NEW BLOCK ITERATIVE METHODS FOR THE NUMERICAL SOLUTION OF BOUNDARY VALUE PROBLEMS

BY

WADALLA S. YOUSIF, 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

August, 1984.

Supervisor: PROFESSOR D.J. EVANS, Ph.D., D.Sc.,

Department of Computer Studies

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 higher degree.

Wadalla S. Yousif.

To my Mother

and

The memory of my Father

ACKNOWLEDGEMENTS

I wish to express my profound gratitude to my supervisor, Professor D.J. Evans, for his kind advice, invaluable guidance and continued encouragement and assistance throughout the course of this thesis.

I am also grateful to Dr. J. Shanehchi for his interest and valuable comments in private communications.

My gratitude also goes to Mr. S. Bedi, Systems Manager, in the Department of Computer Studies and to Mr. G.P. Gerrard, Mr. G. Harris and Mr. D. Thomas in the Computer Centre for their help and cooperation.

I would also like to express my deep thanks to Miss J.M. Briers, for her competent typing of this thesis.

Last but by no means least, I owe a debt of love and gratitude to my mother and sisters for their seemingly unending patience and their moral and financial support.

NEW BLOCK ITERATIVE METHODS FOR THE NUMERICAL SOLUTION OF BOUNDARY VALUE PROBLEMS

ABSTRACT

The work presented in this thesis is wholly concerned with the derivation of a new group technique for the solution of boundary value problems using finite difference approximations.

The thesis commences with a general description and classification of partial differential equations and its related discretised matrix, also a description of some physical problems which involve elliptic partial differential equations are given. The numerical solution of linear elliptic partial differential equations by the method of finite differences always leads to a large number of linear algebraic equations, the determination of the set of linear equations corresponding to an elliptic partial differential equation is shown and different methods for their solution are described. An introduction to the finite element technique is also included as an alternative to the finite difference method of solution.

In Chapter 4, we present the solution of the linear system of equations by group iterative methods. New strategies are established concerning the novel approach of using a small group of points of fixed size. The 2,4,6,9,12,16 and 25 point group structure is proposed, developed and analysed theoretically and experimentally. From these results an analysis of the computational complexity of an optimum group structure can be determined and it can be deduced that such splittings can be both useful and efficient. In a similar manner, an explicit 8

point group is used to solve the elliptic partial differential equations in 3-space dimensions. The method is developed and analysed theoretically and experimentally.

In the fifth Chapter a new method is developed, i.e. the implicit block, explicit overrelaxation (IBEB) iterative method, in which we solve the 2×1 point block (or 2×2 point block) explicitly then grouping the new explicit point equations in an implicit iterative method. In this situation, two iteration parameters are used. These composite methods are analysed and some numerical experiments are carried out.

In Chapter 6 we investigate the solution of one dimensional boundary value problems using the new explicit 2,3,4,6,8 and 12 point group iterative methods. Also, some non-linear boundary value problems are solved using a similar group strategy and in particular the 2,3 and 4 point group nonlinear over-relaxation method. Furthermore in a more convincing and realistic situation we examine the linear and non-linear boundary value problems using the alternating group explicit (AGE) method. Numerical results were carried out to compare these methods with the direct approach using Picard's method.

In a similar manner to the four and nine point groups of Chapter 4, explicit four and nine point groups to the 9 point finite difference equation are presented in Chapter 7. The methods are analysed both theoretically and experimentally. Further in this chapter theoretical results for the explicit four and nine point groups to the 13 point finite difference equation of the biharmonic operator are presented.

Finally, the thesis concludes with a chapter summarizing the main results and recommendations for further work.

CONTENTS

PAGE

PAGE

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

The mathematical formulation of many important scientific and engineering problems involving rates of change with respect to two or more independent variables, leads to *Partial Differential Equations* or a set of such equations. Another class of differential equations which govern physical systems is *Ordinary Differential Equations* in which only one independent variable is present in the differential equations. The analytical solution for the majority of these equations is extremely difficult or too cumbersome to be obtained, thus numerical methods are found to be an attractive alternative, in particular, at the present time where the use of automatic digital computers are becoming widespread.

A number of approaches have been developed over the years for the treatment of partial differential equations (abbreviated as p.d.e.), the most important and widely used of these are the methods of *finite elements* and *finite differences*. The finite difference methods, which are the main subject of this thesis, stand out as being universally applicable to linear and non-linear problems.

A full description of the finite difference methods and an abbreviated description of the finite element method will be given in Sections (2.8.1) and (2.8.2).

1.2 THE CLASSIFICATION OF PARTIAL DIFFERENTIAL EQUATIONS

Let us consider the most general form of the second order p.d.e. of two independent variables, x and y, (these may both be space co-ordinates, or one may be a space coordinate and the other the time variable), with a dependent variable U, which can be expressed as,

$$A\frac{\partial^2 U}{\partial x^2} + B\frac{\partial^2 U}{\partial x\partial y} + C\frac{\partial^2 U}{\partial y^2} + D\frac{\partial U}{\partial x} + E\frac{\partial U}{\partial y} + FU + G = 0\ , \qquad (1.2.1)$$

Equation (1.2.1) is said to be:

(i) *Linear* if the coefficients A,B,C,...,G, are constants or functions of one or both independent variables x and y.

(ii) *Non-Linear* if any of the coefficients A,B,C,...,G, are functions of the dependent variable, U, or its derivatives.

(iii) *Semi-Linear* if the coefficients A,B,...,G, are functions of the independent variables x and y only.

(iv) *Quasi-Linear* if the coefficients A,B and C, are functions of $x,y,U,\partial U/\partial x$ and $\partial U/\partial y$ but not of second-order derivatives.

(v) *Homogeneous* if G=0, otherwise it is called *inhomogeneous*.

(vi) *Self-Adjoint* if the equation (1.2.1) can be replaced by

$$\frac{\partial}{\partial x}\left(A(x)\frac{\partial U}{\partial x}\right) + \frac{\partial}{\partial y}\left(C(y)\frac{\partial U}{\partial y}\right) + FU + G = 0.$$

Equation (1.2.1) can be classified into three particular types, depending on the discriminant B^2-4AC, whether it is positive, negative or zero. To reduce the equation to a form which is more readily analysed, we require a transformation of the independent variables x and y to the new independent variables ξ and η (say), where,

$$\xi = \xi(x,y) \text{ and } \quad \eta = \eta(x,y)\quad , \qquad (1.2.2)$$

by using ξ and η as intermediate variables, we can compute the necessary derivatives of U

$$\frac{\partial U}{\partial x} = \frac{\partial U}{\partial \xi}\frac{\partial \xi}{\partial x} + \frac{\partial U}{\partial \eta}\frac{\partial \eta}{\partial x} ,$$

$$\frac{\partial U}{\partial y} = \frac{\partial U}{\partial \xi}\frac{\partial \xi}{\partial y} + \frac{\partial U}{\partial \eta}\frac{\partial \eta}{\partial y} ,$$

$$\frac{\partial^2 U}{\partial x^2} = \frac{\partial^2 U}{\partial \xi^2}\left(\frac{\partial \xi}{\partial x}\right)^2 + 2\frac{\partial^2 U}{\partial \xi \partial \eta}\frac{\partial \xi}{\partial x}\frac{\partial \eta}{\partial x} + \frac{\partial^2 U}{\partial \eta^2}\left(\frac{\partial \eta}{\partial x}\right)^2 + \frac{\partial U}{\partial \xi}\frac{\partial^2 \xi}{\partial x^2} + \frac{\partial U}{\partial \eta}\frac{\partial^2 \eta}{\partial x^2} ,$$

$$\frac{\partial^2 U}{\partial y^2} = \frac{\partial^2 U}{\partial \xi^2}\left(\frac{\partial \xi}{\partial y}\right)^2 + 2\frac{\partial^2 U}{\partial \xi \partial \eta}\frac{\partial \xi}{\partial y}\frac{\partial \eta}{\partial y} + \frac{\partial^2 U}{\partial \eta^2}\left(\frac{\partial \eta}{\partial y}\right)^2 + \frac{\partial U}{\partial \xi}\frac{\partial^2 \xi}{\partial y^2} + \frac{\partial U}{\partial \eta}\frac{\partial^2 \eta}{\partial y^2} , \quad (1.2.3)$$

$$\frac{\partial^2 U}{\partial x \partial y} = \frac{\partial}{\partial x}\left(\frac{\partial U}{\partial y}\right) = \left(\frac{\partial}{\partial \xi}\frac{\partial \xi}{\partial x} + \frac{\partial}{\partial \eta}\frac{\partial \eta}{\partial x}\right)\left(\frac{\partial U}{\partial \xi}\frac{\partial \xi}{\partial y} + \frac{\partial U}{\partial \eta}\frac{\partial \eta}{\partial y}\right)$$

$$= \frac{\partial^2 U}{\partial \xi^2}\frac{\partial \xi}{\partial x}\frac{\partial \xi}{\partial y} + \frac{\partial^2 U}{\partial \xi \partial \eta}\left(\frac{\partial \xi}{\partial x}\frac{\partial \eta}{\partial y} + \frac{\partial \xi}{\partial y}\frac{\partial \eta}{\partial x}\right) + \frac{\partial^2 U}{\partial \eta^2}\frac{\partial \eta}{\partial x}\frac{\partial \eta}{\partial y} + \frac{\partial U}{\partial \xi}\frac{\partial^2 \xi}{\partial x \partial y} + \frac{\partial U}{\partial \eta}\frac{\partial^2 \eta}{\partial x \partial y} .$$

By substituting these values in (1.2.1) we obtain,

$$A\left[\frac{\partial^2 U}{\partial \xi^2}\left(\frac{\partial \xi}{\partial x}\right)^2 + 2\frac{\partial^2 U}{\partial \xi \partial \eta}\frac{\partial \xi}{\partial x}\frac{\partial \eta}{\partial x} + \frac{\partial^2 U}{\partial \eta^2}\left(\frac{\partial \eta}{\partial x}\right)^2 + \frac{\partial U}{\partial \xi}\frac{\partial^2 \xi}{\partial x^2} + \frac{\partial U}{\partial \eta}\frac{\partial^2 \eta}{\partial x^2}\right]$$
$$+B\left[\frac{\partial^2 U}{\partial \xi^2}\frac{\partial \xi}{\partial x}\frac{\partial \xi}{\partial y} + \frac{\partial^2 U}{\partial \xi \partial \eta}\left(\frac{\partial \xi}{\partial x}\frac{\partial \eta}{\partial y} + \frac{\partial \xi}{\partial y}\frac{\partial \eta}{\partial x}\right) + \frac{\partial^2 U}{\partial \eta^2}\frac{\partial \eta}{\partial x}\frac{\partial \eta}{\partial y} + \frac{\partial U}{\partial \xi}\frac{\partial^2 \xi}{\partial x \partial y} + \frac{\partial U}{\partial \eta}\frac{\partial^2 \eta}{\partial x \partial y}\right]$$
$$+C\left[\frac{\partial^2 U}{\partial \xi^2}\left(\frac{\partial \xi}{\partial y}\right)^2 + 2\frac{\partial^2 U}{\partial \xi \partial \eta}\frac{\partial \xi}{\partial y}\frac{\partial \eta}{\partial y} + \frac{\partial^2 U}{\partial \eta^2}\left(\frac{\partial \eta}{\partial y}\right)^2 + \frac{\partial U}{\partial \xi}\frac{\partial^2 \xi}{\partial y^2} + \frac{\partial U}{\partial \eta}\frac{\partial^2 \eta}{\partial y^2}\right]$$
$$+D\left[\frac{\partial U}{\partial \xi}\frac{\partial \xi}{\partial x} + \frac{\partial U}{\partial \eta}\frac{\partial \eta}{\partial x}\right] + E\left[\frac{\partial U}{\partial \xi}\frac{\partial \xi}{\partial y} + \frac{\partial U}{\partial \eta}\frac{\partial \eta}{\partial y}\right] + FU + G = 0 , \quad (1.2.4)$$

Equation (1.2.1) can be written in terms of the new variables ξ and η, that is,

$$A_1\frac{\partial^2 U}{\partial \xi^2} + B_1\frac{\partial^2 U}{\partial \xi \partial \eta} + C_1\frac{\partial^2 U}{\partial \eta^2} + D_1\frac{\partial U}{\partial \xi} + E_1\frac{\partial U}{\partial \eta} + F_1 U + G_1 = 0, \quad (1.2.5)$$

where,

$$A_1 = A\left(\frac{\partial \xi}{\partial x}\right)^2 + B\frac{\partial \xi}{\partial x}\frac{\partial \xi}{\partial y} + C\left(\frac{\partial \xi}{\partial y}\right)^2 ,$$

$$B_1 = 2A\frac{\partial \xi}{\partial x}\frac{\partial \eta}{\partial x} + B\left(\frac{\partial \xi}{\partial x}\frac{\partial \eta}{\partial y} + \frac{\partial \xi}{\partial y}\frac{\partial \eta}{\partial x}\right) + 2C\frac{\partial \xi}{\partial y}\frac{\partial \eta}{\partial y} ,$$

$$C_1 = A\left(\frac{\partial \eta}{\partial x}\right)^2 + B\frac{\partial \eta}{\partial x}\frac{\partial \eta}{\partial y} + C\left(\frac{\partial \eta}{\partial y}\right)^2 , \quad (1.2.6)$$

$$D_1 = A\frac{\partial^2 \xi}{\partial x^2} + B\frac{\partial^2 \xi}{\partial x \partial y} + C\frac{\partial^2 \xi}{\partial y^2} + D\frac{\partial \xi}{\partial x} + E\frac{\partial \xi}{\partial y} ,$$

$$E_1 = A\frac{\partial^2\eta}{\partial x^2} + B\frac{\partial^2\eta}{\partial x\partial y} + C\frac{\partial^2\eta}{\partial y^2} + D\frac{\partial\eta}{\partial x} + E\frac{\partial\eta}{\partial y} \quad ,$$

$F_1 = F$ and $G_1 = G$.

Now if we want to determine ξ and η such that equation (1.2.5) takes the simplest possible form this occurs when λ_1 and λ_2 (say) are the roots of the quadratic equation,

$$A\lambda^2 + B\lambda + C = 0 \ . \tag{1.2.7}$$

Returning to the general classification scheme, we see that *three cases* as determined by the sign of the discriminant of the quadratic form B^2-4AC are of particular interest:

(i) $B^2-4AC>0$, this implies that the roots λ_1 and λ_2 of the quadratic algebraic equation (1.2.7) are real and unequal, so A_1 and C_1 can be written as,

$$A_1 = A\left(\frac{\partial\xi}{\partial x} - \lambda_1\frac{\partial\xi}{\partial y}\right)\left(\frac{\partial\xi}{\partial x} - \lambda_2\frac{\partial\xi}{\partial y}\right) \ , \tag{1.2.8}$$

$$C_1 = A\left(\frac{\partial\eta}{\partial x} - \lambda_1\frac{\partial\eta}{\partial y}\right)\left(\frac{\partial\eta}{\partial x} - \lambda_2\frac{\partial\eta}{\partial y}\right) \ . \tag{1.2.9}$$

If we choose ξ and η such that,

$$\frac{\partial\xi}{\partial x} - \lambda_1\frac{\partial\xi}{\partial y} = 0 \tag{1.2.10}$$

and

$$\frac{\partial\eta}{\partial x} - \lambda_2\frac{\partial\eta}{\partial y} = 0 \ , \tag{1.2.11}$$

then the coefficients of $\frac{\partial^2 U}{\partial\xi^2}$ and $\frac{\partial^2 U}{\partial\eta^2}$ will vanish, i.e., $A_1=C_1=0$, and we can solve for x and y in terms of ξ and η. For example, if we let,

$$\xi = \lambda_1 x + y \quad \text{and} \quad \eta = \lambda_2 x + y \ . \tag{1.2.12}$$

Thus, the two equations (1.2.10) and (1.2.11) are acceptable. Hence from equation (1.2.12)

$$x = \frac{\xi-\eta}{\lambda_1-\lambda_2} \quad \text{and} \quad y = \frac{\lambda_1\eta-\lambda_2\xi}{\lambda_1-\lambda_2} \tag{1.2.13}$$

with $\lambda_1-\lambda_2\neq 0$.

Equation (1.2.5) then becomes,

$$B_1 \frac{\partial^2 U}{\partial\xi\partial\eta} + D_1 \frac{\partial U}{\partial\xi} + E_1 \frac{\partial U}{\partial\eta} + F_1 U + G_1 = 0 \ . \tag{1.2.14}$$

This is a second order equation if $B_1 \neq 0$. Let us check whether this is so, as we know,

$$\lambda_1 = \frac{-B+\sqrt{B^2-4AC}}{2A} \quad \text{and} \quad \lambda_2 = \frac{-B-\sqrt{B^2-4AC}}{2A} \tag{1.2.15}$$

Hence by addition and multiplication we get,

$$B = -A(\lambda_1+\lambda_2) \quad \text{and} \quad C = A\lambda_1\lambda_2 \text{ respectively.}$$

Therefore, by substituting these values in equation (1.2.6.ii) we obtain

$$B_1 = A[2\frac{\partial\xi}{\partial x}\frac{\partial\eta}{\partial x} - (\lambda_1+\lambda_2)(\frac{\partial\xi}{\partial x}\frac{\partial\eta}{\partial y} + \frac{\partial\xi}{\partial y}\frac{\partial\eta}{\partial x}) + 2\lambda_1\lambda_2 \frac{\partial\xi}{\partial y}\frac{\partial\eta}{\partial y}]$$

$$= A[\frac{\partial\xi}{\partial x}\frac{\partial\eta}{\partial x} - \lambda_1 \frac{\partial\xi}{\partial y}\frac{\partial\eta}{\partial x} - \lambda_2 \frac{\partial\xi}{\partial x}\frac{\partial\eta}{\partial y} + \lambda_1\lambda_2 \frac{\partial\xi}{\partial y}\frac{\partial\eta}{\partial y} + \frac{\partial\xi}{\partial x}\frac{\partial\eta}{\partial x} - \lambda_2 \frac{\partial\xi}{\partial y}\frac{\partial\eta}{\partial x} -$$

$$\lambda_1 \frac{\partial\xi}{\partial x}\frac{\partial\eta}{\partial y} + \lambda_1\lambda_2 \frac{\partial\xi}{\partial y}\frac{\partial\eta}{\partial y}]$$

$$= A[(\frac{\partial\xi}{\partial x} - \lambda_1 \frac{\partial\xi}{\partial y})(\frac{\partial\eta}{\partial x} - \lambda_2 \frac{\partial\eta}{\partial y}) + (\frac{\partial\xi}{\partial x} - \lambda_2 \frac{\partial\xi}{\partial y})(\frac{\partial\eta}{\partial x} - \lambda_1 \frac{\partial\eta}{\partial y})] \ . \tag{1.2.16}$$

By substituting equation (1.2.10) in (1.2.16),

$$B_1 = A(\frac{\partial\xi}{\partial x} - \lambda_2 \frac{\partial\xi}{\partial y})(\frac{\partial\eta}{\partial x} - \lambda_1 \frac{\partial\eta}{\partial y}) \ , \tag{1.2.17}$$

But the two brackets in equation (1.2.17) gives us $(\lambda_1-\lambda_2)$ and $(\lambda_2-\lambda_1)$ respectively.

Hence, $B_1 \neq 0$. Equation (1.2.14) can be written as,

$$\frac{\partial^2 U}{\partial\xi\partial\eta} + D_2 \frac{\partial U}{\partial\xi} + E_2 \frac{\partial U}{\partial\eta} + F_2 U + G_2 = 0 \ , \tag{1.2.18}$$

where, $D_2 = \frac{D_1}{B_1}$, $E_2 = \frac{E_1}{B_1}$, $F_2 = \frac{F_1}{B_1}$ and $G_2 = \frac{G_1}{B_1}$. (1.2.19)

These type of equations are called *Hyperbolic equations*.

(ii) If $B^2-4AC=0$, then the roots of the quadratic equation (1.2.7) are real and equal $(\lambda_1=\lambda_2=\lambda)$, hence from equation (1.2.6),

$$A_1 = A(\frac{\partial\xi}{\partial x} - \lambda \frac{\partial\xi}{\partial y})^2 \ ,$$

$$B_1 = 2A(\frac{\partial\xi}{\partial x} - \lambda \frac{\partial\xi}{\partial y})(\frac{\partial\eta}{\partial x} - \lambda \frac{\partial\eta}{\partial y}) \ , \tag{1.2.20}$$

and

$$C_1 = A(\frac{\partial\eta}{\partial x} - \lambda \frac{\partial\eta}{\partial y})^2 \ .$$

If we choose ξ and η such that,

$$\frac{\partial \eta}{\partial x} - \lambda \frac{\partial \eta}{\partial y} = 0 \text{ and } \frac{\partial \xi}{\partial x} - \lambda \frac{\partial \xi}{\partial y} \neq 0 \tag{1.2.21}$$

then $B_1 = C_1 = 0$ and $A_1 \neq 0$. For example, if we let

$$\xi = x \qquad \text{and} \qquad \eta = \lambda x + y \ , \tag{1.2.22}$$

then equation (1.2.21) is acceptable, and equation (1.2.5) can be reduced to the form,

$$A_1 \frac{\partial^2 U}{\partial \xi^2} + D_1 \frac{\partial U}{\partial \xi} + E_1 \frac{\partial U}{\partial \eta} + F_1 U + G_1 = 0 \ . \tag{1.2.23}$$

Since $A_1 \neq 0$, we can divide equation (1.2.23) through by A_1 to get,

$$\frac{\partial^2 U}{\partial \xi^2} + D_3 \frac{\partial U}{\partial \xi} + E_3 \frac{\partial U}{\partial \eta} + F_3 U + G_3 = 0 \ , \tag{1.2.24}$$

where, $D_3 = \dfrac{D_1}{A_1}$, $E_3 = \dfrac{E_1}{A_1}$, $F_3 = \dfrac{F_1}{A_1}$ and $G_3 = \dfrac{G_1}{A_1}$.

These type of equations are called *Parabolic equations*.
(The addition of a second-order derivative such as $H(\frac{\partial^2 U}{\partial z^2})$, where H is a positive number, does not influence the basic parabolic character of the equation).

(iii) If $B^2 - 4AC < 0$, then the roots of equation (1.2.7) are complex conjugate numbers, λ and $\bar{\lambda}$, (say), similar to the hyperbolic case, let us choose ξ and η such that,

$$\frac{\partial \xi}{\partial x} - \lambda \frac{\partial \xi}{\partial y} = 0 \quad \text{and} \quad \frac{\partial \eta}{\partial x} - \bar{\lambda} \frac{\partial \eta}{\partial y} = 0 \ , \tag{1.2.25}$$

the coefficients A_1 and C_1 of equation (1.2.5) will vanish and we will obtain equation (1.2.14) once again. But ξ and η are now complex variables, in order to remain in the domain of the real variables we must write equation (1.2.2) in terms of real variables, say α and β.

Let,

$$\begin{aligned} \xi &= \alpha(x,y) + i\beta(x,y) \\ \eta &= \alpha(x,y) - i\beta(x,y) \end{aligned} \tag{1.2.26}$$

By adding and subtracting the two equations in (1.2.26) we obtain,

$$\alpha = \frac{1}{2}(\xi+\eta) \quad \text{and} \quad \beta = \frac{1}{2i}(\xi-\eta) \text{ respectively,} \tag{1.2.27}$$

Let us compute the appropriate differentiation of U

$$\frac{\partial U}{\partial \xi} = \frac{\partial U}{\partial \alpha}\frac{\partial \alpha}{\partial \xi} + \frac{\partial U}{\partial \beta}\frac{\partial \beta}{\partial \xi} = \tfrac{1}{2}\left(\frac{\partial U}{\partial \alpha} - i\,\frac{\partial U}{\partial \beta}\right) ,$$

$$\frac{\partial U}{\partial \eta} = \frac{\partial U}{\partial \alpha}\frac{\partial \alpha}{\partial \eta} + \frac{\partial U}{\partial \beta}\frac{\partial \beta}{\partial \eta} = \tfrac{1}{2}\left(\frac{\partial U}{\partial \alpha} + i\,\frac{\partial U}{\partial \beta}\right) ,$$

$$\begin{aligned}\frac{\partial^2 U}{\partial \xi \partial \eta} &= \frac{\partial}{\partial \xi}\left(\frac{\partial U}{\partial \alpha}\frac{\partial \alpha}{\partial \eta} + \frac{\partial U}{\partial \beta}\frac{\partial \beta}{\partial \eta}\right) \\ &= \frac{1}{2}\frac{\partial}{\partial \xi}\left(\frac{\partial U}{\partial \alpha} + i\,\frac{\partial U}{\partial \beta}\right) \\ &= \frac{1}{2}\left[\frac{\partial^2 U}{\partial \alpha^2}\frac{\partial \alpha}{\partial \xi} + \frac{\partial^2 U}{\partial \beta \partial \alpha} + \frac{\partial \beta}{\partial \xi} + i\left(\frac{\partial^2 U}{\partial \alpha \partial \beta}\frac{\partial \alpha}{\partial \xi} + \frac{\partial^2 U}{\partial \beta^2}\frac{\partial \beta}{\partial \xi}\right)\right] \\ &= \frac{1}{4}\left(\frac{\partial^2 U}{\partial \alpha^2} + \frac{\partial^2 U}{\partial \beta^2}\right) .\end{aligned} \tag{1.2.28}$$

Hence equation (1.2.14) can be written in the form,

$$\frac{B_1}{4}\left(\frac{\partial^2 U}{\partial \alpha^2} + \frac{\partial^2 U}{\partial \beta^2}\right) + \frac{D_1}{2}\left(\frac{\partial U}{\partial \alpha} - i\,\frac{\partial U}{\partial \beta}\right) + \frac{E_1}{2}\left(\frac{\partial U}{\partial \alpha} + i\frac{\partial U}{\partial \beta}\right) + F_1 U + G_1 = 0 .$$

This equation can be rearranged to give,

$$\frac{B_1}{4}\left(\frac{\partial^2 U}{\partial \alpha^2} + \frac{\partial^2 U}{\partial \beta^2}\right) + \frac{\partial U}{\partial \alpha}\left(\frac{D_1}{2} + \frac{E_1}{2}\right) + i\,\frac{\partial U}{\partial \beta}\left(\frac{E_1}{2} - \frac{D_1}{2}\right) + F_1 U + G_1 = 0 \tag{1.2.29}$$

and since $B_1 \neq 0$, we can divide equation (1.2.29) through by $\frac{B_1}{4}$ to obtain,

$$\frac{\partial^2 U}{\partial \alpha^2} + \frac{\partial^2 U}{\partial \beta^2} + D_4\,\frac{\partial U}{\partial \alpha} + E_4\,\frac{\partial U}{\partial \beta} + F_4 U + G_4 = 0 , \tag{1.2.30}$$

where,
$$D_4 = \frac{2(D_1+E_1)}{B_1} , \quad E_4 = \frac{2i(E_1-D_1)}{B_1} , \quad F_4 = \frac{4F_1}{B_1} ,$$

and
$$G_4 = \frac{4G_1}{B_1} .$$

These types of equations are called *Elliptic equations*. (This definition still applies even if another term $H\left(\frac{\partial^2 U}{\partial z^2}\right)$, say, where H is a positive number, is added to the left-hand side of equation (1.2.1)).

