IEX)+K))
25 CONTINUE
CALL SOLVE(OMEGA,R,M,A)
DO 30 J=1,M
R(J)=W*(J)*(1.-W)U(J,1)
IF(IS(0(J)-U(J,1)),GT,EPS) IFLAG=1
U(J,1)=p(J)
30 CONTINUE
35 CONTINUE
IF(IEX.EQ.NS) GO TO 50
HI=L(IEX)+1-L(IEX)+K
OMEGA=1./(HI*HI*TITA*TITA)
ALFA=1./3+1./(6+HI)
BITA=2./3-2./(3+HI)
DO 40 J=1,M
R(J)=BITA*U(J,L(IEX)+1)+ALFA*U(J,L(IEX)+1)*1)
A(J)=1.-1./(2+HI)+2*OMEGA
40 CONTINUE
CALL SOLVE(OMEGA,R,M,A)
DO 45 J=1,M
R(J) = U*R(J) + (1. - W)*U(J, L(IEX + 1))
IF(ABS(R(J) - U(J, L(IEX + 1)))/GT.EPS) IFLAG=1
U(J, L(IEX + 1)) = R(J)

CONTINUE
50 CONTINUE
IF(FLAG.EQ.0) GO TO 55
ITER = ITER + 1
GO TO 20
55 WRITE(2,75) ITER
RA = 1.
DO 70 IEX = 1, NS
IJ = L(IEX) + 1
IF(IEX, EQ, 1) IJ = 1
II = L(IEX + 1)
S = 0.
DO 65 I = IJ, II
ERRMAX = 0.
DO 60 J = 1, M
ERROR = ARS(U(J, 1) - (0.5 + (1./RA) * COS(I*ITA))
IF(ABS.ERROR.GT.ERRMAX) ERRMAX = ERROR
60 CONTINUE
WRITE(2, 80) IERQMAX
S = S + ERRMAX
RA = RA - H
65 CONTINUE
AVER = S / (II - IJ)
WRITE(2, 85) IEX, AVER
RA = RA + H
H = 2*H
70 CONTINUE
75 FORMAT(1H '20X' 'THE NUMBER OF ITERATION=', 18/)
80 FORMAT(1H 'MAX. OF ERROR ON PERIPHERAL=', 14, 1 = ', E12.5/)
85 FORMAT(1H 'THE AVERAGE ERROR ON SUBDOMAIN=', 13, 1 = ', E12.5/)
STOP
END

SUBROUTINE SOLVE(OMEGA, R, M, A)

**************************************************
* THIS SUBROUTINE EVALUATED THE SOLUTION VECTOR *
* OF EACH BLOCK OF EQUATIONS USING THE ALGORITHM *
* ONE OF CHAPTER 6. *
*
* THE SOLUTION WAS STORED ON RIGHT HAND SIDE *
* VECTOR. *
**************************************************
DIMENSION A(64), B(64), C(64), D(64), E(64), R(64)
DO 10 J = 1, M
B(J) = OMEGA
C(J) = OMEGA
10 CONTINUE
E (1) = -OMEGA
D (1) = OMEGA
MM = M - 3
DO 15 K = 1, MM
R (K + 1) = R (K + 1) / A (K)
A (K + 1) = A (K + 1) - R (K + 1) * C (K)
D (K + 1) = -R (K + 1) * D (K)
E (K) = E (K) / A (K)
E (K + 1) = -C (K) * E (K)
A (M) = A (M) - E (K) * D (K)
15 CONTINUE
B (M - 1) = B (M - 1) / A (M - 2)
A (M - 1) = A (M - 1) - B (M - 1) * C (M - 2)
C (M - 1) = C (M - 1) - B (M - 1) * D (M - 2)
E (M - 2) = E (M - 2) / A (M - 2)
R (M) = B (M) * E (M - 2) * C (M - 2)
A (M) = A (M) * E (M - 2) * D (M - 2)
B (M) = B (M) / A (M - 1)
A (M) = A (M) * B (M) * C (M - 1)
DO 20 K = 1, MM
R (K + 1) = R (K + 1) - B (K + 1) * R (K)
R (M) = R (M) - E (K) * R (K)
20 CONTINUE
R (M - 1) = R (M - 1) - R (M - 1) * R (M - 2)
R (M) = R (M) * E (M - 2) * R (M - 2) * R (M - 1) * B (M)
R (M) = R (M) / A (M)
R (M - 1) = R (M - 1) * R (M) * C (M - 1) / A (M - 1)
J = M - 2
25 U (J) = (R (J) - C (J) * R (J + 1) - D (J) * R (M)) / A (J)
IF (J .EQ. 1) GO TO 30
J = J - 1
GO TO 25
30 RETURN
END
FINISH