This classification of p.d.e.'s into these three categories is crucial in determining the fundamental features of the solution.

The above classification scheme is rather interesting, since the coefficients A,B and C are functions of the independent variables x and y and/or the dependent variable U, thus this classification depends in general on the region in which the p.d.e. (1.2.1) is defined. For example, the differential equation,

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

is hyperbolic in the region when $x^2-y^2>0$, parabolic along the boundary $x^2-y^2=0$ and elliptic in the region where $x^2-y^2<0$.

The best known examples for the above cases are:

Hyperbolic

The wave equation $$\frac{\partial^2 U}{\partial x^2} = \frac{1}{c^2}\frac{\partial^2 U}{\partial t^2} \qquad (1.2.31)$$

where c is the propagation velocity.

Parabolic

Diffusion or heat conduction equation $$\frac{\partial^2 U}{\partial x^2} = \frac{1}{a^2}\frac{\partial U}{\partial t} \qquad (1.2.32)$$

where a^2 is a physical constant.

Elliptic

Laplace's equation $$\nabla^2 U = \frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} = 0 \ , \qquad (1.2.33a)$$

Poisson's equation $$\nabla^2 U = \frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} = f(x,y) \ , \qquad (1.2.33b)$$

where ∇^2 is the two-dimensional Laplacian differential operator $(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2})$.

In general, parabolic and hyperbolic p.d.e.'s, result from diffusion, equalization or oscillatory processes and the usual independent variables are time and space. On the other hand the two elliptic p.d.e.'s (1.2.33a)-(1.2.33b) are generally associated with steady-state or equilibrium problems. For example, Laplace's equation describes the velocity potential for the steady flow of an incompressible non-viscous fluid and is the mathematical expression of the physical law that the rate at which such fluid enters a given region is equal to the rate at which it leaves it. In contrast, the electric potential, associated with a two dimensional electron distribution of charge density, satisfies Poisson's equation, stating that the total electric flux through any closed surface is equal to the total charge enclosed.

1.3 PROPERLY POSED AND WELL-CONDITIONED PROBLEMS

In practical applications, a particular solution for a differential equation is required to satisfy specified auxiliary conditions. These auxiliary conditions may appear either as boundary and/or initial conditions and hence the appropriate numerical method for solving the differential problem depends upon the nature of these auxiliary conditions.

We say that the differential problem is:

(i) *Properly posed (well-posed)* If the auxiliary conditions are specified in such a way that there exists one and only one solution to the problem, furthermore this solution depends continuously on the given data. A problem that is not well-posed is said to be ill-posed or non-well-posed.

(ii) *Well-conditioned* If every small perturbation (error) in the data of the well-posed problem results in a relatively small change in the solution. If the change in the solution is large we say that the problem is ill-conditioned.

To clarify the well-posedness condition, consider the Laplace equation in two-dimensions,

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} = 0 , \tag{1.3.1}$$

with the given initial-value data ,

$$U(x,0) = 0 , \tag{1.3.2}$$

$$\frac{\partial U}{\partial y}(x,0) = \frac{\sin(nx)}{n} \tag{1.3.3}$$

where n is a parameter.

The analytical solution can be found by the method of separation of variables. The main assumption in this method consists in stating that the solution of equation (1.3.1) can be written in the following form,

$$U(x,y) = X(x)Y(y) \; . \tag{1.3.4}$$

By substituting equation (1.3.4) into (1.3.1) we obtain,

$$X''Y + XY'' = 0 \; , \tag{1.3.5}$$

or
$$\frac{X''}{X} = -\frac{Y''}{Y} \; , \tag{1.3.6}$$

where the primes on the functions X and Y represent differentiation with respect to the only variable present. Since the left and the right-hand sides of equation (1.3.6) are independent of y and x respectively, therefore their value is a constant, say k. Thus,

$$X'' = kX \quad \text{and} \quad Y'' = -kY \; ,$$

we shall assume that $k=-n^2<0$. The general solution of

$$X'' + n^2X = 0 \; , \tag{1.3.7}$$

is
$$X(x) = A\cos(nx) + B\sin(nx) \; , \tag{1.3.8}$$

and the general solution of

$$Y'' - n^2Y = 0 \; , \tag{1.3.9}$$

is,
$$Y(y) = C\cosh(ny) + D\sinh(ny) \; . \tag{1.3.10}$$

Now from (1.3.2) ,

$$0 = U(x,0) = X(x).C \; , \tag{1.3.11}$$

$$\Rightarrow \qquad C = 0 \; , \tag{1.3.12}$$

hence
$$Y(y) = D\sinh(ny) \; , \tag{1.3.13}$$

$$\frac{\partial U(x,y)}{\partial y} = X'Y + XY' \; ,$$

hence,
$$\frac{\partial U(x,y)}{\partial y} = -n(A\sin(nx) - B\cos(nx)).D\sinh(ny) + (A\cos(nx) + B\sin(nx))\,nD\cosh(ny) \; . \tag{1.3.14}$$

From the boundary condition (1.3.3), we get,

$$\frac{\sin(nx)}{n} = nDA\cos(nx) + nDB\sin(nx) \; , \tag{1.3.15}$$

or
$$\sin(nx) = n^2 DA\cos(nx) + n^2DB\sin(nx) \; . \tag{1.3.16}$$

From equation (1.3.16) we obtain,

$$n^2 DB = 1 \quad \text{and} \quad n^2 DA = 0 \ , \tag{1.3.17}$$

which implies that

$$A = 0 \ , \tag{1.3.18}$$

and

$$BD = \frac{1}{n^2} \ , \tag{1.3.19}$$

hence by substituting the value of A in equation (1.3.8) we get,

$$X(x) = B \sin nx \ , \tag{1.3.20}$$

by substituting the values of $X(x)$ and $Y(y)$ from equations (1.3.20) and (1.3.13) respectively in equation (1.3.4), we obtain,

$$U(x,y) = \frac{1}{n^2} \sin(nx) \sinh(ny) \ . \tag{1.3.21}$$

As $n \to \infty$, the initial data converge to $U(x,0) = 0$, and $\frac{\partial U}{\partial y} = 0$.

However, for $y > 0$, as $n \to \infty$, $U(x,y)$ given by (1.3.21) oscillates between limits that increase indefinitely. Consequently, the problem defined by (1.3.I)-(1.3.3) is not well-posed and could not be associated with any physical phenomenon.

1.4 TYPES OF BOUNDARY CONDITIONS

For each of the classified equations, i.e., the elliptic, parabolic and hyperbolic equations, we must have the appropriate boundary and/or initial conditions, which assists to complete the formulation of a 'meaningful problem'. Usually, the elliptic p.d.e.'s are classified as boundary value problems, since the boundary conditions are given round the *(closed)* region (see Figure (1.4.1)).

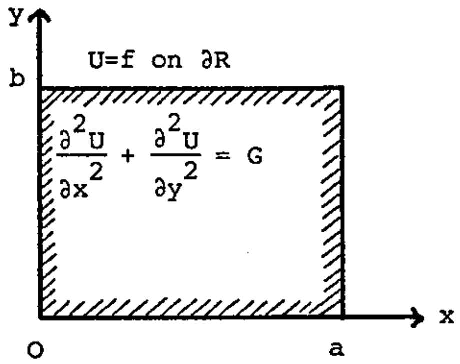
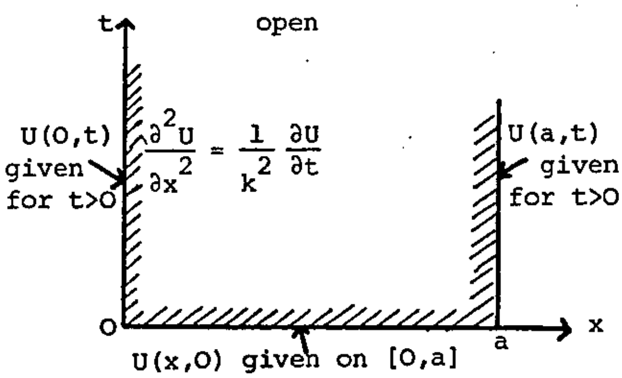
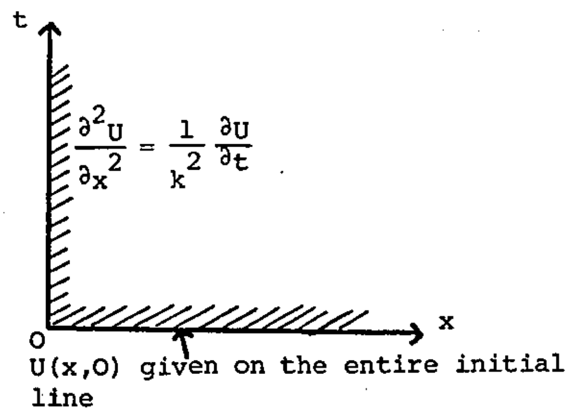


FIGURE (1.4.1): Closed region (boundary value problem)

The parabolic and hyperbolic types of equations are either initial value problems or initial boundary value problems, where the initial or/ and boundary conditions are supplied on the sides of the *open* region, and the solution proceeds towards the open side (see Figure (1.4.2)-(1.4.3)).



(a) Initial boundary value problem for a parabolic equation



(b) Initial value problem for a parabolic equation

FIGURE (1.4.2)

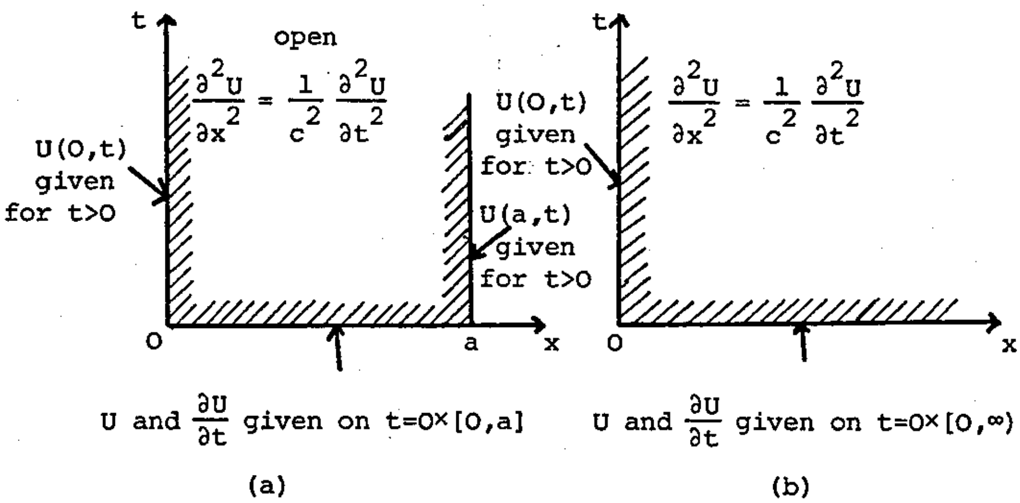


FIGURE (1.4.3): Initial boundary value problem for Hyperbolic equation

In this thesis we are mainly concerned with elliptic problems. In general, the classification of elliptic problems lie in five main categories depending on the boundary conditions defined on the boundary (∂R) of the closed region R, these are:

1. *Dirichlet problem*, where the solution U is specified at each point on ∂R.
2. *Neumann problem*, where values of the normal derivatives $(\frac{\partial U}{\partial n})$ are given on ∂R. ($\frac{\partial U}{\partial n}$ denotes the directional derivative of U along the outward normal to ∂R).
3. *Robin's problem*, where a combination of U and its derivatives are given along the boundary, i.e. $\alpha U+\beta\frac{\partial U}{\partial n}$ given on ∂R.
4. *Mixed problem* or (*Churchill problem* as referred by Kersten (1969)) where the solution U is specified on part of ∂R and $\frac{\partial U}{\partial n}$ is specified on the remainder of ∂R.
5. *Periodic boundary problem*. In this case we seek the solution such

that it satisfies the periodicity conditions, for example,

$$U_x = U_{x+\ell}, \left(\frac{\partial U}{\partial n}\right)_x = \left(\frac{\partial U}{\partial n}\right)_{x+\ell}$$

where ℓ is the period, x and $x+\ell$ are on ∂R.

CHAPTER 2

BASIC MATHEMATICAL CONCEPTS

2.1 BASIC MATRIX ALGEBRA

The solution of p.d.e.'s by numerical approaches such as the finite difference and finite element methods yields a system of linear, simultaneous equations which can be denoted by a matrix system. The iterative methods of solving such systems depend on some properties, for example, *irreducibility*, *diagonal dominance* and *positive definiteness* of the coefficient matrix of the system. These and other properties together with some basic facts about matrix theory are outlined in this chapter.

A *matrix* is a rectangular array of doubly subscripted *elements* such as $a_{i,j}$, where i specifies the row and j specifies the column of the array in which the element appears. A matrix A, say, is of *size* (n×m) if it has n rows and m columns, and can be denoted by,

$$A = [a_{i,j}] = \begin{bmatrix} a_{1,1} & a_{1,2} & --- & a_{1,m} \\ a_{2,1} & a_{2,2} & --- & a_{2,m} \\ \vdots & \vdots & & \vdots \\ a_{n,1} & a_{n,2} & --- & a_{n,m} \end{bmatrix} .$$

Where n=1, we have a row matrix or *row vector*, and for m=1, we have a column matrix or *column vector*. The vectors are usually denoted by a small underlined letter with a single subscript. The term 'vector' without qualification will refer to a column vector, so a vector $\underline{b}$, say, whose elements are $b_1, b_2, \ldots, b_n$, are of order (dimension) n is denoted by,

$$\underline{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} .$$

In this thesis, we shall use capital letters for matrices. In addition, all matrices used are square matrices, unless otherwise specified, the matrix is said to be a *square matrix* of *order* n if n=m.

The set of elements $a_{i,i}$, $i=1,2,\ldots,n$ of matrix A is a *principal* (main) diagonal of A. The *transpose* of a matrix $A=(a_{i,j})$ is written as A^T and obtained by interchanging the rows and columns of A, i.e. the element $a_{i,j}$ of A becomes $a_{j,i}$ of A^T. The *determinant* of a square matrix A will be denoted either by det(A) or $|A|$. An *inverse*, A^{-1}, of a given square matrix A, if it exists, is a square matrix such that,

$$AA^{-1} = A^{-1}A = I \ , \tag{2.1.1}$$

where I is the *identity* (unit) matrix whose order is the same as that of A and is defined as follows,

$$a_{i,i} = 1 \ , \text{ for all } i=1,2,\ldots,n$$
$$a_{i,j} = 0 \ , \text{ for all } i,j=1,2,\ldots,n \text{ and } i\neq j.$$

If A possesses an inverse then it is *non-singular* otherwise it is *singular*. On the other hand, A is singular if $|A|=0$ and non-singular if $|A|\neq 0$.

If x and y are real numbers, the *conjugate* of the complex number $a=x+yi$ is $\bar{a}=x-yi$.

If the entries of a matrix A are complex numbers, the *conjugate* of A is the matrix $\bar{A}$ whose entries are the conjugates of the corresponding entries of A, i.e., if $A=[a_{i,j}]$ then $\bar{A}=[\bar{a}_{i,j}]$.

The *Hermitian transpose* (conjugate transpose) of A, denoted by A^H, is the transpose of $\bar{A}$ (also the conjugate of A^T), i.e.,

$$A^H = (\bar{A})^T = \overline{A^T} = [\bar{a}_{j,i}] \ . \tag{2.1.2}$$

The sum of the diagonal elements of a matrix A is called the *trace* of A, denoted by tr(A), i.e.,

$$tr(A) = \sum_{i=1}^{n} a_{i,i} \ , \tag{2.1.3}$$

A *permutation matrix* $P=[p_{i,j}]$ is a matrix which has elements of zeros and ones only with exactly one non-zero element in each row and each column. Thus, for example,

$$P = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} ,$$

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

$$PP^T = P^TP = I \ , \tag{2.1.4}$$

hence

$$P^T = P^{-1} \ .$$

For any two vectors $\underline{a}$ and $\underline{b}$, both of dimension n we define the *inner product* of $\underline{a}$ and $\underline{b}$ by,

$$(\underline{a},\underline{b}) = \underline{a}^H\underline{b} = \sum_{i=1}^{n} \bar{a}_i b_i \ . \tag{2.1.5}$$

Further, for any matrix A,

$$(\underline{a},A\underline{b}) = (A^H\underline{a},\underline{b}) \ . \tag{2.1.6}$$

Given a matrix $A=[a_{i,j}]$, the integers i and j are *associated* with respect to A if $a_{i,j}\neq 0$ or $a_{j,i}\neq 0$.

The matrix $A=[a_{i,j}]$ of order n is,

Symmetric, if $A=A^T$.

Orthogonal, if $A^{-1}=A^T$.

Hermitian, if $A^H=A$.

Null, usually denoted by $\bigcirc$, if $a_{i,j}=0$, $(i,j=1,2,\ldots,n)$.

Diagonal, if $a_{i,j}=0$ for $i\neq j$ $(|i-j|>0)$

where $|\alpha|$ represents the modulus of any number α.

Banded, if $a_{i,j}=0$ for $|i-j|>r$, where $2r+1$ is the bandwidth of A.

Tridiagonal, if r=1 .

Quindiagonal, if r=2, (see Figure (2.1.1)) .

Lower triangular, (strictly lower triangular), if $a_{i,j}=0$ for $i<j$ ($i\leq j$).

Upper triangular, (strictly upper triangular), if $a_{i,j}=0$ for $i>j$ ($i\geq j$).

Sparse, if a relatively large number of its elements $a_{i,j}$ are zero.

Dense, (full), if a relatively large number of its elements are non-zero.

$$A = \begin{bmatrix} x & x & & & & \\ x & x & x & & \bigcirc & \\ & x & x & x & & \\ & & \ddots & \ddots & \ddots & \\ & \bigcirc & & x & x & x \\ & & & & x & x & x \\ & & & & & x & x \end{bmatrix}, \text{ or } \begin{bmatrix} x & x & x & & & & \\ x & x & x & x & & \bigcirc & \\ x & x & x & x & x & & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & x & x & x & x & x \\ & \bigcirc & & x & x & x & x \\ & & & & x & x & x \end{bmatrix}$$

r=1 r=2

FIGURE (2.1.1): Banded matrices (tridiagonal, quindiagonal) where x denotes a possible non-zero element

Block Diagonal, if

$$A = \begin{bmatrix} D_1 & & & & \\ & D_2 & & \bigcirc & \\ & & D_3 & & \\ & \bigcirc & & \ddots & \\ & & & & D_s \end{bmatrix},$$

where each D_i ($i=1,2,\ldots,s$) is a square matrix, not necessarily of the same order.

Block Tridiagonal, if

$$A = \begin{bmatrix} D_1 & F_1 & & & \\ E_2 & D_2 & F_2 & & \bigcirc \\ & E_3 & D_3 & F_3 & \\ & & \ddots & \ddots & F_{s-1} \\ & \bigcirc & & E_s & D_s \end{bmatrix},$$

where the D_i are square diagonal matrices, not necessarily of the same order, while the E's and the F's are rectangular matrices. Young (1971) referred to such a matrix as a *T-matrix*.

2.2 DIAGONAL DOMINANCE AND IRREDUCIBILITY

Definition 2.2.1

An $(n\times n)$ matrix A is *diagonally dominant* if

$$|a_{i,i}| \geq \sum_{\substack{j=1\\ j\neq i}}^{n} |a_{i,j}| \text{ , for all } 1\leq i\leq n \text{ ,} \tag{2.2.1}$$

and for at least one i,

$$|a_{i,i}| > \sum_{\substack{j=1\\ j\neq i}}^{n} |a_{i,j}| . \tag{2.2.2}$$

Definition 2.2.2

An $(n\times n)$ matrix A is *irreducible* if $n=1$ or if $n>1$ and given any two non-empty disjoint subsets S and T of W, the set of the first n positive integers, such that $S+T=W$, there exists $i \in S$ and $j \in T$ such that $a_{i,j}\neq 0$. This definition is given by Young (1971). Varga (1962) stated that the (1×1) matrix is irreducible if its single element is non-zero and reducible otherwise.

The following theorems give an alternative definition of irreducibility. Young (1971).

Theorem 2.2.1

A is irreducible if and only if there does not exist a permutation matrix P such that,

$$P^{-1}AP = \begin{bmatrix} F & \bigcirc \\ & \\ G & H \end{bmatrix} \tag{2.2.3}$$

where F and H are square matrices and $\bigcirc$ is the null matrix.

Theorem 2.2.2

An $(n\times n)$ matrix A is irreducible if and only if $n=1$ or, given any two distinct integers i and j with $1 \leq i,j \leq n$, then $a_{i,j} \neq 0$ or there exists $i_1, i_2, \ldots, i_s$ such that,

$$a_{i,i_1} a_{i_1,i_2}, \ldots, a_{i_s,j} \neq 0 .$$

Further, we can illustrate the concept of irreducibility graphically, which is a simple and quite useful method. We start by introducing the definition of a finite directed graph, as given in Varga (1962).

Definition 2.2.3

Let A be an $(n \times n)$ matrix, and consider any n distinct points, $P_1, P_2, \ldots, P_n$, in the plane, which we shall call *nodes*. For every non-zero element $a_{i,j}$ of the matrix, connect the node P_i to the node P_j by means of a path $\overrightarrow{P_iP_j}$, directed from P_i to P_j, as in Figure (2.2.1). (For non-zero diagonal elements $a_{i,i}$ the path goes from P_i to itself forming a *loop*,, as in Figure (2.2.2)). The resulting diagram is called a *finite directed graph*. G(A).

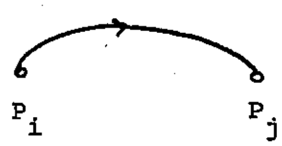


FIGURE (2.2.1)

FIGURE (2.2.2)

As an example, consider the two matrices,

$$B = \begin{bmatrix} 4 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 4 \end{bmatrix} \quad \text{and} \quad C = \begin{bmatrix} 1 & 0 \\ 1 & 2 \end{bmatrix}$$

G(B) and G(C) are given in Figures (2.2.3) and (2.2.4) respectively.

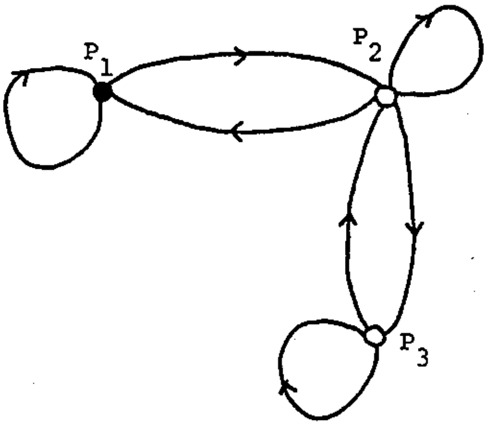


FIGURE (2.2.3)

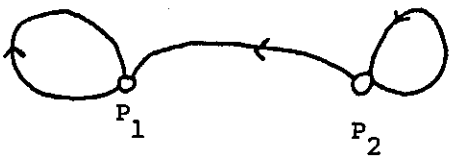


FIGURE (2.2.4)

Definition 2.2.4

A directed graph is *strongly connected* if for any ordered pair of nodes P_i and P_j, there exists a directed path,

$$\overrightarrow{P_iP_{k_1}}, \overrightarrow{P_{k_1}P_{k_2}}, \overrightarrow{P_{k_2}P_{k_3}}, \ldots, \overrightarrow{P_{k_{r-1}}P_{k_r=j}}$$

connecting P_i to P_j.

Such a path is said to have *length r* (Varga, [1962], p.20).

Clearly, the directed graph G(B) in Figure (2.2.3) is strongly connected. On the other hand G(C) is not strongly connected, since there does not exist a path from P_1 to P_2.

The next theorem describes the relationship between the irreducibility of a matrix and its directed graph.

Theorem 2.2.3

A square matrix A is irreducible if and only if its directed graph G(A) is strongly connected.

Definition 2.2.5

An irreducible matrix which is also diagonally dominant with strict inequality holding for at least one i in (2.2.2) is said to be *irreducibly diagonally dominant*.

If (2.2.2) holds for all i, then A has *strong diagonal dominance*.

Theorem 2.2.4

If A is irreducibly diagonally dominant matrix, then $\det(A) \neq 0$ and none of the diagonal elements of A vanishes. The proof of this is given by Young (1971), p.40, also see Varga (1962), p.23.

2.3 VECTOR AND MATRIX NORMS

It is useful to have some measure of the 'size' or magnitude of a vector or matrix. This measure is called a *norm* and is denoted by $||\cdot||$.

Definition 2.3.1

The *norm* of a vector $\underline{x}$, denoted by $||\underline{x}||$, is a non-negative number satisfying the following three axioms:

$$\left.\begin{array}{l} (1)\ ||\underline{x}||>0 \text{ for } \underline{x}\neq\underline{0} \text{ and } ||\underline{x}||=0 \text{ if } \underline{x}=\underline{0}, \\ (2)\ ||\beta\underline{x}||=|\beta|.||\underline{x}|| \text{ for any complex scalar } \beta, \\ (3)\ ||\underline{x}+\underline{y}||\leq||\underline{x}||+||\underline{y}|| \text{ for vectors } \underline{x} \text{ and } \underline{y} \end{array}\right\} \qquad (2.3.1)$$

(triangular inequality).

Also, from (3) we have,

$$||\underline{x}-\underline{y}|| \geq |\ ||\underline{x}||-||\underline{y}||\ |. \qquad (2.3.2)$$

The most commonly used norms are the L_1, L_2 and L_∞ norms of $\underline{x}$ and they are defined as follows:

Definition 2.3.2

If $\underline{x}=(x_1,x_2,\ldots,x_n)^T$ is a n-dimensional vector then

$$||\underline{x}||_1 = \sum_{i=1}^{n} |x_i| \ , \qquad (2.3.3)$$

$$||\underline{x}||_2 = (\sum_{i=1}^{n} |x_i|^2)^{\frac{1}{2}} \ , \text{ (Euclidean norm)}, \qquad (2.3.4)$$

$$||\underline{x}||_\infty = \max_i |x_i|, \text{ (maximum or uniform norm)}. \qquad (2.3.5)$$

In fact, these three norms are special cases of the general L_p norm defined for $p\geq1$ by,

$$||\underline{x}||_p = (\sum_{i=1}^{n} |x_i|^p)^{1/p} \ . \qquad (2.3.6)$$

In a similar fashion, we can proceed to define a matrix norm.

Definition 2.3.3

A norm of an $(n\times n)$ matrix A, written as $||A||$, is a scalar satisfying the following axioms:

$$\left.\begin{array}{l} (1)\ ||A||>0 \text{ and } ||A||=0 \text{ if and only if } A=[0], \\ (2)\ ||\beta.A||=|\beta|.||A||, \text{ for any scalar } \beta, \\ (3)\ ||A+B||\leq||A||+||B|| \text{ for matrices A and B}, \\ (4)\ ||AB||\leq||A||.||B|| \text{ for matrices A and B.} \end{array}\right\} \qquad (2.3.7)$$

The matrix norms subordinate to the L_1, L_2 and L_∞ vector norms are

$$||A||_1 = \max_j \sum_{i=1}^{n} |a_{i,j}| \quad \text{(maximum absolute column sum)}, \qquad (2.3.8)$$

$$||A||_2 = (\text{maximum eigenvalue of } A^H A)^{\frac{1}{2}}, \quad \text{(spectral norm)}, \qquad (2.3.9)$$

$$||A||_\infty = \max_i \sum_{j=1}^{n} |a_{i,j}| \quad \text{(maximum absolute row sum)} . \qquad (2.3.10)$$

Another norm compatible with the L_2 vector norm is the *Euclidean* or *Schür norm* and is defined as follows:

$$||A||_E = (\sum_{i,j} |a_{i,j}|^2)^{\frac{1}{2}} . \qquad (2.3.11)$$

Definition 2.3.4

A matrix norm $||A||$ is said to be *compatible* with a vector norm $||\underline{x}||$ if

$$||A\underline{x}|| \leq ||A||.||\underline{x}|| \text{ , for all } \underline{x}\neq\underline{0} . \qquad (2.3.12)$$

Definition 2.3.5

A matrix norm is said to be *subordinate* to the corresponding vector norm, if it can be constructed in the following form:

$$||A|| = \max_{\underline{x}\neq\underline{0}} \frac{||A\underline{x}||}{||\underline{x}||} , \qquad (2.3.13)$$

or equivalently,
$$||A|| = \max_{||\underline{x}||=1} ||A\underline{x}|| . \qquad (2.3.14)$$

Definition 2.3.6

A vector is said to be *normalised vector* if it is multiplied by a scalar in order to keep the element size down without changing its direction.

The most useful normalisation for our work is described below: Suppose that $\underline{x}=(x_1,x_2,\ldots,x_n)^T$, in order to normalise $\underline{x}$ we select a scalar s such that $s=\max_{1\leq i\leq n} |x_i|$, and the normalised vector is then given by $(\frac{x_1}{s}, \frac{x_2}{s},\ldots,\frac{x_n}{s})^T$. This method of normalisation ensures that the modulus of every element of the vector is less than or equal to 1.

2.4 EIGENVALUES AND EIGENVECTORS

Definition 2.4.1

Suppose that A is an (n×n) matrix and $\underline{x} \neq \underline{0}$ is a vector of order n. If there exists a scalar λ such that,

$$A\underline{x} = \lambda \underline{x} , \qquad (2.4.1)$$

then λ is called an *eigenvalue* (characteristic root or latent root) of A and $\underline{x}$ its corresponding *eigenvector* (characteristic vector or latent vector) of A.

The system (2.4.1) can be written as,

$$(A-\lambda I)\underline{x} = \underline{0} . \qquad (2.4.2)$$

The non-trivial solution, $\underline{x} \neq \underline{0}$ to this matrix equation exists if and only if the matrix of the system is singular, i.e.,

$$\det(A-\lambda I) = 0 . \qquad (2.4.3)$$

Equation (2.4.3) is called the *characteristic equation* of A and the left-hand side is called the *characteristic polynomial* of A, which can be written as,

$$\alpha_0+\alpha_1\lambda+ \ldots +\alpha_{n-1}\lambda^{n-1}+(-1)^n\lambda^n = 0 . \qquad (2.4.4)$$

Since the coefficient of λ^n is not zero, the above equation has always n roots (complex or real) which are the n eigenvalues of the matrix A, namely, $\lambda_1,\lambda_2,\ldots,\lambda_n$ (not necessarily all distinct), each of them possessing a unique corresponding eigenvector.

In physical problems we rarely need to determine all the eigenvalues of equation (2.4.2). In particular, it is common for only the eigenvalue which is largest in modulus to be required, (often termed the *dominant eigenvalue* or *spectral radius*). We consider in this section one of the methods for obtaining this dominant eigenvalue, along with the corresponding eigenvector, its called the *"Power Method"*. Full details of other methods can be obtained from Wilkinson (1965).

The Power Method

Let us consider an n^{th} order matrix A which possesses the eigenvalues λ_i, i=1,2,...,n such that one of them is the dominant eigenvalue, λ_1, say, i.e.,

$$|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \dots \geq |\lambda_n| . \qquad (2.4.5)$$

By assumption, there exists n linearly independent eigenvectors $\underline{x}_1, \underline{x}_2, \dots, \underline{x}_n$, such that their linear combination can be expressed as an arbitrary vector $\underline{y}^{(0)}$, i.e.,

$$\underline{y}^{(0)} = \sum_{i=1}^{n} \alpha_i \underline{x}_i \, , \qquad (2.4.6)$$

where α_i, i=1,2,...,n, are constant coefficients, not all zero.

If we successively multiply $\underline{y}^{(0)}$ by matrix A, such that,

$$\underline{y}^{(1)} = A\underline{y}^{(0)} \, , \qquad (2.4.7)$$

$$\underline{y}^{(2)} = A\underline{y}^{(1)} = A^2\underline{y}^{(0)} \, , \qquad (2.4.8)$$

$$\vdots$$

$$\underline{y}^{(k)} = A\underline{y}^{(k-1)} = A^k\underline{y}^{(0)} \, , \qquad (2.4.9)$$

and for any eigenvalue λ_i we have from (2.4.1)

$$A\underline{x}_i = \lambda_i\underline{x}_i \quad , \; 1 \leq i \leq n \qquad (2.4.10)$$

hence

$$A^2\underline{x}_i = \lambda_i A\underline{x}_i = \lambda_i^2\underline{x}_i \, , \qquad (2.4.11)$$

$$\vdots$$

and

$$A^k\underline{x}_i = \lambda_i^{k-1} A\underline{x}_i = \lambda_i^k\underline{x}_i \, . \qquad (2.4.12)$$

Now, we have,

$$\underline{y}^{(k)} = A^k\underline{y}^{(0)} = A^k \sum_{i=1}^{n} \alpha_i\underline{x}_i = \sum_{i=1}^{n} \alpha_i A^k \underline{x}_i = \sum_{i=1}^{n} \alpha_i \lambda_i^k \underline{x}_i \, , \qquad (2.4.13)$$

or

$$y^{(k)} = \lambda_1^k\{\alpha_1\underline{x}_1 + \sum_{i=2}^{n} \alpha_i \left(\frac{\lambda_i}{\lambda_1}\right)^k \underline{x}_i\} \, , \qquad \alpha_1 \neq 0 \, . \qquad (2.4.14)$$

Since $\left|\frac{\lambda_i}{\lambda_1}\right|<1$, $i=2,3,\ldots,n$ from the initial assumption, therefore the terms $(\frac{\lambda_i}{\lambda_1})^k$, $i=2,3,\ldots,n$ will converge to zero as $k\to\infty$ so that:

$$\underline{y}^{(k)} \to \lambda_1^k \alpha_1 \underline{x}_1 \ , \tag{2.4.15}$$

hence the ratio between the j^{th} component, $1\leq j\leq n$ of $\underline{y}^{(k+1)}$ and $\underline{y}^{(k)}$ tends to λ_1, i.e.,

$$\lim_{k\to\infty} \frac{y_j^{(k+1)}}{y_j^{(k)}} = \lambda_1 \ . \tag{2.4.16}$$

and $y^{(k+1)}$ is the un-normalized eigenvector of A corresponding to λ_1.

It can be seen that the rate of convergence depends upon the ratio $|\lambda_2/\lambda_1|$ $(\lambda_2=\max_{2\leq i\leq n} |\lambda_i|)$ being very small, i.e., the smaller the values of this ratio the faster the convergence.

In practice, since λ_1 is not known until the end of the process, and to avoid the possibility of overflow, we always divide by the element of largest modulus in the vector produced. The algorithm can then be summarised as follows:

Let $\underline{y}_0$ be the initial vector.

Let $\underline{z}_1=A\underline{y}_0$, and let β_1 be the element of largest modulus in $\underline{z}_1$.

Define $\underline{y}_1 = \frac{1}{\beta_1}\underline{z}_1$.

Now the iteration proceeds with:

Step 1 $\quad \underline{z}^{(s+1)} = A\underline{y}^{(s)}$

Step 2 $\quad$ Choose $\beta^{(s+1)}$ = the element of largest modulus in $\underline{z}^{(s+1)}$

Step 3 $\quad \underline{y}^{(s+1)} = \frac{1}{\beta^{(s+1)}}\underline{z}^{(s+1)}$

Step 4 $\quad$ If $\underline{y}^{(s+1)}$ and $\underline{y}^{(s)}$ are sufficiently close, then terminate the procedure, otherwise repeat from Step 1.

The process should result in the values of $\underline{y}^{(s+1)}$ and $\beta^{(s+1)}$ as two good approximations to $\underline{x}_1$ and λ_1 respectively.

Definition 2.4.2

Let A be an (n×n) matrix with eigenvalues λ_i, $1 \leq i \leq n$, then,

$$\rho(A) = \max_i |\lambda_i| \quad , \tag{2.4.17}$$

is the *spectral radius* of A.

From Definition (2.3.4) it can be easily shown that for any (n×n) matrix A and any norm,

$$\rho(A) \leq ||A|| \quad . \tag{2.4.18}$$

Proof:

Let λ_i be an arbitrary eigenvalue of A and $\underline{x}_i$ its corresponding eigenvector, then,

$$A\underline{x}_i = \lambda_i \underline{x}_i \; .$$

and

$$|\lambda_i| \cdot ||\underline{x}_i|| = ||A\underline{x}_i||$$

$$\leq ||A|| \cdot ||\underline{x}_i|| \text{, for any compatible norm.}$$

Thus,

$$\lambda_i \leq ||A|| \; ,$$

since λ_i was arbitrarily chosen, hence, from Definition (2.4.2)

$$\rho(A) \leq ||A|| .$$

It can be shown, Graham (1979), Gantmakher (1959), that,

$$\mathrm{tr}(A) = \sum_{i=1}^{n} \lambda_i \; , \tag{2.4.19}$$

$$\det(A) = \prod_{i=1}^{n} \lambda_i \; . \tag{2.4.20}$$

Definition 2.4.3

Two matrices A and B of order n are *similar* if there exists a non-singular matrix P such that,

$$B = P^{-1}AP \; . \tag{2.4.21}$$

Matrix B is said to be obtained from matrix A by a similarity-transformation and if B is symmetric then P will in general be orthogonal ($P^{-1}=P^T$), and hence,

$$B = P^T AP \; . \tag{2.4.22}$$

The usefulness of such a transformation is that the eigenvalues of A and B are the same. This can be shown quite easily as follows: Let λ and $\underline{x}$ be the eigenvalue and the eigenvector of the matrix A respectively. Hence,

$$A\underline{x} = \lambda\underline{x} \ . \tag{2.4.23}$$

then premultiply by P^{-1}, we have,

$$P^{-1}A\underline{x} = \lambda P^{-1}\underline{x} \ . \tag{2.4.24}$$

Hence, if $\underline{y}=P^{-1}\underline{x}$, then,

$$\underline{x} = P\underline{y} \ . \tag{2.4.25}$$

Substituting (2.4.25) in (2.4.23) we get,

$$P^{-1}AP\underline{y} = \lambda\underline{y} \ , \tag{2.4.26}$$

or

$$B\underline{y} = \lambda\underline{y} \ , \tag{2.4.27}$$

thus λ is the eigenvalue of the matrix B and $\underline{y}$ is the associated eigenvector.

<u>Theorem 2.4.1</u> (Gerschgorin's Theorem)

If $A=[a_{i,j}]$ is an $(n\times n)$ matrix, then all the eigenvalues of A lie in the union of the discs,

$$|\lambda - a_{i,i}| \leq \sum_{\substack{j=1\\ j\neq i}}^{n} |a_{i,j}| , i=1,2,\ldots,n \tag{2.4.28}$$

<u>Proof:</u>

Let λ be an eigenvalue of A, and let $\underline{x}$ be the corresponding eigenvector. We can normalize $\underline{x}$ so that $\max_i|x_i|=1$. Hence, from (2.4.1)

$$\lambda\underline{x} = A\underline{x} \ . \tag{2.4.29}$$

$$\lambda x_i = \sum_{j=1}^{n} a_{i,j}x_j \ , \ 1\leq i\leq n \ , \tag{2.4.30}$$

i.e.

$$(\lambda-a_{i,i})x_i = \sum_{\substack{j=1\\ j\neq i}}^{n} a_{i,j}x_j \ , \ 1\leq i\leq n \ , \tag{2.4.31}$$

Now if $|x_k|=1$, then,

$$|\lambda - a_{k,k}| \leq \sum_{\substack{j=1\\ j\neq k}}^{n} |a_{k,j}| \ |x_j| \leq \sum_{\substack{j=1\\ j\neq k}}^{n} |a_{k,j}| \tag{2.4.32}$$

Thus, the eigenvalue λ lies in the disc D_r, say, and since λ is arbitrary, it follows that all the eigenvalues of A must lie in the union of the discs, i.e.,

$$|\lambda - a_{i,i}| \leqslant \sum_{\substack{j=1 \\ j \neq i}}^{n} |a_{i,j}| .$$

<u>Corollary 2.4.1</u>

If $A=[a_{i,j}]$ is an $(n \times n)$ matrix and we have,

$$v_1 = \max_{1 \leqslant i \leqslant n} \sum_{j=1}^{n} |a_{i,j}| , \tag{2.4.33}$$

$$v_2 = \max_{1 \leqslant j \leqslant n} \sum_{i=1}^{n} |a_{i,j}| , \tag{2.4.34}$$

then,

$$\rho(A) \leqslant \min(v_1, v_2) \ . \tag{2.4.35}$$

The condition (2.4.35) is a direct consequence of the fact that A and A^T have the same eigenvalues.

Let A be an $(n \times n)$ *tridiagonal* matrix,

$$A = \begin{bmatrix} a & b & & & \\ c & a & b & & \bigcirc \\ & \ddots & \ddots & \ddots & \\ & & c & a & b \\ \bigcirc & & & c & a \end{bmatrix} ,$$

where a,b and c may be real or complex numbers, the *eigenvalues of A* are given by,

$$\lambda_r = a + 2\sqrt{bc} \cos \frac{r\pi}{n+1} \quad , \ r=1,2,\ldots,n \tag{2.4.36}$$

(the proof may be found in Smith (1978) p.113).

<u>Theorem 4.2.4</u>

Let $A=[a_{i,j}]$ be an $(n \times n)$ strictly or irreducibly diagonally dominant complex matrix. Then, the matrix A is non-singular. If $a_{i,i}$ $1 \leqslant i \leqslant n$, are positive real numbers, then the eigenvalues λ_i of A satisfy,

$$\mathrm{Re}\{\lambda_i\} > 0 \quad , \quad 1 \leq i \leq n \ , \qquad (2.4.37)$$

Proof:

The proof can be obtained from Theorem (2.4.1) and given in Varga (1962), p.23.

2.5 POSITIVE DEFINITE MATRICES AND SPECIAL MATRICES

Definition 2.5.1

If a matrix A is Hermitian, and

$$(\underline{x},A\underline{x}) > 0 \;, \tag{2.5.1}$$

for all $\underline{x}\neq\underline{0}$, then A is *positive definite*.

A is non-negative definite if $(\underline{x},A\underline{x}) \geq 0$.

The following theorem, given without proof, can be used as an alternative definition of positive definiteness.

Theorem 2.5.1

Necessary and sufficient conditions for a Hermitian (or a real symmetric) matrix A to be positive definite is that, the eigenvalues of A are all positive.

Theorem 2.5.2

An irreducibly diagonally dominant matrix which is also symmetric and has positive real diagonal elements is positive definite.

This follows immediately from Theorems (2.4.2) and (2.5.1) since the eigenvalues of a symmetric matrix are real.

Definition 2.5.2

If $A=[a_{i,j}]$ is a real matrix of order n then A is said to be,

(i) *L-matrix* if

$$a_{i,i} > 0 \;,\; 1\leq i\leq n \;, \tag{2.5.2}$$

and

$$a_{i,j} \leq 0 \;,\; i\neq j,\; 1\leq i,j\leq n \;, \tag{2.5.3}$$

(ii) *Stieltjes matrix* if A is positive definite matrix and if (2.5.3) holds.

(iii) *M-matrix* if A is non-singular, if (2.5.3) holds and if $A^{-1} \geq 0$.

(Note: By $A \geq 0$ we mean that all elements of the matrix A are real and non-negative).

A simple (3×3) matrix which satisfies (i),(ii) and (iii) is,

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

2.6 CONVERGENCE OF SEQUENCES OF MATRICES

A sequence of matrices $A^{(1)}, A^{(2)}, A^{(3)}, \ldots$, of the same dimension is said to convergesto a matrix A (say) if and only if,

$$\lim_{k\to\infty} ||A-A^{(k)}|| = 0 . \qquad (2.6.1)$$

Definition 2.6.1

Let A be a square matrix, then A *converges* to zero if the sequence of matrices $A, A^2, A^3, \ldots$ converges to the null matrix, $\mathcal{O}$, and is divergent otherwise.

Theorem 2.6.1

If A is an $(n\times n)$ matrix, then,

$$\lim_{k\to\infty} A^k = \mathcal{O}, \quad \text{if } ||A|| < 1$$

Proof:

$$\begin{aligned} ||A^k|| = ||AA^{k-1}|| &\leq ||A||.||A^{k-1}|| \\ &\leq ||A||^2.||A^{k-2}|| \\ &\vdots \\ &\leq ||A||^k \end{aligned}$$

and since $||A||<1$. The result follows.

Theorem 2.6.2

If A is an $(n\times n)$ matrix, then A is convergent if and only if $\rho(A)<1$.

Proof:

See Varga (1962), p.13.

2.7 PROPERTY A AND CONSISTENTLY ORDERED MATRICES

Definition 2.7.1a

A matrix $A=[a_{i,j}]$ of order n has *Property A* if there exists two disjoint subsets S_1 and S_2 of $W=\{1,2,...,n\}$ such that if $i\neq j$ and if either $a_{i,j}\neq 0$ or $a_{j,i}\neq 0$, then $i \in S_1$ and $j \in S_2$ or else $i \in S_2$ and $j \in S_1$.

An alternative definition of Property A is given as follows:

Definition 2.7.1b

A matrix A of order n is said to have *Property A* if there exists a permutation matrix P such that PAP^T has the form,

$$PAP^T = \begin{bmatrix} D_1 & F \\ E & D_2 \end{bmatrix}$$

where D_1 and D_2 are square diagonal matrices.

Definition 2.7.2

A matrix A of order n is *consistently ordered* if for some t there exist disjoint subsets $S_1,S_2,...,S_t$ of $W=\{1,2,...,n\}$ such that $\sum_{k=1}^{t} S_k=W$ and such that if i and j are associated, then $j \in S_{k+1}$ if $j>i$ and $j \in S_{k-1}$ if $j<i$ where S_k is the subset containing i.

Theorem 2.7.1

If A is a T-matrix, then A is consistently ordered.

Proof: [See Young (1971), p.145].

Definition 2.7.3

A column vector $\underline{v}$ of order n with integer elements, is an *ordering vector* for the matrix A of order n if for any pair of associated integers i and j with $i \neq j$ we have $|v_i - v_j| = 1$.

Definition 2.7.4

An ordering vector $\underline{v}^T = (v_1, v_2, \ldots, v_n)$ for the matrix A of order n is a *compatible ordering vector* for A if:

(i) $v_i - v_j = 1$ if i and j are associated and $i > j$.

(ii) $v_i - v_j = -1$ if i and j are associated and $i < j$.

Theorem 2.7.2

There exists an ordering vector for a matrix A if and only if A has Property A. Moreover, if A is consistently ordered, then the matrix A has Property A.

Proof: [See Young (1971), p.148].

2.8 SOLUTION OF P.D.E.'S BY FINITE DIFFERENCE AND FINITE ELEMENT METHODS

As mentioned in Chapter 1, a number of approaches have been developed for the treatment of partial differential equations. The most widely used of these are the methods of finite elements and finite differences. Our main interest is the finite difference method which will be discussed in Section (2.8.2). Whilst in Section (2.8.1) a basic approach will be taken to introduce the finite element method.

2.8.1 FUNDAMENTALS OF THE FINITE ELEMENT METHOD

The finite element method is a mathematical procedure for approximating the solution to p.d.e.'s. It is commonly used in engineering problems, in particular civil, aeronautical and mechanical engineering, especially for the analysis of stress in solid components. Furthermore, it has been applied even to three-dimensional problems, such as the time-dependent problems involving fluid flow, heat transfer, magnetic field analysis,... etc. (Fenner (1975), Bathe and Wilson (1976)).

The finite element method can be applied to all of these problems with some minor modifications, and it is based on the idea of dividing the region R of the problem into a finite number of non-overlapping subregions or *elements*, R_e (say). The shape of these finite elements is generally simple polygons such as triangles and quadrilaterals in two-dimensions, pyramids and triangular and rectangular prisms in three-dimensions, the side of those elements may be curved. Further, the size of the elements and the degree of the interpolation polynomials can be varied readily. Small elements or higher order polynomials may be used in regions of rapid changes and large elements or low order polynomials may be used where changes are less severe.

In general, the finite element method consists of the following steps:

1. Choose the geometric shape of the subregions R_e in which R shall be divided.
2. Divide the region R into a large but finite number of elements.
3. Define the number and location of nodes that will be associated with this division of R into the $\{R_e\}$.
4. Derive an elemental coefficient matrix by using suitable physical principles and then by using these matrices one can assemble the total coefficient matrix.
5. Apply the appropriate boundary conditions to the coefficient matrix, then solve this modified matrix.

The Construction of the Basis Function

Triangular Elements

Let us divide the region R into a mesh of triangular elements as shown in Figure (2.8.1), let the mesh points on each element be the vertices of the triangles, where all the elements and mesh points are numbered. The typical (shaded) element is numbered m and its meshes are numbered a,b and c.

The finite element solution is then expressed as the function expansion

$$U(x,y) \simeq u(x,y) = \sum_{i=1}^{N} u_i\phi_i(x,y) \ , \qquad (2.8.1)$$

where N is the number of mesh points and $\phi_i(x,y)$, $i=1,2,\ldots,N$ are piecewise *basis functions*, and u_i are the corresponding mesh variables.

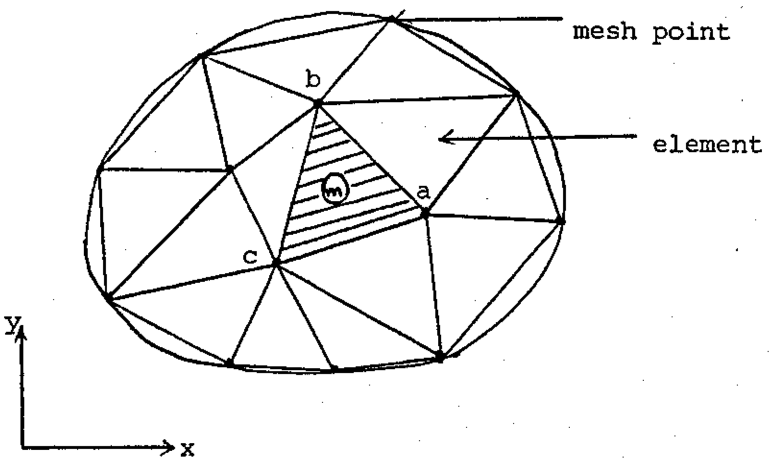


FIGURE (2.8.1): A two-dimensional solution domain divided into triangular finite elements

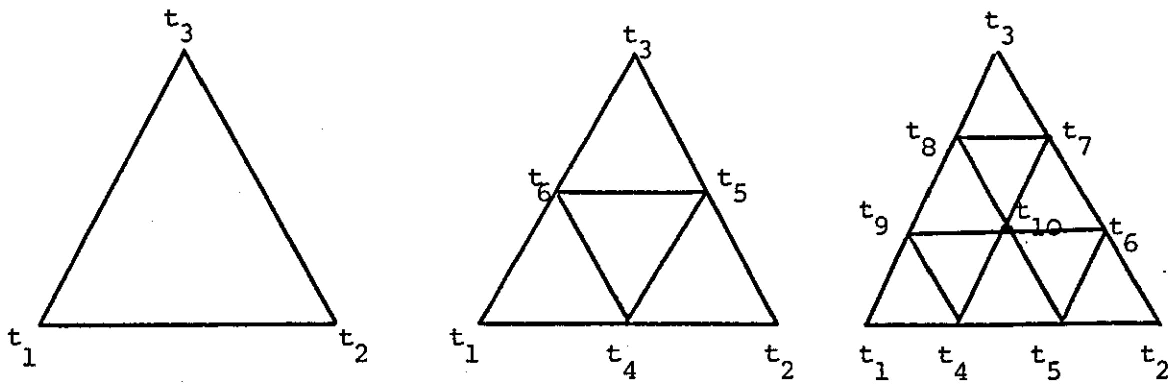
The basis functions are polynomials such that,

$$\phi_i(x_j,y_j) = \begin{cases} 1, & i=j, \\ 0, & i\neq j, \end{cases} \quad 1\leq i,j\leq N \ . \qquad (2.8.2)$$

To construct the basis functions many techniques are suggested in the literature such as Lagrange, Hermite interpolation formulae for polygonal regions (which can be divided into triangular elements). Let us take the two-dimensional case, the function $u(x,y)$ can be interpolated at $\frac{1}{2}(s+1)(s+2)$ points with a polynomial of order s, i.e.,

$$u(x,y) = \sum_{j=1}^{\frac{1}{2}(s+1)(s+2)} u_j\phi_j^{(s)}(x,y) \ . \qquad (2.8.3)$$

If the smallest element (the basis unit) is assumed to be the triangle $t_1t_2t_3$ (Fig. 2.8.2), then the polynomial (2.8.3) interpolates $u(x,y)$ at $\frac{1}{2}(s+1)(s+2)$ symmetrically placed points on the triangle $t_1t_2t_3$. For $s>1$, the non-vertex points can be obtained geometrically by dividing each side of the triangle $t_1t_2t_3$ into s equal segments and by joining the points of subdivision by lines drawn parallel to the sides of the triangle (see Fig. (2.8.2) as an example for s=2,3).



(FIGURE 2.8.2)

For s=1 (the linear case). The polynomial in each element has the form,

$$u(x,y) = \alpha_1+\alpha_2x+\alpha_3y \ , \tag{2.8.4}$$

where α_1,α_2 and α_3 are parameters defined for each element, and u_j, j=1,2,3 are the values of u(x,y) at the vertices t_j. Thus,

$$\begin{aligned} u_1 &= \alpha_1+\alpha_2x_1+\alpha_3y_1 \ , \\ u_2 &= \alpha_1+\alpha_2x_2+\alpha_3y_2 \ , \\ u_3 &= \alpha_1+\alpha_2x_3+\alpha_3y_3 \ . \end{aligned} \tag{2.8.5}$$

Solving equations (2.8.5) for α_1,α_2 and α_3 and substituting the result in (2.8.4), we obtain,

$$u(x,y) = u_1\phi_1^{(1)}(x,y)+u_2\phi_2^{(1)}(x,y)+u_3\phi_3^{(1)}(x,y) \ , \tag{2.8.6}$$

where,

$$\phi_1^{(1)} = \frac{1}{D} \cdot \det \begin{bmatrix} 1 & x & y \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix} \equiv P_1 \ ,$$

$$\phi_2^{(1)} = \frac{1}{D} \cdot \det \begin{bmatrix} 1 & x & y \\ 1 & x_1 & y_1 \\ 1 & x_3 & y_3 \end{bmatrix} \equiv P_2 \ , \tag{2.8.7}$$

and

$$\phi_3^{(1)} = \frac{1}{D} \cdot \det \begin{bmatrix} 1 & x & y \\ 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \end{bmatrix} \equiv P_3$$

where,

$$D = \det \begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix}$$

(D is twice the area of the triangle $t_1t_2t_3$). It is easily seen that,

$$P_i(x_j,y_j) = \begin{cases} 1, & (i=j) \\ 0, & (i\neq j) \end{cases}, \quad (1\leq i,j\leq 3)$$

<u>For s=2 (the quadratic case)</u>, we have,

$$u(x,y) = \alpha_1+\alpha_2x+\alpha_3y+\alpha_4x^2+\alpha_5xy+\alpha_6y^2$$

$$= \sum_{j=1}^{6} u_j\phi_j^{(2)}(x,y) , \qquad (2.8.8)$$

where u_j, $(j=1,2,\ldots,6)$ are the values of $u(x,y)$ at the vertices of the triangle together with the values at the mid-points of the triangle (Figure 2.8.2(b)). The functions $\phi_j^{(2)}(x,y)$ $(j=1,2,\ldots,6)$ are given by

$$\left.\begin{aligned} \phi_k^{(2)} &= P_k(2P_k-1) , \quad k=1,2,3 \\ \phi_4^{(2)} &= 4P_1P_2 \\ \phi_5^{(2)} &= 4P_2P_3 \\ \phi_6^{(2)} &= 4P_3P_1 \end{aligned}\right\} \qquad (2.8.9)$$

where P_1,P_2 and P_3 are given in (2.8.7).

<u>For s=3 (the cubic case)</u>, we have,

$$u(x,y) = \alpha_1+\alpha_2x+\alpha_3y+\alpha_4x^2+\alpha_5xy+\alpha_6y^2+\alpha_7x^3+\alpha_8x^2y+\alpha_9xy^2+\alpha_{10}y^3$$

$$= \sum_{j=1}^{10} u_j\phi_j^{(3)}(x,y) , \qquad (2.8.10)$$

where $\phi_j^{(3)}(x,y)$ $(j=1,2,\ldots,10)$ refer to the basis functions at the mesh points t_j, $j=1,2,\ldots,10$ in Fig. (2.8.2(c)) and are determined as follows:

$$\phi_k^{(3)} = \tfrac{1}{2}P_k(3P_k-1)(3P_k-2) \quad , \; k=1,2,3,$$

$$\phi_4^{(3)} = \frac{9}{2}P_1P_2(3P_1-1) \quad ,$$

$$\phi_5^{(3)} = \frac{9}{2}P_1P_2(3P_2-1) \quad .$$

Similarly, $\phi_6^{(3)},\phi_7^{(3)}$ can be expressed in terms of P_2,P_3 and $\phi_8^{(3)},\phi_9^{(3)}$ in terms of P_3,P_1, and

$$\phi_{10}^{(3)} = 27P_1P_2P_3 \; .$$

$u_{10}(x,y)$ can be eliminated as follows,

$$u_{10}(x,y) = \frac{1}{4}\sum_{j=4}^{9} u_j - \frac{1}{6}\sum_{j=1}^{3} u_j \; .$$

In general, the r^{th} degree piecewise polynomial of the form,

$$u(x,y) = \sum_{k+\ell=0}^{s} \alpha_{k\ell}x^k y^\ell \; , \tag{2.8.11}$$

can be constructed if we choose to locate symmetrically $\tfrac{1}{2}(s+1)(s+2)$ nodes in a triangular element.

Rectangular Elements

Let us consider the rectangular element abcd with sides parallel to the x and y directions (Fig. 2.8.3)

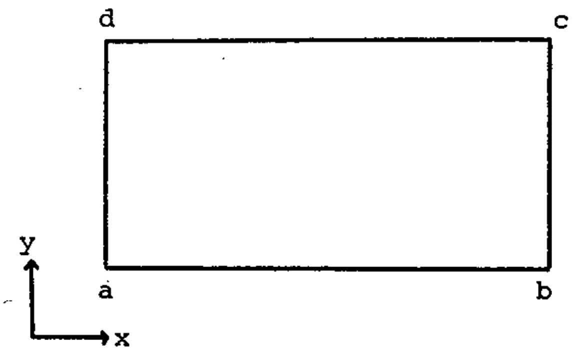


FIGURE (2.8.3)

The piecewise polynomial is of the form,

$$u(x,y) = \alpha_1+\alpha_2x+\alpha_3y+\alpha_4xy \ , \qquad (2.8.12)$$

where α_i (i=1,2,3,4) are defined for each element, and u_i (i=1,2,3,4) are the values of u(x,y) at the vertices a,b,c and d respectively.

Hence,

$$\left.\begin{aligned} u_1 &= \alpha_1+\alpha_2x_a+\alpha_3y_a+\alpha_4x_ay_a \ , \\ u_2 &= \alpha_1+\alpha_2x_b+\alpha_3y_b+\alpha_4x_by_b \ , \\ u_3 &= \alpha_1+\alpha_2x_c+\alpha_3y_c+\alpha_4x_cy_c \ , \\ u_4 &= \alpha_1+\alpha_2x_d+\alpha_3y_d+\alpha_4x_dy_d \end{aligned}\right\} \qquad (2.8.13)$$

By solving equations (2.8.13) for $\alpha_1,\alpha_2,\alpha_3$ and α_4 and substituting the result in (2.8.12) we obtain

$$u(x,y) = u_1\phi_1(x,y)+u_2\phi_2(x,y) +u_3\phi_3(x,y) + u_4\phi_4(x,y) \ , \qquad (2.8.14)$$

where $\phi_i(x,y)$, i=1,2,3,4, are the basis functions at the corners a,b,c and d respectively. The basis functions are found to be,

$$\left.\begin{aligned} \phi_1(x,y) &= \frac{(x-x_b)(y-y_d)}{(x_a-x_b)(y_a-y_d)} \\ \phi_2(x,y) &= \frac{(x-x_a)(y-y_c)}{(x_b-x_a)(y_b-y_c)} \\ \phi_3(x,y) &= \frac{(x-x_d)(y-y_b)}{(x_c-x_d)(y_c-y_b)} \\ \phi_4(x,y) &= \frac{(x-x_c)(y-y_a)}{(x_d-x_c)(y_d-y_a)} \end{aligned}\right\} \qquad (2.8.15)$$

The basis function $\phi_1(x,y)$ is unity at a and zero at b,c and d, it is linear along the sides ab and ad and zero along the sides bc and cd.

Further discussion along these lines and the construction of higher degree basis functions and other possible alternative interpolating schemes can be found in Mitchell and Wait (1977) or Strang and Fix (1973).

2.8.2 Finite Difference Approximation to Derivatives

(I) The Two Dimensional Case

Consider the Dirichlet problem for the linear self adjoint elliptic equation,

$$\frac{\partial}{\partial x}(A(x,y)\frac{\partial U}{\partial x}) + \frac{\partial}{\partial y}(B(x,y)\frac{\partial U}{\partial y}) - F(x,y)U = G(x,y), \quad (x,y) \in R, \tag{2.8.16}$$

$$U(x,y) = g(x,y), \quad (x,y) \in \partial R, \tag{2.8.17}$$

defined for a bounded region R, where $A(x,y)>0$, $B(x,y)>0$ and $F(x,y)\geq 0$.

The strategy of the finite difference method is based on the replacement of the continuous problem (differential equations) to a discrete mathematical model (difference equations) which are suitable for solution on a high-speed computer.

Suppose that the region under consideration $\bar{R}$, $\bar{R}=R \cup \partial R$, lie in the space co-ordinates x & y, Fig.(2.8.4) and is covered by two sets of lines parallel to the x- and y-axis. The intersections of these lines are called the (unknown) *grid points* (Other names used in the literature are *nodal*, *mesh*, *net*, *pivotal* or *lattice* points). At each grid point, internal to ∂R, we find an approximation to the differential equation in terms of the function values at the grid point itself and other certain neighbouring grid points and boundary points. This always leads to a set of algebraic equations (linear, if the original differential equation is linear). The solution of this set of equations is then an approximate solution to the partial differential equation and will be referred to by u.

Let h_x and h_y be the grid spacings in the x- and y-direction respectively (see Figure (2.8.4)) and let x_i and y_j be the coordinates of a typical internal grid point, where,

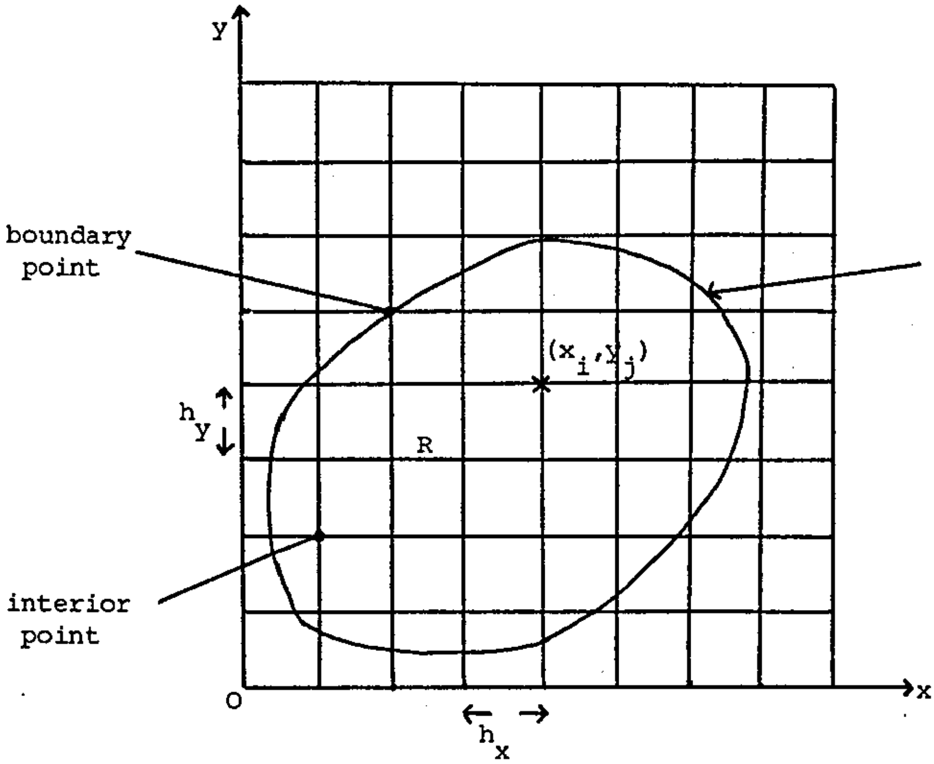


FIGURE (2.8.4)

$$x_i = ih_x \quad , \; i=0,1,2,3,\ldots,N$$

$$y_j = jh_y \quad , \; j=0,1,2,3,\ldots,M.$$

Further, assume that R is a square or rectangular region, in which the grid point (x_0,y_0) coincide with the origin, and the values of the function $U(x,y)$ are approximated by $u_{i,j}=u(x_i,y_j)$. To obtain a finite difference replacement of equation (2.8.16), define $D_x = \frac{\partial}{\partial x}$ and $D_y = \frac{\partial}{\partial y}$. The exact formulae which connect D_x, D_y with δ_x and δ_y respectively can be written as [Hildebrand (1974), Mitchell (1976)]

$$D_x = \frac{2}{h_x}\sinh^{-1}\frac{\delta_x}{2} = \frac{1}{h_x}\left(\delta_x - \frac{1^2}{2^2.3!}\delta_x^3 + \frac{1^2.3^2}{2^4.5!}\delta_x^5 \ldots\right), \tag{2.8.18}$$

$$D_y = \frac{2}{h_y}\sinh^{-1}\frac{\delta_y}{2} = \frac{1}{h_y}\left(\delta_y - \frac{1^2}{2^2.3!}\delta_y^3 + \frac{1^2.3^2}{2^4.5!}\delta_y^5 \ldots\right), \tag{2.8.19}$$

where δ_x and δ_y are the central difference operators in the x- and y-direction respectively, i.e.,

$$\delta_x u_{i,j} = u_{i+\frac{1}{2},j} - u_{i-\frac{1}{2},j} \text{ and } \delta_y u_{i,j} = u_{i,j+\frac{1}{2}} - u_{i,j-\frac{1}{2}} ,$$

$$(u_{i,j} \equiv u(x_i,y_j))$$

which obviously, gives,

$$\delta_x^2 u_{i,j} = u_{i-1,j} - 2u_{i,j} + u_{i+1,j} \text{ and } \delta_y^2 u_{i,j} = u_{i,j-1} - 2u_{i,j} + u_{i,j+1}$$

and so on for $\delta_x^3, \delta_y^3, \ldots$.

From equation (2.8.18),

$$D_x^2 = \frac{1}{h_x^2}(\delta_x^2 - \frac{1}{12}\delta_x^4 + \frac{1}{90}\delta_x^6 \ldots) , \tag{2.8.20}$$

similarly,

$$D_y^2 = \frac{1}{h_y^2}(\delta_y^2 - \frac{1}{12}\delta_y^4 + \frac{1}{90}\delta_y^6 \ldots) . \tag{2.8.21}$$

Hence, from equation (2.8.16) by using difference replacement and by ignoring the terms involving δ of order greater than two in equations (2.8-18,19,20 and 21) we get,

$$\frac{1}{h_x^2}\delta_x(A_{i,j}\delta_x u_{i,j}) + \frac{1}{h_y^2}\delta_y(B_{i,j}\delta_y u_{i,j}) - F_{i,j}u_{i,j} = G_{i,j} . \tag{2.8.22}$$

By simple calculation we write equation (2.8.22) as,

$$\frac{1}{h_x^2}[A_{i+\frac{1}{2},j}(u_{i+1,j} - u_{i,j}) - A_{i-\frac{1}{2},j}(u_{i,j} - u_{i-1,j})] + \frac{1}{h_y^2}[B_{i,j+\frac{1}{2}}(u_{i,j+1} - u_{i,j}) - B_{i,j-\frac{1}{2}}(u_{i,j} - u_{i,j-1})] - F_{i,j}u_{i,j} = G_{i,j} , \tag{2.8.23}$$

$$(A_{i+\frac{1}{2},j} \equiv A(x_i+\tfrac{1}{2}h_x,y_j), \quad B_{i,j+\frac{1}{2}} \equiv B(x_i,y_j+\tfrac{1}{2}h_y)).$$

In practice, it is common to have $\underline{h_x = h_y \equiv h}$, resulting in considerable simplification, whence,

$$\frac{1}{h^2}[A_{i+\frac{1}{2},j}u_{i+1,j} + A_{i-\frac{1}{2},j}u_{i-1,j} + B_{i,j+\frac{1}{2}}u_{i,j+1} + B_{i,j-\frac{1}{2}}u_{i,j-1} - (A_{i+\frac{1}{2},j} + A_{i-\frac{1}{2},j} + B_{i,j+\frac{1}{2}} + B_{i,j-\frac{1}{2}} + h^2F_{i,j})u_{i,j}] = G_{i,j} , \tag{2.8.24}$$

which may be written in the form,

$$\alpha_1 u_{i+1,j}+\alpha_2 u_{i-1,j}-\alpha_0 u_{i,j}+\alpha_3 u_{i,j+1}+\alpha_4 u_{i,j-1} = h^2 G_{i,j} \, , \qquad (2.8.25)$$

where $\alpha_1 = A_{i+\frac{1}{2},j}$, $\alpha_2 = A_{i-\frac{1}{2},j}$, $\alpha_3 = B_{i,j+\frac{1}{2}}$, $\alpha_4 = B_{i,j-\frac{1}{2}}$

and $\alpha_0 = \alpha_1+\alpha_2+\alpha_3+\alpha_4+h^2F_{i,j}$,

or

$$\alpha_1 u(x+h,y)+\alpha_2 u(x-h,y)-\alpha_0 u(x,y)+\alpha_3 u(x,y+h)+\alpha_4 u(x,y-h) = h^2 G(x,y), \qquad (2.8.26)$$

where $\alpha_1 = A(x+\frac{h}{2},y)$, $\alpha_2 = A(x-\frac{h}{2},y)$, $\alpha_3 = B(x,y+\frac{h}{2})$,

$\alpha_4 = B(x,y-\frac{h}{2})$ and $\alpha_0 = \sum_{i=1}^{4} \alpha_i + h^2F(x,y)$.

At this stage it is instructive to establish a criterion for the local accuracy for formula (2.8.26), let $z_{i,j}$ be the difference between the exact solutions of the differential and difference equations at the grid point (i,j), i.e.,

$$z_{i,j} = U_{i,j} - u_{i,j} \, . \qquad (2.8.27)$$

By using Taylor's theorem and for simplicity let us take $U=U(x,y)$, $A=A(x,y)$, $B=B(x,y)$, $F=F(x,y)$ and $G=G(x,y)$, we have,

$$U(x\pm h,y) = U \pm h\frac{\partial U}{\partial x} + \frac{h^2}{2}\frac{\partial^2 U}{\partial x^2} \pm \frac{h^3}{6}\frac{\partial^3 U}{\partial x^3} + \frac{h^4}{24}\frac{\partial^4 U}{\partial x^4} \pm \frac{h^5}{120}\frac{\partial^5 U}{\partial x^5} \cdots$$

$$U(x,y\pm h) = U \pm h\frac{\partial U}{\partial y} + \frac{h^2}{2}\frac{\partial^2 U}{\partial y^2} \pm \frac{h^3}{6}\frac{\partial^3 U}{\partial y^3} + \frac{h^4}{24}\frac{\partial^4 U}{\partial y^4} \pm \frac{h^5}{120}\frac{\partial^5 U}{\partial y^5} \cdots$$

$$A(x\pm\frac{h}{2},y) = A \pm \frac{h}{2}\frac{\partial A}{\partial x} + \frac{h^2}{8}\frac{\partial^2 A}{\partial x^2} \pm \frac{h^3}{48}\frac{\partial^3 A}{\partial x^3} + \frac{h^4}{16\text{x}24}\frac{\partial^4 A}{\partial x^4} \pm \frac{h^5}{32\text{x}120}\frac{\partial^5 A}{\partial x^5} \cdots$$

$$B(x,y\pm\frac{h}{2}) = B \pm \frac{h}{2}\frac{\partial B}{\partial y} + \frac{h^2}{8}\frac{\partial^2 B}{\partial y^2} \pm \frac{h^3}{48}\frac{\partial^3 B}{\partial y^3} + \frac{h^4}{16\text{x}24}\frac{\partial^4 B}{\partial y^4} \pm \frac{h^5}{32\text{x}120}\frac{\partial^5 B}{\partial y^5} \cdots$$

hence,

$$\begin{aligned}
&\alpha_1 U(x+h,y)+\alpha_2 U(x-h,y)+\alpha_3 U(x,y+h)+\alpha_4 U(x,y-h) = \\
&2AU+2BU+h^2\left(A\frac{\partial^2 U}{\partial x^2} + \frac{\partial A}{\partial x}\frac{\partial U}{\partial x} + \frac{1}{4}\frac{\partial^2 A}{\partial x^2}U + B\frac{\partial^2 U}{\partial y^2} + \frac{\partial B}{\partial y}\frac{\partial U}{\partial y} + \frac{1}{4}\frac{\partial^2 B}{\partial y^2}U\right) \\
&+h^4\left(\frac{1}{12}A\frac{\partial^4 U}{\partial x^4} + \frac{1}{6}\frac{\partial A}{\partial x}\frac{\partial^3 U}{\partial x^3} + \frac{1}{8}\frac{\partial^2 A}{\partial x^2}\frac{\partial^2 U}{\partial x^2} + \frac{1}{24}\frac{\partial^3 A}{\partial x^3}\frac{\partial U}{\partial x} + \frac{1}{12}B\frac{\partial^4 U}{\partial x^4} + \frac{1}{6}\frac{\partial B}{\partial y}\right. \\
&\qquad \left.\frac{\partial^3 U}{\partial y^3} + \frac{1}{8}\frac{\partial^2 B}{\partial y^2}\frac{\partial^2 U}{\partial y^2} + \frac{1}{24}\frac{\partial^3 B}{\partial y^3}\frac{\partial U}{\partial y}\right) + O(h^6) \, ,
\end{aligned}$$

and
$$\sum_{i=1}^{4}\alpha_i U = 2AU+2BU + \frac{h^2}{4}\left(\frac{\partial^2 A}{\partial x^2}U + \frac{\partial^2 B}{\partial y^2}U\right) + \frac{h^4}{192}\left(\frac{\partial^4 A}{\partial x^4}U + \frac{\partial^4 B}{\partial y^4}U\right)+O(h^6)$$

and so,

$$\alpha_1 U(x+h,y)+\alpha_2 U(x-h,y)-\alpha_0 U+\alpha_3 U(x,y+h)+\alpha_4 U(x,y-h)-h^2G =$$
$$-h^2(FU+G)+h^2\left(A\frac{\partial^2 U}{\partial x^2} + \frac{\partial A}{\partial x}\frac{\partial U}{\partial x} + B\frac{\partial^2 U}{\partial y^2} + \frac{\partial B}{\partial y}\frac{\partial U}{\partial y}\right) + h^4\left[\frac{1}{12}\left(A\frac{\partial^4 U}{\partial x^4} + B\frac{\partial^4 U}{\partial y^4}\right)+\right.$$
$$\frac{1}{6}\left(\frac{\partial A}{\partial x}\frac{\partial^3 U}{\partial x^3} + \frac{\partial B}{\partial y}\frac{\partial^3 U}{\partial y^3}\right) + \frac{1}{8}\left(\frac{\partial^2 A}{\partial x^2}\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 B}{\partial y^2}\frac{\partial^2 U}{\partial y^2}\right) + \frac{1}{24}\left(\frac{\partial^3 A}{\partial x^3}\frac{\partial U}{\partial x} + \frac{\partial^3 B}{\partial y^3}\frac{\partial U}{\partial y}\right)-$$
$$\left.\frac{1}{192}\left(\frac{\partial^4 A}{\partial x^4}U + \frac{\partial^4 B}{\partial y^4}U\right)\right] + O(h^6) \ . \tag{2.8.28}$$

Then from equations (2.8.26), (2.8.27) and (2.8.28), we obtain,

$$\alpha_1 z(x+h,y)+\alpha_2 z(x-h,y)-\alpha_0 z+\alpha_3 z(x,y+h)+\alpha_4 z(x,y-h) = -h^2(FU+G)+$$
$$h^2\left(A\frac{\partial^2 U}{\partial x^2} + \frac{\partial A}{\partial x}\frac{\partial U}{\partial x} + B\frac{\partial^2 U}{\partial y^2} + \frac{\partial B}{\partial y}\frac{\partial U}{\partial y}\right) + h^4\left[\frac{1}{12}\left(A\frac{\partial^4 U}{\partial x^4} + B\frac{\partial^4 U}{\partial y^4}\right) + \frac{1}{6}\left(\frac{\partial A}{\partial x}\frac{\partial^3 U}{\partial x^3} +\right.\right.$$
$$\left.\frac{\partial B}{\partial y}\frac{\partial^3 U}{\partial y^3}\right) + \frac{1}{8}\left(\frac{\partial^2 A}{\partial x^2}\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 B}{\partial y^2}\frac{\partial^2 U}{\partial y^2}\right) + \frac{1}{24}\left(\frac{\partial^3 A}{\partial x^3}\frac{\partial U}{\partial x} + \frac{\partial^3 B}{\partial y^3}\frac{\partial U}{\partial y}\right) - \frac{1}{192}\left(\frac{\partial^4 A}{\partial x^4}U +\right.$$
$$\left.\left.\frac{\partial^4 B}{\partial y^4}U\right] + O(h^6) \ . \tag{2.8.29}$$

The terms on the right hand side of formula (2.8.29), excluding $-h^2(FU+G)$, are defined as the *"local truncation error"* of formula (2.8.26), and it is of $O(h^2)$, while the term $h^2\left(A\frac{\partial^2 U}{\partial x^2} + \frac{\partial A}{\partial x}\frac{\partial U}{\partial x} + B\frac{\partial^2 U}{\partial y^2} + \frac{\partial B}{\partial y}\frac{\partial U}{\partial y}\right)$ is defined as the *"principal part of the truncation error"*.

The local truncation error is usually neglected, so if we scan over the grid points with formula (2.8.26), a set of simultaneous equations can be obtained whose solution $\{u_{i,j}\}$ is a finite difference approximation to the exact solution $\{U_{i,j}\}$ at the internal grid points.

If the boundary values are assumed to be known for i=0, i=N, j=0

$j=M$, the above set of equations can be written out (for $i=1,2,\ldots,N-1$ and $j=1,2,\ldots,M-1$) in matrix form as,

$$A\underline{u} = \underline{b} \ , \tag{2.8.30}$$

where $\underline{b}$ is a column vector composed of the known values $h^2G_{i,j}$ plus values of the finite difference approximate solution $u_{i,j}$ given on the boundary ∂R, i.e.,

$$\begin{aligned}\underline{b} = (&h^2G_{1,1}-\alpha_3u_{0,1}-\alpha_4u_{1,0}, h^2G_{1,2}-\alpha_3u_{0,2},\ldots,h^2G_{1,M}-\alpha_3u_{0,M}-\alpha_2u_{1,M},\\ &h^2G_{2,1}-\alpha_4u_{2,0}, h^2G_{2,2}, h^2G_{2,3},\ldots,h^2G_{2,M}-\alpha_2u_{2,M},\ldots,h^2G_{N,1}-\alpha_4u_{N,0}\\ &-\alpha_1u_{0,M}, h^2G_{N,2}-\alpha_1u_{1,M},\ldots,h^2G_{N,M}-\alpha_1u_{N,M}-\alpha_2u_{N,M})^T \ ,\end{aligned}$$

and,

$$\begin{aligned}\underline{u} = (&u_{1,1},u_{1,2},\ldots,u_{1,M-1},u_{2,1},u_{2,2},\ldots,u_{2,M-1},\ldots,u_{N-1,1},u_{N-1,2},\\ &\ldots,u_{N-1,M-1})^T \ ,\end{aligned}$$

the two column vectors $\underline{u}$ and $\underline{b}$ are of size $(N-1)\times(M-1)$ and the matrix A of order $(N-1)\times(M-1)$ has the tridiagonal *block* form,

$$A = \begin{bmatrix} D & -C & & & \\ -B & D & -C & & O \\ & \ddots & \ddots & \ddots & \\ O & & -B & D & -C \\ & & & -B & D \end{bmatrix} \tag{2.8.31}$$

where,

$$D = \begin{bmatrix} \alpha_0 & -\alpha_2 & & & \\ -\alpha_4 & \alpha_0 & -\alpha_2 & & O \\ & \ddots & \ddots & \ddots & \\ O & & -\alpha_4 & \alpha_0 & -\alpha_2 \\ & & & -\alpha_4 & \alpha_0 \end{bmatrix} \ , \tag{2.8.32}$$

and the two matrices B and C are given by α_3I and α_1I respectively (I is the unit matrix).

The equations of (2.8.30) are assumed to be ordered row-wise or column-wise.

For a rectangular mesh of M×N points, the coefficient matrix A is a square matrix of order (M-1)(N-1) and can be shown to have the following properties:

(i) $a_{i,i}>0$ and $a_{i,j}\leq 0$, for all i,j with $i\neq j$.

(ii) Diagonally dominant

(iii) Irreducible.

Properties (ii) and (iii) followsfrom Definition (2.2.1) and Theorem (2.2.3). It also follows from Theorem (2.2.4) that A is non-singular and hence system (2.8.30) has a unique solution. If all the interior mesh points in the region R are regular, then A will be symmetric and hence by Theorem (2.5.2) is positive definite.

In problems where the boundary of the region is curved, we have a set of mesh points adjacent to the boundary which require special treatment. Consider the general case of a group of five points whose spacing is non-uniform, we represent each distance by s_ih, (i=1,2,3,4), where s_i is the fraction of the standard spacing h that the particular distance represents, Fig. (2.8.5).

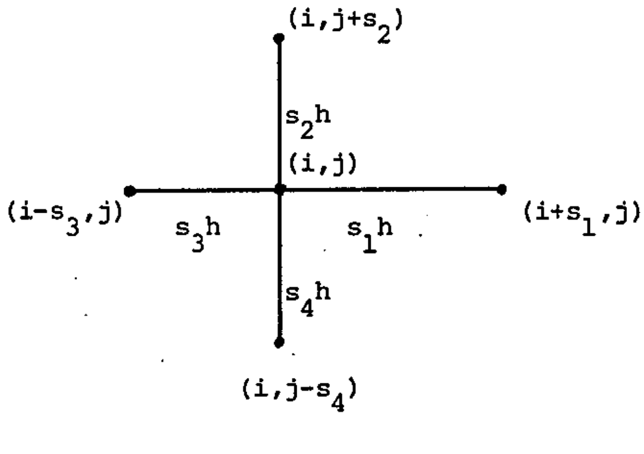


FIGURE (2.8.5)

Hence, in order to derive the finite difference approximation of (2.8.16) for an irregular region and denoting $h_x=h_y=h$, we can write the central difference operators in the x and y directions as,

$$\delta_x u_{i,j} = u_{i+\frac{1}{2}s_1,j} - u_{i-\frac{1}{2}s_3,j} ,$$

$$\delta_y u_{i,j} = u_{i,j+\frac{1}{2}s_2} - u_{i,j-\frac{1}{2}s_4} ,$$

and hence the finite difference approximations,

$$[\frac{\partial}{\partial x}(A(x,y)\frac{\partial U}{\partial y}]_{i,j} \approx [A_{i+\frac{1}{2}s_1,j}(\frac{u_{i+s_1,j}-u_{i,j}}{s_1 h}) - A_{i-\frac{1}{2}s_3,j}(\frac{u_{i,j}-u_{i-s_3,j}}{s_3 h})]/\frac{h}{2}(s_1+s_3) , \quad (2.8.33)$$

and

$$[\frac{\partial}{\partial y}(B(x,y)\frac{\partial U}{\partial y}]_{i,j} \approx [B_{i,j+\frac{1}{2}s_2}(\frac{u_{i,j+s_2}-u_{i,j}}{s_2 h}) - B_{i,j-\frac{1}{2}s_4}(\frac{u_{i,j}-u_{i,j-s_4}}{s_4 h})]/\frac{h}{2}(s_2+s_4) \quad (2.8.34)$$

The resulting finite difference equation at such a mesh point becomes,

$$\beta_1 u_{i+s_1,j}+\beta_2 u_{i,j+s_2}+\beta_3 u_{i-s_3,j}+\beta_4 u_{i,j-s_4}-\beta_0 u_{i,j} = \frac{1}{2} h^2 G_{i,j} , \quad (2.8.35)$$

where,

$$\beta_1 = \frac{A_{i+\frac{1}{2}s_1,j}}{s_1(s_1+s_3)} , \quad \beta_2 = \frac{B_{i,j+\frac{1}{2}s_2}}{s_2(s_2+s_4)} ,$$

$$\beta_3 = \frac{A_{i-\frac{1}{2}s_3,j}}{s_3(s_1+s_3)} , \quad \beta_4 = \frac{B_{i,j-\frac{1}{2}s_4}}{s_4(s_2+s_4)} , \quad (2.8.36)$$

and

$$\beta_0 = \sum_{i=1}^{4} \beta_i + \frac{1}{2} h^2 F_{i,j} .$$

Where $s_1=s_2=s_3=s_4=1$, the difference equation (2.8.35) reduces to (2.8.36).

(II) The Three Dimensional Case

Consider now a class of problems defined by the linear self-adjoint elliptic equation in three space dimensions, namely:

$$\frac{\partial}{\partial x}(A(x,y,z)\frac{\partial U}{\partial x}) + \frac{\partial}{\partial y}(B(x,y,z)\frac{\partial U}{\partial y}) + \frac{\partial}{\partial z}(C(x,y,z)\frac{\partial U}{\partial z}) - F(x,y,z) = G(x,y,z),$$

$$(x,y,z) \in R, \qquad (2.8.37)$$

$$U(x,y,z) = g(x,y,z), \quad (x,y,z) \in \partial R, \qquad (2.8.38)$$

defined for a bounded volumetric region R, where A,B and C are strictly positive functions and $F(x,y,z) \geq 0$.

The volume under consideration $\bar{R} \equiv R \cup \partial R$, is covered by a volumetric grid system, with spacings h_x, h_y, h_z in the x, y and z-directions respectively. Assume that (i,j,k) denotes the grid point (x_i, y_j, z_k) and $u_{i,j,k}$ denotes $u(x_i, y_j, z_k)$, where $x_i = ih_x$, $y_j = jh_y$ and $z_k = kh_z$, $0 \leq i,j,k \leq N$.

A finite difference replacement of equation (2.8.37) can be obtained by applying similar steps as in the two dimensional case and by assuming that $h_x = h_y = h_z = h$, the following finite difference equation is obtained,

$$\alpha_1 u_{i+1,j,k} + \alpha_2 u_{i-1,j,k} + \alpha_3 u_{i,j+1,k} + \alpha_4 u_{i,j-1,k} + \alpha_5 u_{i,j,k+1} + \alpha_6 u_{i,j,k-1} - \alpha_0 u_{i,j,k} = h^2 G_{i,j}, \qquad (2.8.39)$$

where,

$$\alpha_1 = A_{i+\frac{1}{2},j,k}, \quad \alpha_2 = A_{i-\frac{1}{2},j,k}, \quad \alpha_3 = B_{i,j+\frac{1}{2},k},$$

$$\alpha_4 = B_{i,j-\frac{1}{2},k}, \quad \alpha_5 = C_{i,j,k+\frac{1}{2}}, \quad \alpha_6 = C_{i,j,k-\frac{1}{2}},$$

and $$\alpha_0 = \sum_{\ell=1}^{6} \alpha_\ell + h^2 F_{i,j,k}.$$

Under the same consideration as in the two-dimensional case, the obtained set of difference equations may be expressed as the system (2.8.30), the coefficient matrix A being a sparse, real, seven-diagonal, square matrix with the properties given on page (53).

2.9 SYMBOLIC COMPUTATION - AN INTRODUCTION TO REDUCE

The purpose of this section is to introduce the new tool of symbolic algebraic computation (or symbolic computation, for short) and point out its potential in the numerical modelling and simulation of field problems.

Symbolic computation refers to the technique of manipulating on a computer, symbolic expressions that may not necessarily have numerical values. Therefore, techniques of symbolic computation can be used, among other things, to perform algebraic manipulations of mathematical formulae. Crudely, one can think of symbolic computation as a computerized version of the traditional "paper and pencil" manipulations of algebraic expressions commonly arising in applied mathematics. Therefore, symbolic computation can significantly reduce the tedium of analytic calculations and increase their reliability. This capability permits one to carry on the analytic calculations before numerical computations start.

A number of symbolic manipulation systems suitable for manipulating algebraic expressions have been developed over the past few years. A representative sample of these are listed as follows:

ALPAK	CONFORM	SAC-1
ALTRAN	FORMAC	SCHOONSCHIP
ASHMEDAI	MACSYMA	SCRATCHPAD
CAMAL	REDUCE	SYMBAL

Jenson and Niordson (1977) have given a comparative study of some of these systems.

REDUCE is a system for carrying out algebraic operations accurately, on relatively complicated expressions. It can manipulate polynomials in a variety of forms, both expanding and factoring them, and extracting various parts of them as required. REDUCE can also do differentiation and integration, the use of arrays and other topics of interest to physicists, mathematicians and engineers.

REDUCE is also designed to be an interactive system, so that an algebraic expression can be input and its value inspected before moving on to the next calculation. However, REDUCE can also be used in a batch mode by inputting a sequence of calculations and obtaining results without any possibility of interaction during the calculations.

To show the interactive use of REDUCE, we shall give some examples which illustrates comprehensively the capabilities of the system. If one wishes to work with the expression $(x+y)^3$, then after the appropriate logging-in procedure and when REDUCE is called, one proceeds by typing a FORTRAN-like expression, terminated by a semi-colon as follows:

```
1: (x+y)**3;
```

The semi-colon indicates the end of the expression. By pressing the Return key, the system would then input the expression, evaluate it, and return the result in a form like:

$$X^3+3*X^2*Y+3*X*Y^2+Y^3$$

```
2:
```

where (1:) is automatically assigned to the first command and after the result the next input line is labelled (2:). Input may be in lower case or upper case, but lower case is converted to upper case by the sytem, such that output is in upper case.

The results of a given calculation are saved in the variable WS (for workspace), so this can be used in the next calculation for further processing. For example, one could enter on line (2:) the expression

```
df(ws,y);
```

which calculates the derivative of the previous evaluation with respect to y, and REDUCE responds with

```
3*(X^2+2*X*Y+Y^2)
```

Alternatively,

```
int(ws,x);
```

would calculate the integral of the same expression with respect to x and REDUCE responds with

```
(X*(X^3+4*X^2*Y+6*X*Y^2+4*Y^3))/4
```

In many cases, it is necessary to use the result of one calculation in succeeding calculations. One way to do this is via an assignment for a variable, such as,

```
v:=(x+y)**3;
```

If we now use v in later calculations, the value of the right hand side of the above will be used.

An important class of commands in REDUCE is that which defines substitutions for variables and expressions to be made during the evaluation of expressions. Such substitutions use forms of the command LET.

LET rules stay in effect until replaced or CLEARed. For example, after assigning the expression $(x+y)^3$ to v, we can give numerical values to x and y and hence find the numerical value of v by using the command LET as follows:

```
let x=5, y=3;
v;
```

REDUCE responds with

```
512
```

But if we want to have the value assigned to another variable u (say), we proceed as follows:

```
let x=5, y=3;
u:=v;
```

REDUCE then responds with

```
u:=512
```

A very powerful feature of the REDUCE system is the ease with which matrix calculations can be performed and its handling of symbolic matrices. For example,

```
matrix m(3,3);
```

declares m to be a (3×3) matrix, and

```
m:=mat((a,b,c),(d,e,f),(p,q,r));
```

gives it specific element values. Expressions which include m and make algebraic sense may now be evaluated, such as 1/m to give the inverse, det(m) to give the determinant of m and 2*m-c*m**2 to give another matrix.

A REDUCE program consists of a set of functional commands which are evaluated sequentially by the computer. For example, consider the problem of finding the inverse matrix Q (say) of a (4×4) matrix R given by

$$R = \begin{bmatrix} 1 & b & c & 0 \\ a & 1 & 0 & c \\ d & 0 & 1 & b \\ 0 & d & a & 1 \end{bmatrix}$$

and then evaluate the matrix S (say) which results from multiplying the matrix Q by the matrix P, where P is given by:

$$P = \begin{bmatrix} a & 0 \\ b & a \\ a & b \\ 0 & b \end{bmatrix} .$$

The input to the system REDUCE can be written as follows:

The Input	Comments
`matrix r(4,4),q(4,4),p(4,2),s(4,2);`	Specify the dimensions of the matrices R,Q,P and S.
`r:=mat((1,b,c,0,),(a,1,0,c),` `(d,0,1,b),(0,d,a,1));`	Assign to the matrix R its elements.
`p:=mat((a,0),(b,a),(a,b),(0,b));`	Assign to the matrix P its elements.
`q:=1/r;`	Find the inverse of matrix R and assign it to the matrix Q.
`s:=q*p;`	Evaluate the matrix product of P and Q and assign the result to S.

In many applications, it is desirable to load previously prepared REDUCE files into the system, or to write output on other files. REDUCE offers some commands for this purpose, two of these commands are IN and OUT. The command <u>IN</u> takes a list of file names as argument and directs the system to input each file (which should contain REDUCE statements and commands) into the system. For example,

```
in fred 1,"fred 2";
```

will first load file fred 1, then fred 2. Files to be read using IN should end with ;END;.

The command <u>OUT</u> takes a single file name as argument, and directs the output to that file from then on, until another OUT changes the output file, or <u>SHUT</u> closes it. For example,

```
out probl;
```

will direct output to the file probl.

There are several reasons why symbolic computations are useful in the context of modelling and simulation of field problems. Some of the reasons cited by Brown and Hearn (1978) are listed below.

1. Sometimes it is prohibitively expensive, or even impossible, to solve an essentially numerical problem by purely numerical means because it involves too many variables, requires greater accuracy, or is presented in an ill-conditioned or intractable form. However, a symbolic transformation may reduce the dimensionality, evade a large source of round-off error, finesse the ill conditioning, and otherwise change the problem into one that can be solved by standard numerical methods.
2. The algebraic result obtained via symbolic computation can be subsequently evaluated over a wide range of parameter values.
3. Symbolic computation provides an opportunity for realizing the vital computational symbiosis between numerical experiments and symbolic theories.
4. Symbolic computation can be used to generate a needed numerical subroutine.
5. Finally, in the realm of partial differential equations, Cloutman and Fullerton (1977) have used symbolic multidimensional Taylor series expansions, computed by the ALTRAN system, to analyse the discretization and round-off errors of various numerical methods and also, more importantly, to eliminate inaccurate or unstable methods prior to coding and testing, and to develop methods in which the lowest order errors cancel each other out.

CHAPTER 3

METHODS FOR SOLVING LINEAR SYSTEMS OF EQUATIONS

3.1 INTRODUCTION

As mentioned in Section (2.1) and described in Sections (2.8.1) and (2.8.2), the application of the finite element and finite difference methods for solving p.d.e.'s yields a system of linear, simultaneous equations which can be represented in matrix notation as,

$$A\underline{u} = \underline{b} \ , \tag{3.1.1}$$

where A is a coefficient matrix of order $(m\times m)$, $(m\equiv N-1)$, the order of the matrix A equals the number of interior mesh points, $\underline{b}$ is a column vector containing known sources and boundary terms, and $\underline{u}$ is an unknown column vector.

This chapter deals with well-known methods of solving the system (3.1.1). First we will introduce the model problem in two and three dimensions.

3.2 THE MODEL PROBLEM

(I) Two Dimensional Case

Let us consider the Dirichlet problem for the Laplace equation (1.2.33a) which requires to determine the solution $u(x,y)$ satisfying (3.2.1) inside a closed region R and is determined on the boundary ∂R by the boundary condition (3.2.2),

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} = 0 \quad , \quad (x,y) \in R \ , \tag{3.2.1}$$

$$U(x,y) = g(x,y) \ , \quad (x,y) \in \partial R \ . \tag{3.2.2}$$

The region R under consideration is covered, as mentioned in Section (2.8.2) by a rectilinear net with mesh spacing h in the X and Y direction and mesh points (x_i,y_j), where $x_i=ih$, $y_j=jh$, $(i,j=0,1,\ldots,m+1)$.

This problem is a special case of the general 2 dimensional self adjoint elliptic equation (2.8.16), (2.8.17) given in Section (2.8.2). $(A(x,y)\equiv B(x,y)\equiv 1,\ F(x,y)\equiv G(x,y)\equiv 0)$.

Substituting the finite difference approximations for the derivatives in (3.2.1) the following five-point formula is obtained,

$$-u_{i+1,j}-u_{i-1,j}+4u_{i,j}-u_{i,j+1}-u_{i,j-1} = 0 \ , \tag{3.2.3}$$

$$u_{i,j} = g_{i,j} \equiv g(ih,jh), \quad \begin{array}{l} i=0,m+1 \text{ for } j=1,2,\ldots,m \\ j=0,m+1 \text{ for } i=1,2,\ldots,m \end{array} \tag{3.2.4}$$

which is equivalent to (2.8.25) by taking $\alpha_0=-1$, $\alpha_1=\alpha_2=\alpha_3=\alpha_4=-1/4$ and $G_{i,j}=F_{i,j}=0$.

The solution $u_{i,j}$ at the point (ih,jh) can be obtained by solving the linear system (3.2.3), which can be represented by the computational *molecule* shown in Figure (3.2.1).

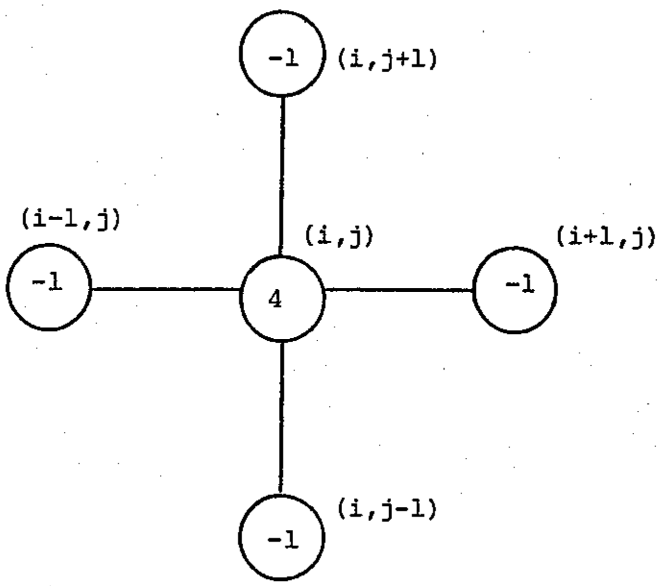


FIGURE (3.2.1): Five-point computational molecule for the Laplace operator

which when applied at each mesh point yields the system

$$A\underline{u} = \underline{b} \ , \tag{3.2.5}$$

If we order the m^2 internal mesh points column-wise (Figure (3.2.2))

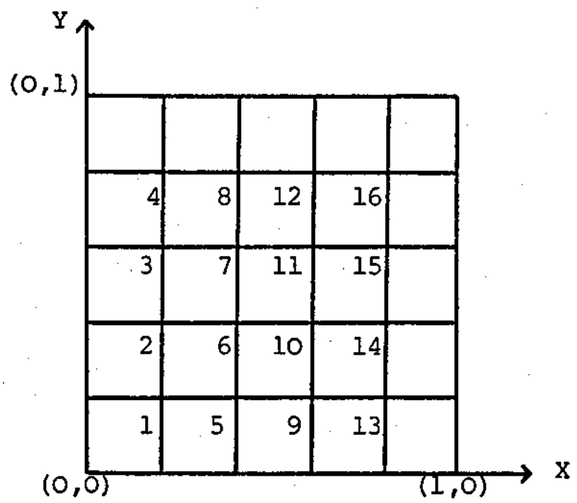


FIGURE (3.2.2)

the coefficient matrix A of the obtained system (3.2.5) is a real symmetric, square, sparse matrix and has the block tridiagonal form,

$$A = \begin{bmatrix} D_1 & -I & & & & \\ -I & D_2 & -I & & \text{O} & \\ & -I & D_3 & -I & & \\ & & \ddots & \ddots & \ddots & \\ & \text{O} & & \ddots & \ddots & -I \\ & & & & -I & D_m \end{bmatrix}, \tag{3.2.6a}$$

where I is the unit matrix of order m and the D_i $(1 \leq i \leq m)$ also of order m, are given by,

$$D_i = \begin{bmatrix} 4 & -1 & & & & \\ -1 & 4 & -1 & & \text{O} & \\ & -1 & 4 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & \text{O} & & \ddots & \ddots & -1 \\ & & & & -1 & 4 \end{bmatrix}, \quad 1 \leq i \leq m \tag{3.2.6b}$$

In addition to the column-wise ordering of the interior mesh points, there exists many different methods of ordering. In general, for a given set of mesh points (x_0+p_ih, y_0+q_ih), $i=1,2,\ldots,m$, some of the methods can be given as:

(i) *Natural ordering*; i.e. row or column-wise: a point (x_0+ph, y_0+qh) occurs before $(x_0+p'h, y_0+q'h)$, if $q<q'$ or if $q=q'$ and $p<p'$.

(ii) *Diagonal ordering*; a point (x_0+ph, y_0+qh) occurs before $(x_0+p'h, y_0+q'h)$ if $p+q<p'+q'$.

(iii) *Red-black ordering*; all points (x_0+ph, y_0+qh) with p+q even (red points) occur before those with p+q odd (black points).

For a problem arising from the solution of a five-point difference equation on a square mesh, the above kinds of ordering all lead to consistently ordered matrices (see Definition (2.7.2)).

Now, we demonstrate the procedure of deriving a formula which is more accurate than the five-point formula, i.e. the *nine-point* formula where the order of the local truncation error is increased.

Consider the two-dimensional Laplacian operator ∇^2 $(\nabla^2 \equiv \frac{\partial}{\partial x^2} + \frac{\partial}{\partial y^2})$, in Section (2.8.2) we defined D_x and D_y so ∇^2 can be written as $\nabla^2 = D_x^2 + D_y^2$.

If we assume that $h_x = h_y = h$, then by Taylor's expansion,

$$\left.\begin{aligned} U(x+h,y) &= e^{hD_x} U(x,y) , \\ U(x,y+h) &= e^{hD_y} U(x,y) , \end{aligned}\right\} , \tag{3.2.7}$$

hence we may write on the basis of the above result,

$$\left.\begin{aligned} u_{i\pm1,j} &= e^{\pm hD_x} u_{i,j} \\ u_{i,j\pm1} &= e^{\pm hD_y} u_{i,j} \\ u_{i\pm1,j\pm1} &= e^{\pm h(D_x+D_y)} u_{i,j} \end{aligned}\right\} , \tag{3.2.8a}$$

$$\left.\begin{aligned} u_{i\pm2,j} &= e^{\pm2hD_x} u_{i,j} \\ u_{i,j\pm2} &= e^{\pm2hD_y} u_{i,j} \end{aligned}\right\} , \tag{3.2.8b}$$

where $u_{i,j}$ represents values of a numerical approximation to $U_{i,j}$.

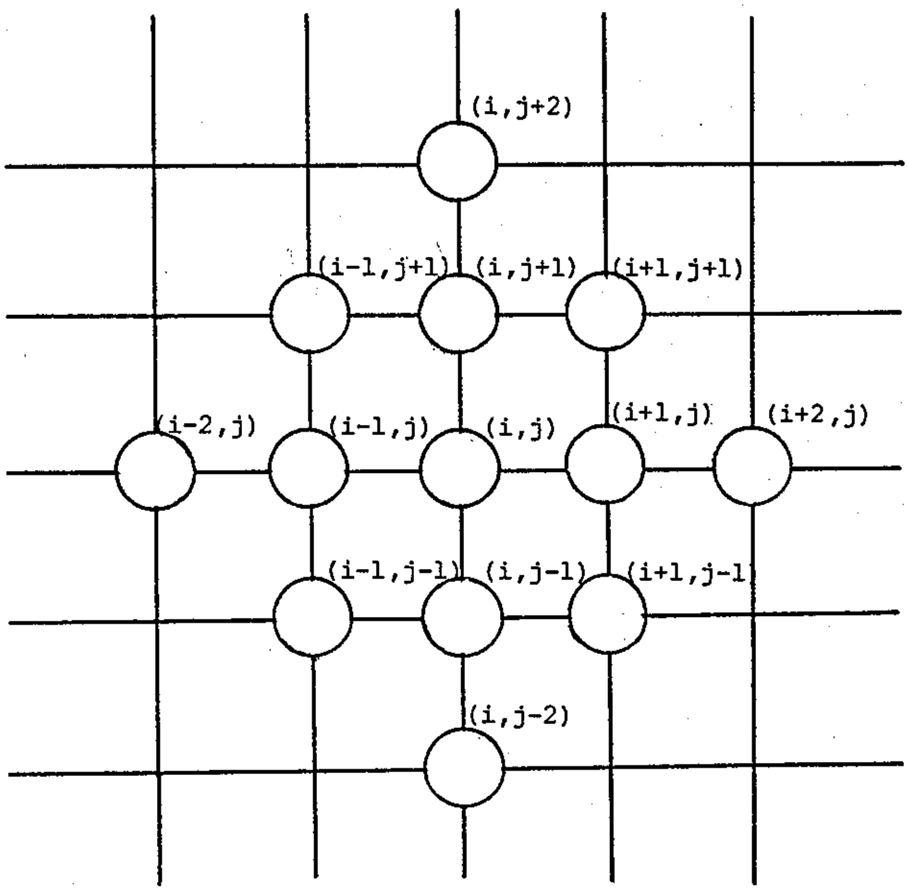


FIGURE (3.2.3)

Then we define S_1, S_2 and S_3 as follows:

$$\left.\begin{aligned} S_1 &= u_{i+1,j}+u_{i,j+1}+u_{i-1,j}+u_{i,j-1} \\ S_2 &= u_{i+1,j+1}+u_{i-1,j+1}+u_{i-1,j-1}+u_{i+1,j-1} \\ S_3 &= u_{i+2,j}+u_{i,j+2}+u_{i-2,j}+u_{i,j-2} \end{aligned}\right\} \qquad (3.2.9)$$

Hence, by using (3.2.8),

$$\begin{aligned} S_1 &= (e^{hD_x}+e^{hD_y}+e^{-hD_x}+e^{-hD_y})u_{i,j} \\ &= [2\cosh(hD_x)+2\cosh(hD_y)]u_{i,j} \\ &= [4+h^2(D_x^2+D_y^2) + \frac{h^4}{12}(D_x^4+D_y^4) + \frac{h^6}{360}(D_x^6+D_y^6)+ \ldots]u_{i,j} \\ &= [4+h^2\nabla^2 + \frac{h^4}{12}(\nabla^4-2D_x^2D_y^2) + \frac{h^6}{360}(\nabla^6-3\nabla^2D_x^4D_y^4)+\ldots]u_{i,j} \qquad (3.2.10) \end{aligned}$$

$$S_2 = (e^{h(D_x+D_y)}+e^{h(-D_x+D_y)}+e^{h(-D_x-D_y)}+e^{h(D_x-D_y)})u_{i,j}$$
$$= 4[\cosh(hD_x)\cosh(hD_y)]u_{i,j}$$
$$= [4+2h^2(D_x^2+D_y^2)+\frac{h^4}{6}(D_x^4+6D_x^2D_y^2+D_y^4) + \frac{h^6}{180}(D_x^6+15D_x^4D_y^2+15D_x^2D_y^4+D_y^6)+\ldots]u_{i,j}$$
$$= [4+2h^2\nabla^2+\frac{h^4}{6}(\nabla^4+4D_x^2D_y^2)+\frac{h^6}{180}(\nabla^6+12\nabla^2D_x^4D_y^4)+\ldots]u_{i,j}, \tag{3.2.11}$$

and

$$S_3 = (e^{2hD_x}+e^{2hD_y}+e^{-2hD_x}+e^{-2hD_y})u_{i,j}$$
$$= [4+4h^2\nabla^2 + \frac{4}{3}h^4(\nabla^4-2D_x^2D_y^2)+\frac{8h^6}{45}(\nabla^6-3\nabla^2D_x^4D_y^4)+\ldots]u_{i,j} \tag{3.2.12}$$

By eliminating the term $D_x^2D_y^2$ between S_1 and S_2, we obtain the following nine-point formula:

$$\nabla^2u_{i,j} = \frac{1}{6h^2}(4S_1+S_2-20u_{i,j}) - \frac{1}{12}h^2\nabla^4u_{i,j} - \frac{1}{180}h^4(\tfrac{1}{2}\nabla^6u_{i,j}+\nabla^2u_{i,j}D_x^4D_y^4) + O(h^6) , \tag{3.2.13}$$

For the Laplace equation $\nabla^2u=0$. Hence, $\nabla^4u=\nabla^2(\nabla^2u)=0$ and $\nabla^6u=0$.

So the nine-point formula for the Laplace operator is given by,

$$20u_{i,j}-4S_1-S_2 = 0 , \tag{3.2.14}$$

with a truncation error of order h^6.

The linear system (3.2.14) can be represented by the computational molecule shown in Figure (3.2.4)

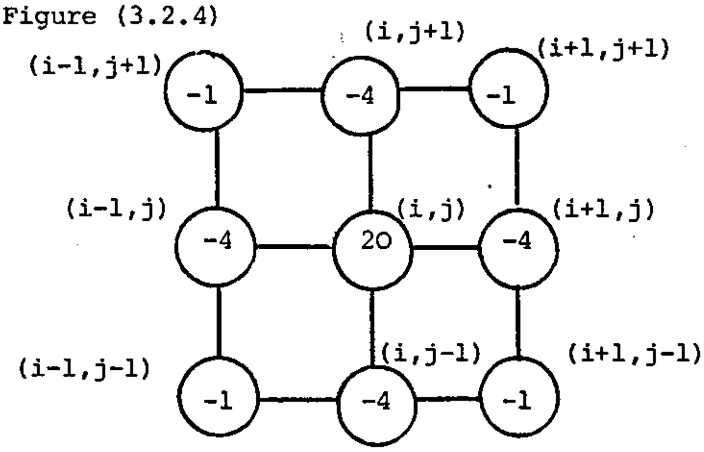


FIGURE (3.2.4): Nine-point computational molecule for the Laplace operator

Other combinations between S_1,S_2 and S_3 may yield different formulae, for example, S_3-16S_1 gives for the Laplace equation the molecule shown in Figure (3.2.5).

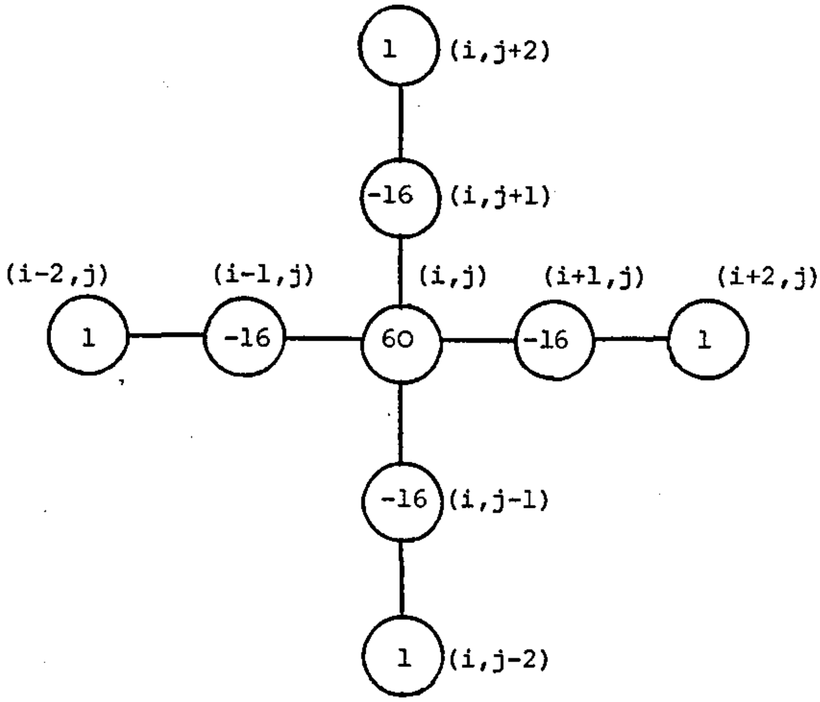


FIGURE (3.2.5)

This molecule suffers from the difficulty that its accuracy is difficult to match near the boundary ∂R. Moreover, its use of points at a distance 2h from $u_{i,j}$ is a mild disadvantage in the solution of the linear equations associated with equation (2.8.16). It is more serious disadvantage for nodes near the boundary, because the greater size of the molecule causes many more interior points of the net to become irregular. (Forsythe and Wasow, (1960), p.194). Techniques for dealing with such points (see Mitchell and Griffiths, (1980),p.131).

The *Biharmonic equation* which is a more complicated partial differential equation of elliptic type has the form (as a model problem),

$$\nabla^4 U \equiv \nabla^2 \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right) U = 0 , \qquad (3.2.15)$$

which can be written in the form,

$$\frac{\partial^4 U}{\partial x^4} + 2\frac{\partial^4 U}{\partial x^2 \partial y^2} + \frac{\partial^4 U}{\partial y^4} = 0 \ , \quad (x,y) \in R \ . \tag{3.2.16}$$

The appropriate boundary conditions usually take one of two forms,

$$\text{(i)} \quad \left.\begin{aligned} U &= g_1(x,y) \\ \frac{\partial U}{\partial n} &= g_2(x,y) \end{aligned}\right\} \text{ for } (x,y) \in \partial R \ , \tag{3.2.17}$$

or

$$\text{(ii)} \quad \left.\begin{aligned} U &= g_1(x,y) \\ \frac{\partial^2 U}{\partial n^2} &= g_2(x,y) \end{aligned}\right\} \text{ for } (x,y) \in \partial R \ , \tag{3.2.18}$$

where $\frac{\partial}{\partial n}$ is the derivative in the direction of the outward normal to the boundary ∂R.

If we take R to be the unit square $0 \le x,y \le 1$, from equation (3.2.16) by using difference replacements, assuming that $h_x = h_y = h$, we get,

$$\delta_x^4 u_{i,j} + 2\delta_x^2 \delta_y^2 u_{i,j} + \delta_y^4 u_{i,j} = 0 \ , \tag{3.2.19}$$

where $\delta_x^4 = \delta_x^2(\delta_x^2)$ and $\delta_y^4 = \delta_y^2(\delta_y^2)$.

By a simple calculation, equation (3.2.19) can be written as,

$$20u_{i,j} - 8S_1 + 2S_2 + S_1 = 0 \ , \tag{3.2.20}$$

with a truncation error of order h^2.

The system (3.2.20) can be represented by the molecule shown in Figure (3.2.6).

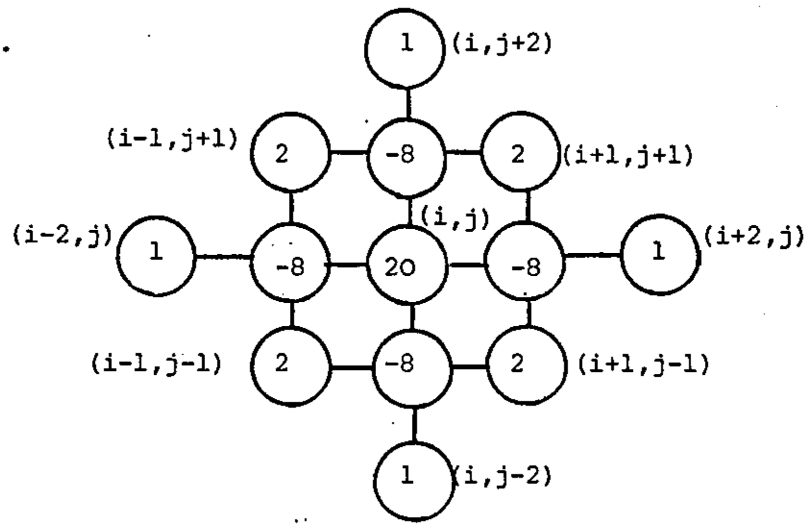


FIGURE (3.2.6): 13-point computational molecule for biharmonic operator

In this 13-point molecule the presence of the terms $u_{i\pm2,j}$ and $u_{i,j\pm2}$ means that special modifications have to be made when it is applied at nodes distance h from the boundary. Techniques for dealing with such points are discussed in detail by Fox (1950), see also (Mitchell and Griffiths, (1980), p.131).

(II) Three Dimensional Model Problem

We consider the Laplace equation in three space dimensions:

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} + \frac{\partial^2 U}{\partial z^2} = 0 , \quad (x,y,z) \in R. \tag{3.2.21}$$

subject to the Dirichlet boundary condition ,

$$U(x,y,z) = g(x,y,z), \quad (x,y,z) \in \partial R. \tag{3.2.22}$$

where R is the unit cube $0 \leq x,y,z \leq 1$. ∂R its boundary.

The volumetric region under consideration R is covered by an equally spaced three-dimensional net with mesh spacing h in the X, Y and Z directions. The mesh points are defined by (x_i, y_j, z_k) where $x_i = ih$, $y_j = jh$ and $z_k = kh$, $(0 \leq i,j,k \leq m+1)$, where $(m+1)h = 1$.

Discrete approximations to the derivatives in (3.2.21), leads to the following seven-point finite difference equation,

$$6u_{i,j,k} - u_{i+1,j,k} - u_{i-1,j,k} - u_{i,j+1,k} - u_{i,j-1,k} - u_{i,j,k+1} - u_{i,j,k-1} = 0, \tag{3.2.23}$$

$$u_{i,j,k} = g_{i,j,k}, \quad \begin{array}{l} i=0,N \text{ for } 1 \leq j,k \leq m \\ j=0,N \text{ for } 1 \leq i,k \leq m \\ k=0,N \text{ for } 1 \leq i,j \leq m. \end{array} \tag{3.2.24}$$

(3.2.23) is equivalent to (2.8.39) by taking $\alpha_i = 1$ $(1 \leq i \leq 6)$ and $G_{i,j,k} = F_{i,j,k} = 0$.

Equation (3.2.23) is generally represented by the molecule shown in Figure (3.2.7).

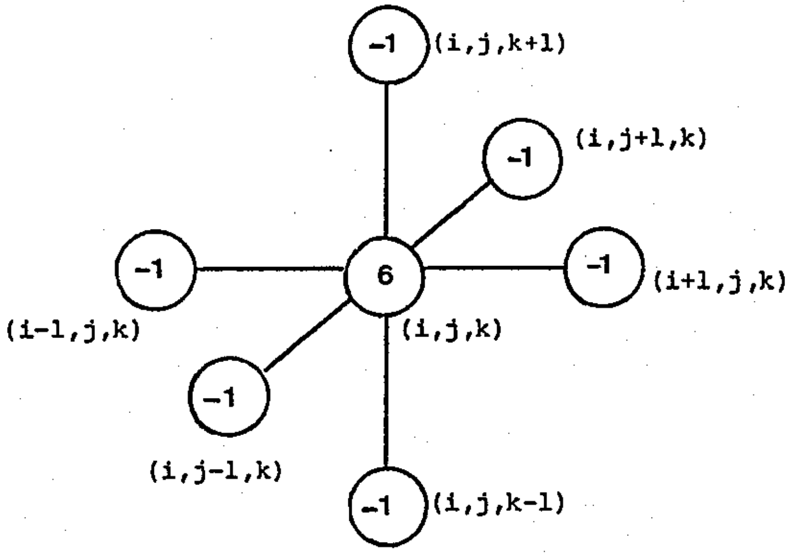


FIGURE (3.2.7)

Ordering the m^3, internal mesh points with increasing values of i, then j, then k (Figure 3.2.8).

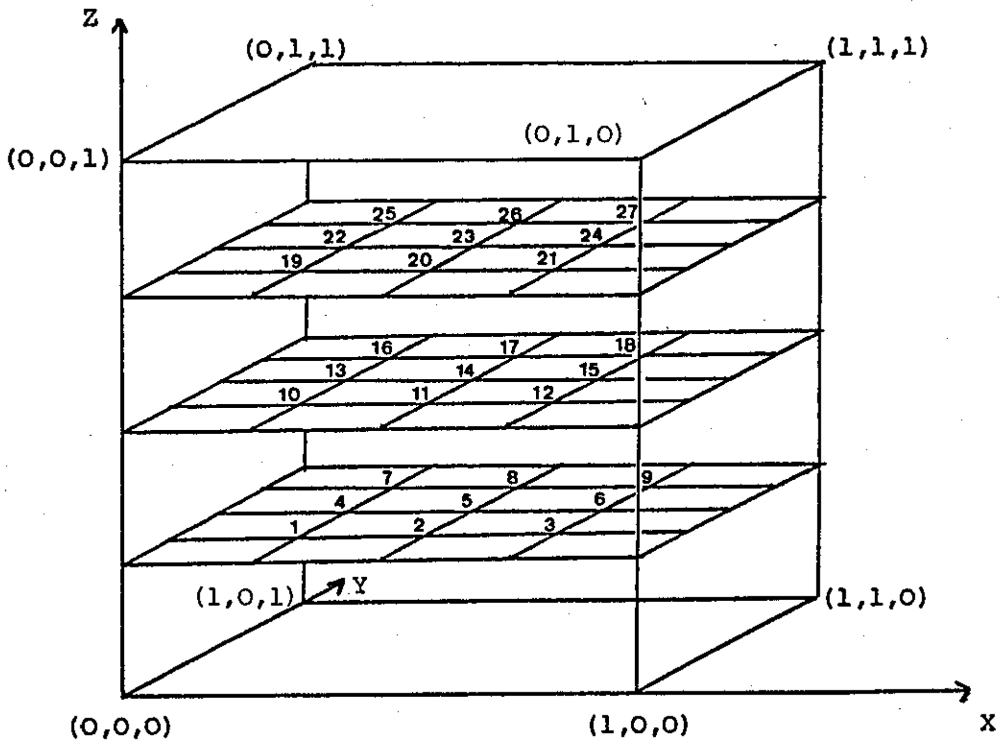


FIGURE (3.2.8)

By applying the seven-point equation at each mesh point we get the system (3.2.5). The coefficient matrix A of the obtained system is a real, symmetric, square, seven-diagonal, sparse matrix of order m^3 and of the general form,

$$A = \begin{bmatrix} A_1 & -I & & & \\ -I & A_2 & -I & & \bigcirc \\ & -I & A_3 & -I & \\ & & \ddots & \ddots & \ddots \\ & \bigcirc & & \ddots & \ddots & -I \\ & & & & -I & A_m \end{bmatrix}, \qquad (3.2.25a)$$

where I is the unit matrix of order m^2 and A_i, $1 \le i \le m$, are matrices of order m^2 given by,

$$A_i = \begin{bmatrix} D_{j+1} & -J & & & \\ -J & D_{j+2} & -J & & \bigcirc \\ & -J & \ddots & \ddots & \\ & & \ddots & \ddots & \ddots \\ & \bigcirc & & \ddots & \ddots & -J \\ & & & & -J & D_{j+m} \end{bmatrix}, \quad j=(i-1)m, \; 1 \le i \le m \qquad (3.2.25b)$$

where J is now the unit matrix of order m and D_r, $1 \le r \le m^2$, are matrices of order m given by,

$$D_r = \begin{bmatrix} 6 & -1 & & & \\ -1 & 6 & -1 & & \bigcirc \\ & -1 & 6 & -1 & \\ & & \ddots & \ddots & \ddots \\ & \bigcirc & & \ddots & \ddots & -1 \\ & & & & -1 & 6 \end{bmatrix}, \quad 1 \le r \le m^2 . \qquad (3.2.25c)$$

The solution vector $\underline{u}$ and the right hand side vector $\underline{b}$ of (3.2.5) are $(m^3 \times 1)$ column vectors.

3.3 THE SOLUTION OF A SYSTEM OF EQUATIONS

Methods used to obtain the numerical solution of the system (3.1.1) that are best suited to the capability of modern electronic computers have been developed. The criteria to compare methods are measured by the number of computer operations needed to obtain a (sufficiently accurate) solution as well as the storage requirements which are measured by the size of computer memory needed to store the matrix A and other matrices occurring in the process of solution. Usually the methods used lie in two classes, the class of *direct methods* (or elimination methods) and the class of *iterative methods* (or indirect methods) which mainly depend upon the structure of the coefficient matrix A.

That is, if A is a large sparse matrix as in problems which arise in large order p.d.e.'s, *iterative methods* are usually used, since these will not change the structure of the original matrix and therefore preserve sparsity. They are essentially based on generating a sequence of approximate solutions $\{\underline{u}^{(s)}\}$, $s=0,1,2,\ldots$ for (3.1.1) and hope that this sequence approaches the solution $A^{-1}b$ provided that the inverse exists. However, no arithmetic is associated with zero coefficients, so considerably fewer numbers have to be stored in the computer (the non-zero elements of the coefficient matrix) and hence minimise the amount of storage used. As a consequence they can be used to solve systems of equations that are too large for the use of direct methods. Programming and data handling are also much simpler than for direct methods. Another advantage, not possessed by direct methods, is their frequent extension to the solution of sets of non-linear equations (Smith, (1978)). 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.

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

For non-linear problems, the most common procedure for solving these kinds of problems is to formulate them as the solution of an iterative sequence of linear (or rather linearized) equations to which solution methods for linear problems still apply.

In the following sections we will describe some well known direct and iterative methods.

3.4 DIRECT METHODS

Direct methods for solving linear systems are compared for efficiency on the basis of the number of arithmetical operations required. One of the methods used for obtaining the solution of a system of equations is given by *Cramer's rule*, which can be stated in the following theorem.

Theorem 3.4.1

In the system (3.1.1), let $\det A \equiv |A| \neq 0$, and let $\det A_i$, $(i=1,2,\ldots,m)$, be the determinant of the matrix obtained from A by replacing the i^{th} column of A by the column vector $\underline{b}$. Then, the solution of the system (3.1.1) is,

$$u_i = \frac{\det A_i}{\det A} \quad , \; i=1,2,\ldots,m \; . \qquad (3.4.1)$$

The use of this method in actual numerical cases is generally grossly inefficient, because of the excessive labour involved in the evaluation of (m+1) determinants of order m, unless m is small. For example to solve (3.1.1) with order 10 (which is a trivially small problem by modern standards) would require some 68 million multiplications. The number of operations involved in this method is of order (m!) if the system is of order m, (Froberg, (1979), p.81).

A more efficient method of evaluating the determinants can reduce this to about 3000 multiplications, but even this is inefficient compared to Gaussian elimination, which requires about 380 multiplications (Gerald, (1980), p.99), this, as well as other methods are described as follows.

Systematic Elimination Methods

These methods are based ultimately on the process of the elimination of variables. The most widely used methods are:-

(i) *Gaussian Elimination* which involves a finite number of transformations of a given rectangular system of equations into an upper triangular system which is a system which is much more easily solved. Precisely the number of the transformation is one less than the size of the given system. Thus, for the system (3.1.1) we have after (m-1) steps the final system, (Ralston, (1965)),

$$\begin{bmatrix} a_{1,1}^{(0)} & a_{1,2}^{(0)} & \cdots & \cdots & \cdots & a_{1,m}^{(0)} \\ & a_{2,2}^{(1)} & a_{2,3}^{(1)} & \cdots & \cdots & a_{2,m}^{(1)} \\ & & a_{3,3}^{(2)} & a_{3,4}^{(2)} & \cdots & a_{3,m}^{(2)} \\ & & & \ddots & \ddots & \vdots \\ & 0 & & & a_{m-1,m-1}^{(m-2)} & a_{m-1,m}^{(m-2)} \\ & & & & & a_{m,m}^{(m-1)} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{m-1} \\ u_m \end{bmatrix} = \begin{bmatrix} b_1^{(0)} \\ b_2^{(1)} \\ b_3^{(2)} \\ \vdots \\ b_{m-1}^{(m-2)} \\ b_m^{(m-1)} \end{bmatrix} \quad (3.4.2)$$

where,

$$\left.\begin{aligned} a_{i,j}^{(k)} &= a_{i,j}^{(k-1)} - \frac{a_{i,k}^{(k-1)}}{a_{k,k}^{(k-1)}} a_{k,j}^{(k-1)} , \quad && k=1,2,\ldots,m-1 \\ & && i=k+1,\ldots,m \\ & && j=k+1,\ldots,m \\ b_i^{(k)} &= b_i^{(k-1)} - \frac{a_{i,k}^{(k-1)}}{a_{k,k}^{(k-1)}} b_k^{(k-1)} , && a_{i,j}^{(0)}=a_{i,j} \\ & && b_1^{(0)}=b_1 \end{aligned}\right\} \quad (3.4.3)$$

The solution for (3.4.2) is given by,

$$\left.\begin{aligned} u_m &= \frac{b_m^{(m-1)}}{a_{m,m}^{(m-1)}} \\ u_i &= \frac{1}{a_{i,i}^{(i-1)}} [b_i^{(i-1)} - \sum_{j=i+1}^{m} a_{i,j}^{(i-1)} u_j] , \\ & \qquad i=m-1,m-2,\ldots,1 \end{aligned}\right\} \quad (3.4.4)$$

Strategies for Avoiding a Small or Zero Pivot

If any of the diagonal elements of the matrix A in the system (3.1.1) becomes zero during the elimination process, i.e. $a_{k,k}^{(k-1)}=0$ for any k, then the final upper triangular form will be unattainable, and hence the process will fail. To overcome this we locate a j $(k+1\leq j\leq m)$ such that $a_{j,k}^{(k-1)}\neq 0$ and we interchange the j^{th} and k^{th} rows of $A^{(k)}$ and corresponding column in $b^{(k)}$ (i.e. we reorder the equations). The interchanges are referred to as *pivoting* which is normally employed to preserve arithmetic accuracy.

Definition 3.4.1

Any of the diagonal elements in (3.4.2), i.e. $a_{k,k}^{(k-1)}$, $1\leq k\leq m$, is termed the k^{th}*-pivot*. If it is zero, then it is called a *zero-pivot*.

There are two basic well known pivoting schemes. They are mainly concerned with avoiding a zero pivot which may arise at any stage of the elimination process.

(1) Partial Pivoting

This strategy involves choosing an element of largest magnitude in the column of each reduced matrix as the pivot, elements of rows which have previously been pivotal being excluded from consideration. This way of pivoting can be easily illustrated in Figure (3.4.1) where x denotes a non-zero element.

$$A^{(2)} = \begin{bmatrix} x & x & x & x & x \\ 0 & x & x & x & x \\ 0 & 0 & x & x & x \\ 0 & 0 & \otimes & x & x \\ 0 & 0 & x & x & x \end{bmatrix}$$

Any element in the box can be taken as the pivot. If $\otimes$ is the largest magnitude then the 3rd and 4th row of $A^{(2)}$ have to be interchanged.

FIGURE (3.4.1)

(2) Complete (Full) Pivoting

The pivot at each stage of the reduction is chosen as the element of largest magnitude in the submatrix of rows which have not been pivotal up to now, regardless of the position of the element in this matrix. This may require both row and column interchanges. Figure (3.4.2) illustrates the strategy.

$$A^{(2)} = \begin{bmatrix} x & x & x & x & x \\ 0 & x & x & x & x \\ 0 & 0 & x & x & x \\ 0 & 0 & x & x & x \\ 0 & 0 & x & x & x \end{bmatrix}$$

Any element in the box can be taken as a pivot. If ⊗ is the element of largest magnitude then we interchange the 3rd and 4th rows of $A^{(2)}$, followed by the 3rd and 5th columns.

It is also necessary to interchange the elements of the solution vector to compensate for the column interchanges of $A^{(2)}$. Further, in both cases the elements of the right-hand side vector, $b^{(2)}$, are interchanged similarly to the rows of $A^{(2)}$.

In practice complete pivoting is time consuming in execution and is not frequently used. In addition, for large m and matrices with special pattern, (e.g. band matrices), partial pivoting is less likely to completely destory this pattern.

(ii) Gauss-Jordan Scheme

An alternative to Gauss elimination method (G.E.) is the Gauss Jordan scheme (G.J.) which leads to a diagonal matrix rather than triangular at the end of the process. In this method, the elements above the diagonal are made zero at the same time that zeros are created below the diagonal, and hence the solution can be obtained by dividing the components of the right-hand side vector, $\underline{b}$, by the corresponding diagonal elements, i.e., there is no need for the back substitution stage as in Gauss elimination.

However, G.E. can be shown to be superior to G.J. since it involves a smaller amount of work. For large m the G.J. method requires almost 50% more operations than the G.E. method (Ralston (1965)). It is showed by Fox (1964) that the amount of work involved in the G.E. and G.J. methods are as follows:

G.E.	G.J.	Operations
m	m	division
$\frac{1}{3}m^3+m^2-\frac{1}{3}m$	$\frac{1}{2}m^3+m^2-\frac{1}{2}m$	multiplication
$\frac{1}{3}m^3+\frac{1}{2}m^2-\frac{5}{6}m$	$\frac{1}{2}m^3-\frac{1}{2}m$	addition

(iii) Triangular, or LU, Decomposition

If we define the multipliers of the k^{th} stage of the transformation for the G.E. method as

$$t_{i,k} = \frac{a_{i,k}}{a_{k,k}}, \quad \begin{array}{l} k=1,2,\ldots,m-1 \\ i=k+1,\ldots,m. \end{array} \tag{3.4.5}$$

where $t_{i,1}=a_{i,1}/a_{1,1}$, $i=2,3,\ldots,m$, is used to generate the zeros required in the 1^{st} column. Then we define the $(m\times m)$ unit lower triangular matrices $M_1,M_2,\ldots,M_k$, $k=1,2,\ldots,m-1$, as follows (see Goult et al (1974), Ralston (1965))

$$M_1 = \begin{bmatrix} 1 & & & & \\ -t_{2,1} & 1 & & O & \\ -t_{3,1} & & 1 & & \\ \vdots & & & \ddots & \\ -t_{m,1} & & O & & 1 \end{bmatrix}, \quad M_2 = \begin{bmatrix} 1 & & & & \\ 0 & 1 & & O & \\ 0 & -t_{3,2} & \ddots & & \\ 0 & -t_{4,2} & O & \ddots & \\ \vdots & \vdots & & & \\ 0 & -t_{m,2} & & & 1 \end{bmatrix} \cdots$$

$$M_k = \begin{bmatrix} 1 & & & & \\ & \ddots & & O & \\ & & 1 & & \\ & & -t_{k+1,k} & \ddots & \\ & O & -t_{k+2,k} & & \ddots \\ & & \vdots & & \\ & & -t_{m,k} & O & 1 \end{bmatrix}$$

Thus, the matrix form (3.4.2) is equivalent to,

$$M A \underline{u} = M \underline{b} \ , \tag{3.4.6}$$

where $$M = M_{m-1} M_{m-2} \cdots M_1 \ , \tag{3.4.7}$$

$$M_{m-1} M_{m-2} \cdots M_1 A = \begin{bmatrix} a_{1,1}^{(0)} & a_{1,2}^{(0)} & - - - & - - - & a_{1,m}^{(0)} \\ & a_{2,2}^{(1)} & a_{2,3}^{(1)} & - - - & a_{2,m}^{(1)} \\ & & a_{3,3}^{(2)} & - - - & a_{3,m}^{(2)} \\ & \bigcirc & & \ddots & \vdots \\ & & & & a_{m,m}^{(m-1)} \end{bmatrix} \equiv \begin{bmatrix} v_{1,1} & v_{1,2} & - - - & v_{1,m} \\ & v_{2,2} & v_{2,3} - - - & v_{2,m} \\ & & \ddots & \vdots \\ & \bigcirc & & v_{m,m} \end{bmatrix} \equiv U . \tag{3.4.8}$$

Hence, we have, $$A = M_1^{-1} M_2^{-1} \cdots M_{m-1}^{-1} U \ . \tag{3.4.9}$$

But M_k^{-1}, $1 \le k \le m-1$, is M_k itself with the signs of its off-diagonal elements reversed and the product $M_1^{-1} M_2^{-1} \cdots M_{m-1}^{-1}$ is given by,

$$M_1^{-1} M_2^{-1} \cdots M_{m-1}^{-1} = \begin{bmatrix} 1 & & & & \\ t_{2,1} & 1 & & \bigcirc & \\ t_{3,1} & t_{3,2} & 1 & & \\ \vdots & \vdots & \ddots & \ddots & \\ t_{m,1} & t_{m,2} & - - - & t_{m,m-1} & 1 \end{bmatrix} \equiv L . \tag{3.4.10}$$

Thus, $$A = LU \ , \tag{3.4.11}$$

where L is a unit lower triangular matrix and U is an upper triangular matrix. The form (3.4.11) is termed *triangular or LU* decomposition.

Now, we can write (3.1.1) as,

$$LU\underline{u} = \underline{b} \ . \tag{3.4.12}$$

The solution of (3.1.1) by this algorithm follows from (3.4.12) by introducing an auxiliary vector, $\underline{y}$ (say), such that the system (3.4.12) will be split into two triangular systems,

$$L\underline{y} = \underline{b}\ , \tag{3.4.13}$$

and

$$U\underline{u} = \underline{y}\ . \tag{3.4.14}$$

The two vectors $\underline{y}$ and $\underline{u}$ can be obtained easily from equations (3.4.13 and 14) by forward and backward substitution processes respectively. The equations for the forward and backward substitutions can be summarised as follows,

$$\left.\begin{aligned} y_1 &= b_1\ , \\ y_i &= b_i - \sum_{k=1}^{i-1} t_{ik} y_k\ ,\ i=2,3,\ldots,m, \end{aligned}\right\} \tag{3.4.15}$$

and,

$$\left.\begin{aligned} u_m &= \frac{y_m}{v_{mm}} \\ u_j &= y_j - \sum_{k=j+1}^{m-1} v_{jk} u_k,\ j=m-1,m-2,\ldots,1 \end{aligned}\right\} \tag{3.4.16}$$

provided $v_{ii} \neq 0$ $(i=1,2,\ldots,m)$, i.e. U is non-singular.

Pivoting with the LU method is somewhat more complicated than with the G.E. method, because we do not usually handle the right-hand side vector simultaneously with our reduction of the matrix A. (Gerald, (1980), p.94). This means we must keep a record of any row interchanges made during the formulation of L and U so that the elements of $\underline{b}$ can be similarly interchanged.

It can be shown that the amount of work in the LU method is the same as for G.E. method. But the disadvantage of Gauss's process in comparison with LU process is that a rounding error occurs when an element of a reduced matrix is computed and stored. This can be largely avoided in the LU process by the use of double-precision arithmetic in the calculation of the elements of L and U, and then the result may be rounded to single precision and recorded on the completion of each calculation. The removal of the need for computing and recording several intermediate matrices has 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 Gauss process were carried out with double precision. This latter is an unattractive proposition, since it would require twice as much computer storage as the corresponding single precision solution. (Goult et al (1974), p.59).

However, the LU-decomposition may be applied when the following theorem is valid.

Theorem 3.4.2

A non-singular matrix A may be decomposed into the product LU if and only if every leading principal submatrix of A is non-singular.

Corollary 3.4.1

If L is a unit lower triangular matrix then the decomposition is unique.

The proofs of Theorem (3.4.2) and Corollary (3.4.1) are given in (Broyden (1975), p.56).

Corollary 3.4.2

If U is a unit upper triangular matrix then the decomposition is unique.

The LU-decomposition is often called *Doolittle's method* if Corollary (3.4.1) is valid, whilst if Corollary (3.4.2) is valid it is called *Crout's method*.

The Decomposition of a Symmetric Matrix

If the matrix A in (3.1.1) is symmetric and satisfies the hypothesis of Theorem (3.4.2), then we can express A in the form,

$$A = LDL^T , \tag{3.4.17}$$

where L is a unit lower triangular matrix and D is a diagonal matrix. This method will fail if any of the pivots is zero, since the use of row interchanges to avoid this destroys the symmetry of the system. But if A is a real symmetric and positive definite matrix then a real lower triangular matrix can be found such that,

$$A = LL^T , \tag{3.4.18}$$

so, if $L=(\ell_{i,j})$, where $\ell_{i,j}=0$ for $i<j$, then,

$$\left.\begin{aligned} \ell_{i,i} &= [a_{j,j} - \sum_{k=1}^{j-1} \ell_{j,k}^2]^{\frac{1}{2}} , \text{ if } i=j , \\ \ell_{i,j} &= \frac{1}{\ell_{j,j}}[a_{j,i} - \sum_{k=1}^{j-1} \ell_{i,k}\ell_{j,k}] , \ j<i\leq m \end{aligned}\right\} j=1,2,\ldots,m \tag{3.4.19}$$

This is known as the Choleski decomposition (or the square root method), and it requires roughly half the storage space and the computational labour of LU decomposition. The main disadvantage being that it needs the calculation of the square roots, but this may be avoided by the decomposition (3.4.17). Thus, Choleski's method is an attractive proposition for problems involving symmetric positive definite matrices such as those occurring in discretised elliptic problems.

3.5 ITERATIVE METHODS

In Section (3.3) we gave a basic definition of the iterative method, in which a sequence of approximate solution vectors $\{\underline{u}^{(k)}\}$ are required to solve the non-singular system,

$$A\underline{u} = \underline{b} \,, \tag{3.5.1}$$

such that $\underline{u}^{(k)} \to A^{-1}\underline{b}$ as $k \to \infty$.

If we define a sequence of functions (iteration functions), $F_0(A,b)$, $F_1(A,\underline{b},\underline{u}^{(0)}),\ldots,F_k(A,\underline{b},\underline{u}^{(0)},\underline{u}^{(1)},\ldots,\underline{u}^{(k-1)})$, where,

$$\begin{aligned} \underline{u}^{(0)} &= F_0(A,\underline{b}) \\ \underline{u}^{(k+1)} &= F_{k+1}(A,\underline{b},\underline{u}^{(0)},\underline{u}^{(1)},\ldots,\underline{u}^{(k)}) \end{aligned} \tag{3.5.2}$$

then, the iterative method is:

Stationary: If for some integer $\ell>0$, F_k is independent of k for all $k \geq \ell$. Otherwise, it is *non-stationary*.

Linear: If for each k, F_k is a linear function of $\underline{u}^{(0)},\underline{u}^{(1)},\ldots$ and $\underline{u}^{(k-1)}$. Otherwise, it is *non-linear*.

Consistent: If at any stage of the iterative process we obtain a solution to (3.5.1), then all subsequent iterants remain unchanged.

The most general linear stationary iteration has the form,

$$\underline{u}^{(k+1)} = G\underline{u}^{(k)} + \underline{r} \,, \tag{3.5.3}$$

where G, *the iteration matrix*, is a matrix depending upon A and $\underline{b}$, and $\underline{r}$ is a vector to be defined later. The exact solution of (3.5.1), i.e. $A^{-1}\underline{b}$, at a fixed point of (3.5.3) can be reproduced, and we have,

$$A^{-1}\underline{b} = GA^{-1}\underline{b} + \underline{r} \,, \tag{3.5.4}$$

hence

$$\underline{r} = (I-G)A^{-1}\underline{b} \,. \tag{3.5.5}$$

This is called the *consistency condition*. It can be shown (see Young, (1971), p.65) that if the consistency condition applies, then for some k,

$$\underline{u}^{(k+1)} = G\underline{u}^{(k)} + \underline{r} = G\underline{u} + \underline{r} = \underline{u} \; .$$

That is, once the solution is obtained, the iterative process makes no further modification of successive iterates.

We consider two classes of stationary iterative methods. These are, (I) *The Simultaneous class* and (II) *The Successive class.* The main difference between these two classes is the ordering. In (I), the order in which the solution vectors are calculated does not affect the values of successive iterates at a particular vector, in other words, all elements of the approximate solution are modified at the same time, i.e. the $(k+1)^{th}$ iterate is a function of the k^{th} and earlier iterates only. In (II), the approximate solution values are modified one after the other, using the latest available values of the iterate, so that some elements of the $(k+1)^{th}$ iterate are functions of the $(k+1)^{th}$ and earlier iterates. In this case, the ordering of the mesh points is significant. Successive methods are themselves of two types, *point iterative methods* in which each component of the iterate is modified by an explicit calculation, and *block iterative methods* in which blocks of equations are modified successively, the blocks themselves being solved simultaneously.

Basic Iterative Methods

Let us consider, without loss of generality that the $(m \times m)$ non-singular coefficient matrix A of the system (3.5.1) can be expressed as,

$$A = Q - S \; , \tag{3.5.6}$$

where Q and S are also $(m \times m)$ matrices, and Q is non-singular. This expression then represents a *splitting* of the matrix A. Equation (3.5.1) then becomes,

$$Q\underline{u} = S\underline{u} + \underline{b} \; , \tag{3.5.7}$$

Different settings of the matrices Q and S will clearly give different iterative methods. The main iterative methods are given by:

(i) *The Jacobi method* or the method of Simultaneous Displacement.

It is one of the famous methods in class (I), in this method we assume, without loss of generality, that $Q=D$ and $S=E+F$, where D is the main diagonal elements of A.E and F are strictly lower and upper $(m\times m)$ triangular matrices respectively. Hence equation (3.5.7) can be written as,

$$D\underline{u} = (E+F)\underline{u} + \underline{b} \ . \qquad (3.5.8)$$

By the assumption that A is non-singular matrix, thus D^{-1} exists and we can replace the system (3.5.8) by the equivalent system,

$$\underline{u} = D^{-1}(E+F)\underline{u} + D^{-1}\underline{b} \ . \qquad (3.5.9)$$

The Jacobi iterative method is defined by,

$$\underline{u}^{(k+1)} = B\underline{u}^{(k)} + \underline{g} \ , \qquad (3.5.10)$$

where B *is the Jacobi iterative matrix* associated with the matrix A and is given by,

$$B = D^{-1}(E+F) \ , \qquad (3.5.11)$$

and $\underline{g}=D^{-1}\underline{b}$. In this method the components of the vector $\underline{u}^{(k)}$ must be saved while computing the components of $\underline{u}^{(k+1)}$.

(ii) *The Gauss-Seidel Method (The GS Method)* also known as the Successive Displacement Method. This method is based on the immediate use of the improved values $u_i^{(k+1)}$ instead of $u_i^{(k)}$. By setting $Q=D-E$ and $S=F$, the matrices, D,E and F as defined before, equation (3.5.7), become,

$$(D-E)\underline{u} = F\underline{u} + \underline{b} \ . \qquad (3.5.12)$$

Then, the GS iterative method is defined by,

$$D\underline{u}^{(k+1)} = E\underline{u}^{(k+1)} + F\underline{u}^{(k)} + \underline{b} \ . \qquad (3.5.13)$$

By multiplying both sides by D^{-1}, the following equation is obtained,

$$\underline{u}^{(k+1)} = L\underline{u}^{(k+1)} + R\underline{u}^{(k)} + \underline{g} \ , \qquad (3.5.14)$$

where $L=D^{-1}E$, $R=D^{-1}F$ and $\underline{g}=D^{-1}\underline{b}$. L and R are strictly lower and strictly upper triangular matrices. Equation (3.5.14) can be written as,

$$(I-L)\underline{u}^{(k+1)} = R\underline{u}^{(k)} + \underline{g} \tag{3.5.15}$$

Since L is strictly lower triangular matrix then det(I-L)=1, hence (I-L) is a non-singular matrix, therefore $(I-L)^{-1}$ exists and the GS iterative method will take the form,

$$\underline{u}^{(k+1)} = \mathcal{L}\underline{u}^{(k)} + \underline{t} \,, \tag{3.5.16}$$

where $\mathcal{L}$ *is the GS iterative matrix* and is given by,

$$\mathcal{L} = (I-L)^{-1}R \,, \tag{3.5.17}$$

and

$$\underline{t} = (I-L)^{-1}\underline{g} \,. \tag{3.5.18}$$

The computational advantage of this method that it does not require the simultaneous storage of the two approximations $u_i^{(k+1)}$ and $u_i^{(k)}$ in the course of the computations as does the Jacobi iterative method.

Now, the following two methods are related to the Jacobi method and GS method respectively.

(iii) The Simultaneous Overrelaxation Method (JOR Method)

This method is a modification to the Jacobi method. If we assume that $\underline{\tilde{u}}^{(k+1)}$ is the vector obtained from the Jacobi method, then from (3.5.10),

$$\underline{\tilde{u}}^{(k+1)} = B\underline{u}^{(k)} + \underline{g} \,, \tag{3.5.19}$$

and by choosing a real parameter ω, the actual vector $\underline{u}^{(k+1)}$ of this iteration method is determined from,

$$\underline{u}^{(k+1)} = \omega\underline{\tilde{u}}^{(k+1)} + (1-\omega)\underline{u}^{(k)} \,. \tag{3.5.20}$$

Elimination of $\underline{\tilde{u}}^{(k+1)}$ between equations (3.5.19) and (3.5.20) leads to

$$\underline{u}^{(k+1)} = B_\omega\underline{u}^{(k)} + \omega\underline{g} \,, \tag{3.5.21}$$

where B_ω *is the iteration matrix of the J.O.R.* method and is given by,

$$B_\omega = [\omega D^{-1}(E+F)+(1-\omega)I] \,. \tag{3.5.22}$$

The real parameter ω is called the *relaxation factor*.

If $\omega=1$, we have the Jacobi method. If $\omega>1$ (<1) then we are in a sense carrying out the operation of "overrelaxation"(underrelaxation) at each of the nodal points. Both, the Jacobi and J.O.R. methods are clearly independent of the order in which the mesh points are scanned.

(iv) The Successive Overrelaxation Method (S.O.R. Method)

Related to the GS method is the S.O.R. method. This method is the same as the J.O.R. method except that one uses the values of $u_i^{(k+1)}$ whenever possible. Hence, from equation (3.5.14), the S.O.R. method is defined as,

$$\underline{u}^{(k+1)} = \omega(L\underline{u}^{(k+1)}+R\underline{u}^{(k)}+\underline{g})+(1-\omega)\underline{u}^{(k)} \ . \tag{3.5.23}$$

Equation (3.5.23) can be written in the form,

$$(I-\omega L)\underline{u}^{(k+1)} = [\omega R+(1-\omega)I]\underline{u}^{(k)}+\omega\underline{g} \ . \tag{3.5.24}$$

But $(I-\omega L)$ is non-singular for any choice of ω, since $\det(I-\omega L)=1$. So we can solve (3.5.24) for $\underline{u}^{(k+1)}$ obtaining,

$$\underline{u}^{(k+1)} = \mathcal{L}_\omega\underline{u}^{(k)}+(I-\omega L)^{-1}\omega\underline{g} \ , \tag{3.5.25}$$

where $\mathcal{L}_\omega$ *is the S.O.R. iteration matrix* and is given by,

$$\mathcal{L}_\omega = (I-\omega L)^{-1}(\omega R+(1-\omega)I) \ . \tag{3.5.26}$$

For $\omega=1$, we have the GS method. And $\omega>1$ (<1) corresponds to the cases of overrelaxation (underrelaxation) as discussed before. The GS method and the S.O.R. method both depend upon the order in which the points are scanned.

(v) The Symmetric S.O.R. Method (The S.S.O.R. Method)

This method involves two half iterations using the S.O.R. method. The first half iteration is the ordinary S.O.R. method while the second half iteration is the S.O.R. method using the nodal points in reverse order. Hence, we can define the S.S.O.R. iterative method by,

$$\underline{u}^{(k+\frac{1}{2})} = \mathcal{L}_\omega\underline{u}^{(k)}+(I-\omega L)^{-1}\omega\underline{g} \ , \tag{3.5.27a}$$

and $$\underline{u}^{(k+1)} = K_\omega \underline{u}^{(k+\frac{1}{2})} + (I-\omega R)^{-1}\omega \underline{g} \;, \tag{3.5.27b}$$

where $\underline{u}^{(k+\frac{1}{2})}$ is an intermediate approximation to the solution, L_ω is given in (3.5.26) and K_ω is given by,

$$K_\omega = (I-\omega R)^{-1}[\omega L+(1-\omega)I] \;. \tag{3.5.28}$$

By eliminating $\underline{u}^{(k+\frac{1}{2})}$ between equations (3.5.27a) and (3.5.27b) we have,

$$\underline{u}^{(k+1)} = H_\omega \underline{u}^{(k)} + \underline{d}_\omega \;, \tag{3.5.29}$$

where H_ω is the S.S.O.R. iteration matrix and is given by,

$$H_\omega = K_\omega L_\omega = (I-\omega R)^{-1}(I-\omega L)^{-1}[\omega L+(1-\omega)I][\omega R+(1-\omega)I] \tag{3.5.30}$$

and,

$$\underline{d}_\omega = \omega(2-\omega)(I-\omega R)^{-1}(I-\omega L)^{-1}\underline{g} \;. \tag{3.5.31}$$

It is clear that the iterative methods described above can be presented in the form,

$$\underline{u}^{(k+1)} = G\underline{u}^{(k)} + \underline{r} \;, \tag{3.5.32}$$

according to the following table.

METHOD	THE ITERATION MATRIX G	THE VECTOR $\underline{r}$
Jacobi	$D^{-1}(E+F)=L+R$	$D^{-1}\underline{b}$
G.S.	$(I-L)^{-1}R$	$(I-L)^{-1}D^{-1}\underline{b}$
J.O.R.	$\omega D^{-1}(E+F)+(1-\omega)I$ $=\omega(L+R)+(1-\omega)I$	$\omega D^{-1}\underline{b}$
S.O.R.	$(I-\omega L)^{-1}[\omega R+(1-\omega)I]$	$(I-\omega L)^{-1}\omega D^{-1}\underline{b}$
S.S.O.R.	$I-\omega(2-\omega)(I-\omega R)^{-1}(I-\omega L)^{-1}D^{-1}A$	$\omega(2-\omega)(I-\omega R)^{-1}(I-\omega L)^{-1}D^{-1}\underline{b}$

<u>TABLE (3.5.1)</u>

where G, $\underline{r}$, D, E, F, L and R are as defined earlier in this section.

A Non-Linear Overrelaxation Method (N.L.O.R.)

When the finite difference approximations are directly applied to a nonlinear elliptic partial differential equation, a system of nonlinear algebraic equations is obtained. Several methods have been proposed to solve these systems (see Ames (1965), p.389) in which the solution is obtained by an outer iteration (Newton's method say) which linearizes the equations, followed by some iterative technique, for example the S.O.R. method. This process is repeated until convergence is obtained, thus constructing a cascade of outer iterations alternated with a large sequence of inner linear iterations.

Alternative to the above strategy, a direct and simple method which is particularly well adapted for solving algebraic systems associated with non-linear elliptic equations is due to Lieberstein (1959) - a method called *nonlinear overrelaxation* (N.L.O.R.).

Consider a system of m equations each having continuous first derivatives

$$f_p(x_1,x_2,\ldots,x_m) = 0 \ , \ p=1,2,\ldots,m. \tag{3.5.33}$$

The basic idea in this N.L.O.R. method is to introduce a relaxation factor ω and solve the iterative sequence of equations given by,

$$\begin{aligned}
x_1^{(k+1)} &= x_1^{(k)} - \omega \frac{f_1(x_1^{(k)},x_2^{(k)},\ldots,x_m^{(k)})}{f_{11}(x_1^{(k)},x_2^{(k)},\ldots,x_m^{(k)})} \ , \\
x_2^{(k+1)} &= x_2^{(k)} - \omega \frac{f_2(x_1^{(k+1)},x_2^{(k)},\ldots,x_m^{(k)})}{f_{22}(x_1^{(k+1)},x_2^{(k)},\ldots,x_m^{(k)})} \ . \\
&\vdots \\
x_m^{(k+1)} &= x_m^{(k)} - \omega \frac{f_m(x_1^{(k+1)},x_2^{(k+1)},\ldots,x_m^{(k)})}{f_{mm}(x_1^{(k+1)},x_2^{(k+1)},\ldots,x_m^{(k)})} \ ,
\end{aligned} \tag{3.5.34}$$

where $f_{pq} = \partial f_p/\partial x_q$.

This method has a feature of the Gauss-Seidel method in that it uses corrected intermediate results immediately upon them becoming available. In addition, if the F_p are linear functions of the x_j this method reduces to the S.O.R. method.

The convergence criteria for the N.L.O.R. method can be shown to be the same as that for the S.O.R. method and is given in Section (3.6) with the coefficient matrix A replaced by the Jacobian matrix of the equations (3.5.33), $J(f_{pq}^{(k)})$, where

$$J(f_{pq}^{(k)}) = [\frac{\partial f_p^{(k)}(x)}{\partial x_q}] \ , \ p,q = 1,2,\ldots,m. \qquad (3.5.35)$$

This is accomplished by use of the Taylor Series so that the vector form for $\underline{x}^{(k+1)}-\underline{x}^{(k)}$ is

$$\underline{x}^{(k+1)}-\underline{x}^{(k)} = -\omega[D_k^{-1}L_k\underline{e}^{(k+1)} + (I+D_k^{-1}U_k)\underline{e}^{(k)}] \ , \qquad (3.5.36)$$

where $\underline{e}^{(k)}=\underline{x}^{(k)}-\underline{x}$ is the error vector and L_k, D_k and U_k are the lower triangular, diagonal and upper triangular component matrices of the Jacobian matrix respectively, i.e.,

$$J(f_{pq}^{(k)}) = L_k + D_k + U_k \ . \qquad (3.5.37)$$

For convergence, the Jacobian matrix (3.5.35), at each stage of the iteration, must have the same properties required for A in the S.O.R. method, e.g. Property (A). (see Ames (1965), p.408).

3.6 THE BASIC CONVERGENCE CRITERION

Definition 3.6.1

The iterative method (3.5.32) converges if

$$\lim_{k\to\infty} u_i^{(k)} = u_i \ , \quad \text{for all } i \ , \tag{3.6.1}$$

and for all starting vectors $\underline{u}^{(0)}$.

Theorem 3.6.1

An iterative method which can be expressed in the form of equation (3.5.32) converges if and only if $\rho(G)<1$.

Proof

We define the error vector after k iterations to be

$$\underline{e}^{(k)} = \underline{u}^{(k)} - \underline{u} \ , \tag{3.6.2}$$

where $\underline{u}$ is the unique vector solution of (3.5.1), and assume that the iterative method is consistent, i.e.,

$$\underline{u} = G\underline{u} + \underline{r} \ , \tag{3.6.3}$$

then from (3.5.32) and (3.6.3) we obtain,

$$\underline{e}^{(k+1)} = G\underline{e}^{(k)} \ , \tag{3.6.4}$$

and hence,

$$\underline{e}^{(k)} = G\underline{e}^{(k-1)} = G^2\underline{e}^{(k-2)} = \ldots = G^k\underline{e}^{(0)} \ , \tag{3.6.5}$$

where $\underline{e}^{(0)}$ is the error vector associated with the initial vector $\underline{u}^{(0)}$. We require the conditions under which $\underline{e}^{(k)} \to \underline{0}$ as $k\to\infty$. From equation (3.6.5), this can happen if and only if $G^k \to (0)$ (the null matrix) as $k\to\infty$. By Theorem (2.6.2), this will be true if and only if $\rho(G)<1$, which completes the proof.

Corollary 3.6.1

A sufficient condition for convergence of (3.5.32) is merely that

$$||G|| < 1 \ , \tag{3.6.6}$$

since $\rho(G) \leq ||G||$ (see Chapter 2, equation (2.4.18)).

It is not a necessary condition because it may happen in some cases that $||G||>1$ but $\rho(G)<1$ which guarantees the convergence of the iteration process according to Theorem (3.6.1). On the other hand as confirmed by Theorem (3.6.1), the convergence of (3.5.32) is totally independent of the choice of the initial vector $\underline{u}^{(0)}$, as long as the matrix A in (3.5.1) is non-singular, whilst it is dependent on $\underline{u}^{(0)}$ if A is singular (Meyer and Plemmons, (1977)).

3.7 RATE OF CONVERGENCE

In practical computation, even if an iterative method converges, it may converge too slowly to be of practical value. Hence, it is essential to evaluate the effectiveness of an iterative method. To carry out this we should consider both the work required per iteration and the number of iterations required for convergence to a specified accuracy. For the latter and normally in practice, the usual approach is to iterate until the norm of the error vector $\underline{e}^{(k)}$ is reduced to less than some predetermined factor, say ε, of the norm of the initial vector $\underline{e}^{(0)}$. From (3.6.5) we have,

$$||\underline{e}^{(k)}|| = ||G^k\underline{e}^{(0)}|| \leq ||G^k||.||\underline{e}^{(0)}||. \tag{3.7.1}$$

Then, if $\underline{e}^{(0)} \neq \underline{0}$,

$$||\underline{e}^{(k)}||/||\underline{e}^{(0)}|| \leq ||G^k|| . \tag{3.7.2}$$

We require, $$||\underline{e}^{(k)}|| \leq \varepsilon||\underline{e}^{(0)}||, \tag{3.7.3}$$

where $||.||$ denotes $||.||_2$ as defined in Chapter 2. By Theorem (3.6.1) we know that $||G^k||\to 0$ as $k\to\infty$ if and only if $\rho(G)<1$. Hence, equation (3.7.3) can be satisfied by choosing k sufficiently large that,

$$||G^k|| \leq \varepsilon . \tag{3.7.4}$$

If k is large enough so that $||G^k||<1$, it follows that (3.7.4) is equivalent to, $$k \geq -\log\varepsilon/(-\frac{1}{k}\log||G^k||) , \tag{3.7.5}$$

and from this inequality we can obtain a lower bound for the number of iterations for the iterative methods (3.5.32).

Definition 3.7.1

For any convergent iterative method of the form (3.5.32), the quantity, $$R_k(G) = \frac{-\log||G^k||}{k} , \tag{3.7.6}$$

is the *average rate of convergence* after k iterations.

If $R_k(G_1)<R_k(G_2)$, then G_2 is iteratively faster for k iterations than G_1.

Definition 3.7.2

The *asymptotic average rate of convergence* is defined by,

$$R(G) = \lim_{k\to\infty} R_k(G) = -\log\rho(G) \; . \tag{3.7.7}$$

The latter equality holds, since,

$$\rho(G) = \lim_{k\to\infty} (||G^k||)^{1/k}, \tag{3.7.8}$$

which is a result proved by (Young (1971), p.87). We shall refer to R(G) as the *rate of convergence*.

It is normal for iterative processes to converge slowly in substantial problems corresponding to $\rho(G)$ only slightly less than 1, and a rate of convergence nearly 0.

Now, to obtain a crude estimate of the number of iterations, k, upon replacing $||G^k||$ by $[\rho(G)]^k$ in equation (3.7.5) we see that $\varepsilon \simeq [\rho(G)]^k$, and hence,

$$k \simeq \frac{-\log\,\varepsilon}{-\log\rho(G)} = \frac{-\log\,\varepsilon}{R(G)} \; . \tag{3.7.9}$$

However, the value of k obtained from (3.7.9) could be very much lower when compared with the number required, in which $||G^k||$ will behave like $k\rho(G)^{k-1}$, rather than $\rho(G)^k$ (see Young, (1971), p.88). In this case, the smallest value of k such that,

$$\boxed{k[\rho(G)]^{k-1} \leq \varepsilon} \tag{3.7.10}$$

estimates the number of iterations required.

3.8 CONVERGENCE THEOREMS FOR BASIC ITERATIVE METHODS

The basic criterion, regarding convergence, as shown in Section (3.6) is that the relevant iteration matrix G must have spectral radius $\rho(G)$ less than 1. Unfortunately, this test is difficult to apply in practice for large matrices, since the determination of the largest eigenvalue of G might well involve more computation than the actual solution of the equations. However, in this section we will state some convergence theorems for the basic iterative methods which were considered in Section (3.5).

Theorem 3.8.1

Let A be a strictly or irreducibly diagonally dominant ($m \times m$) complex matrix. Then, both the associated point Jacobi and the point GS matrices are convergent, and the Jacobi iterative method and GS iterative method for the matrix problem $A\underline{u}=b$ are convergent for any initial approximation vector $\underline{u}^{(0)}$.

Proof: (see Varga (1962), p.73).

Theorem 3.8.2: (The Stein-Rosenberg comparison theorem, 1948)

If the system $A\underline{u}=\underline{b}$ has a Jacobi iteration matrix $B=L+R$ which contains no negative elements, and if $\mathcal{L}$ is the GS iteration matrix, then one and only one of the following mutually exclusive relations holds:

(a) $\rho(B) = \rho(\mathcal{L}) = 0$,

(b) $0<\rho(\mathcal{L})<\rho(B)<1$

(c) $\rho(B) = \rho(\mathcal{L}) = 1$,

or (d) $1<\rho(B)<\rho(\mathcal{L})$.

Proof: (see Varga (1962), p.70).

The contents of this theorem is that the Jacobi iteration matrix and the GS iteration matrix are either both convergent or both divergent. Furthermore, if the matrix B is non-negative and $0<\rho(B)<1$, then the GS iterative method is asymptotically faster than the Jacobi iterative method. Hence,

$$R(L) > R(B) \ . \tag{3.8.1}$$

Notice the importance of the two conditions, B contains no negative elements and $0<\rho(B)<1$, for the results given above, since sometimes it happens that the Jacobi iterative method might converge and the GS iterative method diverge. We can illustrate this by the following example, (Fox, (1964), p.194).

The matrix,

$$A = \begin{bmatrix} 1 & 0 & 1 \\ -1 & 1 & 0 \\ 1 & 2 & -3 \end{bmatrix} ,$$

gives rise to the iteration matrices,

$$B = \begin{bmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ -\frac{1}{3} & -\frac{2}{3} & 0 \end{bmatrix} \quad \text{and} \quad L = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} .$$

If λ_i and ξ_i are the eigenvalues of the matrices B and L respectively, for i=1,2,3, then,

$$\lambda_1 = 0.748, \quad \lambda_{2,3} = -0.374 \pm 0.868i$$

$$\xi_1 = 1, \qquad \xi_2 = 0, \qquad \xi_3 = 0$$

so that the Jacobi process converges, very slowly and the GS process diverges.

But in general the GS method is superior to the Jacobi method and its superiority is given by the following theorem:

Theorem 3.8.3

If A is symmetric, positive definite matrix, then its GS iteration matrix $L=(D-E)^{-1}F$ has spectral radius less than unity, (i.e. in this case the GS iterative method always converges).

Proof: (see Lieberstein (1968), Fox (1964) and Goult et al (1974)).

In Lieberstein (1968), p.62, there is a given counter example which verifies the invalidity of Theorem (3.8.3) for the Jacobi process, i.e. although the matrix A is symmetric and positive definite, its iteration matrix $D^{-1}(E+F)$ may have eigenvalue(s) greater than 1 in modulus.

Theorem 3.8.4

If the Jacobi method converges, then the J.O.R. method converges for $0<\omega\leq 1$.

Proof: (see Young (1971), p.107).

Theorem 3.8.5

Let A be an irreducible matrix with weak diagonal dominance.
Then,

(a) The Jacobi method converges and the J.O.R. method converges for $0<\omega\leq 1$.

(b) The GS method converges, and the S.O.R. method converges for $0<\omega\leq 1$.

Proof: (See Young (1971), p.108).

So far, the Jacobi and GS iterative methods were compared in terms of their spectral radii, and we saw how the matrix property of being irreducible with weak diagonal dominance was a sufficient condition for the convergence of the Jacobi and GS iterative methods. Whereas $\rho(G)<1$ is both a necessary and sufficient condition that G be a convergent matrix. Next we give different necessary and sufficient conditions for the convergence of the S.O.R. and GS iterative methods. We commence with the following theorem of Kahan (1958).

Theorem 3.8.6

If L_ω is the S.O.R. iteration matrix then,

$$\rho(L_\omega) \geq |\omega-1| \ , \tag{3.8.2}$$

for all real ω. Moreover, if the S.O.R. method converges, then,

$$0<\omega<2 \ . \tag{3.8.3}$$

Proof

If η_i are the eigenvalues of L_ω, then from (2.4.20) we have,

$$\det(L_\omega) = \prod_{i=1}^{m} \eta_i \ . \tag{3.8.4}$$

Hence, by (3.5.26) we have,

$$\det(L_\omega) = \det[(I-\omega L)^{-1}(\omega R+(1-\omega)I]$$
$$= \det(I-\omega L)^{-1} \det[\omega R+(1-\omega)I] .$$

But $(I-\omega L)$ is a lower triangular matrix with diagonal elements equal one and $\omega R+(1-\omega)I$ is an upper triangular matrix with diagonal elements $1-\omega$. Hence,

$$\det(L_\omega) = 1.(1-\omega)^m \ . \tag{3.8.5}$$

Also, from (3.8.4) and (3.8.5) we have,

$$\prod_{i=1}^{m} \eta_i = (1-\omega)^m \ . \tag{3.8.6}$$

But,

$$\rho(L_\omega) = \max_i |\eta_i| \ , \tag{3.8.7}$$

hence,

$$\rho(L_\omega) \geq [|1-\omega|^m]^{1/m} \ ,$$

which implies,

$$\rho(L_\omega) \geq |1-\omega| \ .$$

Now, if the S.O.R. method converges, then from Theorem (3.6.1)

$$\rho(L_\omega) < 1 \ ,$$

$$\Rightarrow \quad |1-\omega|<1,$$

and this implies $0<\omega<2$.

Theorem 3.8.7

Let A be a symmetric matrix with positive diagonal elements. Then, the S.O.R. method converges if and only if A is positive definite and $0<\omega<2$.

Proof: (See Young, (1971), p.113, or Varga (1962), p.77)

Corollary 3.8.1

Let A be a symmetric matrix with positive diagonal elements. Then, the GS method converges if and only if A is positive definite.

Now, the following theorem is an extension of the result of Theorem (3.8.2). (The Theorem and the Corollaries which follow are given in Young (1971).

Theorem 3.8.8

If A is an L-matrix and if $0<\omega\leq 1$, then:

(a) $\rho(B)<1$, if and only if $\rho(\mathcal{L}_\omega)<1$,

(b) $\rho(B)<1$ (and $\rho(\mathcal{L}_\omega)<1$) if and only if A is an M-matrix,

if $\rho(B)<1$, then,

$$\rho(\mathcal{L}_\omega) \leq (1-\omega+\omega\rho(B)) \ , \tag{3.8.8}$$

(c) if $\rho(B)\geq 1$ and $\rho(\mathcal{L}_\omega)\geq 1$, then,

$$\rho(\mathcal{L}_\omega) \geq (1-\omega+\omega\rho(B)) \geq 1. \tag{3.8.9}$$

Corollary 3.8.2

If A is an L-matrix, then:

(a) $\rho(B)<1$, if and only if $\rho(\mathcal{L})<1$,

(b) $\rho(B)<1$ and $\rho(\mathcal{L})<1$, if and only if A is an M-matrix,

if $\rho(B)<1$, then,

$$\rho(\mathcal{L}) < \rho(B) \tag{3.8.10}$$

(c) if $\rho(B) \geq 1$ and $\rho(\mathcal{L}) \geq 1$, then,

$$\rho(\mathcal{L}) \geq \rho(B) . \tag{3.8.11}$$

Corollary 3.8.3

If A is a Stieltjes matrix, then the Jacobi iterative method converges and the J.O.R. iterative method converges for $0<\omega\leq 1$. [See definition (2.5.2) for the L-matrix, M-matrix and Stieltjes matrix].

For the S.S.O.R. iterative method, the eigenvalues of its iteration matrix H_ω, are real and non-negative and the method is convergent for $0<\omega<2$. In other words, if $\rho(H_\omega)<1$, then $0<\omega<2$.

In practice the S.S.O.R. iterative method is not as efficient as the normal S.O.R. method, (Jennings, (1977). However, it is superior in problems with limited application, (Habetler and Wachspress, (1961)).

3.9 DETERMINATION OF THE OPTIMUM RELAXATION FACTOR AND COMPARISON OF RATES OF CONVERGENCE

The determination of a suitable value for the relaxation factor ω of the S.O.R. method is of paramount importance, and in particular the optimum value of ω, denoted by ω_b, which minimises the spectral radius of the S.O.R. iteration matrix and thereby maximise the rate of convergence of the method. For an arbitrary set of linear equations, no formula exists for the determination of ω_b, though a simple but time consuming procedure for estimating ω_b is to run the problem on a computer for a range of values of ω to obtain some idea of the value which gives the most rapid convergence. 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 of the S.O.R. iteration matrix L_ω associated with A are related to the eigenvalues μ of the corresponding Jacobi iteration matrix B of A by the equation,

$$\frac{(\lambda+\omega-1)^2}{\lambda} = \omega^2\mu^2 \quad , \tag{3.9.1}$$

from this equation it can be seen that,

$$\lambda-\omega\mu\lambda^{\frac{1}{2}}+\omega-1 = 0 \ . \tag{3.9.2}$$

If we assume that A is symmetric, then, the eigenvalues of B are real and occur in pairs $\pm\mu$. Let $\bar{\mu}$ denote the largest eigenvalue of B, it can be shown, (see Young, (1954)), that ω_b defined by,

$$\omega_b^2\,\bar{\mu}^2 = 4(\omega_b-1), \quad 1\leq\omega_b<2 \ , \tag{3.9.3a}$$

or equivalently,

$$\omega_b = \frac{2}{1+\sqrt{1-\bar{\mu}^2}} \quad , \tag{3.9.3b}$$

is the value of ω which minimises $\rho(L_\omega)$, i.e. $\omega \neq \omega_b$, then,

$$\rho(L_\omega) > \rho(L_{\omega_b}) \quad . \tag{3.9.4}$$

Using this optimum value of ω, it can also be shown that,

$$\rho(L_{\omega_b}) = \frac{1-\sqrt{1-\bar{\mu}^2}}{1+\sqrt{1-\bar{\mu}^2}} = \omega_b - 1 . \tag{3.9.5}$$

Moreover, for any ω in the range $0<\omega<2$, we have,

$$\rho(L_\omega) = \begin{cases} \left[\dfrac{\omega\bar{\mu}+(\omega^2\bar{\mu}^2-4(\omega-1))^{\frac{1}{2}}}{2}\right]^2, & \text{if } 0<\omega\leq\omega_b \\ \omega-1 & , \text{if } \omega_b\leq\omega<2 . \end{cases} \tag{3.9.6}$$

For the Gauss-Seidel iterative method ($\omega=1$), equation (3.9.2) gives,

$$\rho(L) = [\rho(B)]^2 = (\bar{\mu})^2 \quad . \tag{3.9.7}$$

Using (3.9.7) it is clear from (3.9.3b) that,

$$\omega_b = \frac{2}{1+\sqrt{1-\rho(L)}} \quad . \tag{3.9.8}$$

The estimation of ω_b depends on whether $\rho(B)$ or $\rho(L)$ can be estimated. Several methods have been suggested by (Carre, (1961), (Varga, (1962)) and (Hageman and Kellogg, (1968)), one of which is the power method that can be described as follows:-

Assuming the matrix of the finite difference equations is consistently ordered and has Property (A), calculate the sequence of approximations $\underline{u}^{(1)}, \underline{u}^{(2)}, \ldots, \underline{u}^{(k)}$, to the solution of the system of equations $A\underline{u}=\underline{b}$ by the Gauss-Seidel method and then we have,

$$\rho(L) = \lim_{k\to\infty} \frac{||\underline{d}^{(k)}||}{||\underline{d}^{(k-1)}||} \quad , \tag{3.9.9}$$

where $\underline{d}^{(k)}$ is defined as,

$$\underline{d}^{(k)} = \underline{u}^{(k)} - \underline{u}^{(k-1)} \quad , \tag{3.9.10}$$

and $$||\underline{d}^{(k)}|| = \{\sum_{j=1}^{n}(u_j^{(k)}-u_j^{(k-1)})^2\}^{\frac{1}{2}} \quad . \tag{3.9.11}$$

Hence, using the power method we can determine an approximate value of $\rho(L)$, which, in turn, can be substituted into equation (3.9.8) to give an estimate of the optimum acceleration factor, ω_b.

For Poisson's equation over a rectangle of sides ph and qh, with Dirichlet boundary conditions, it can be shown, see (Smith, (1978),p.252), that $\rho(B)$ for the five point difference approximation, using a square mesh of side h, is,

$$\rho(B) = \frac{1}{2}(\cos\frac{\pi}{p} + \cos\frac{\pi}{q}) \quad . \tag{3.9.12}$$

Hence for the model problem of Laplace's equation in the unit square with m^2 internal mesh points,

$$\rho(B) = \bar{\mu} = \cos\frac{\pi}{m+1} = \cos\pi h. \tag{3.9.13}$$

In this case, if we choose $h = \frac{1}{20}$ then $\bar{\mu} \cong 0.9877$, it is interesting to see the behaviour of $\rho(L_\omega)$ as a function of ω. It can be shown that as ω increases from 0 to 1, $\rho(L_\omega)$ decreases very slowly from 1 to $\bar{\mu}^2 \cong 0.9755$. As ω increases further $\rho(L_\omega)$ decreases slightly more rapidly until ω gets close to $\omega_b \cong 1.72945$ at which point the decrease is very rapid such that, as $\omega \to \omega_b-$ the slope of $\rho(L_\omega)$ approaches $-\infty$ (see Young, (1971), p.201-202). The value of $\rho(L_\omega)$ for $\omega=\omega_b$ is $\omega_b-1\cong 0.72945$. As ω increases further, $\rho(L_\omega)$ increases linearly to a value of unity when $\omega=2$. This is best seen by plotting $\rho(L_\omega)$ as a function of ω for $0\leq\omega\leq 2$, as shown in Figure (3.9.1).

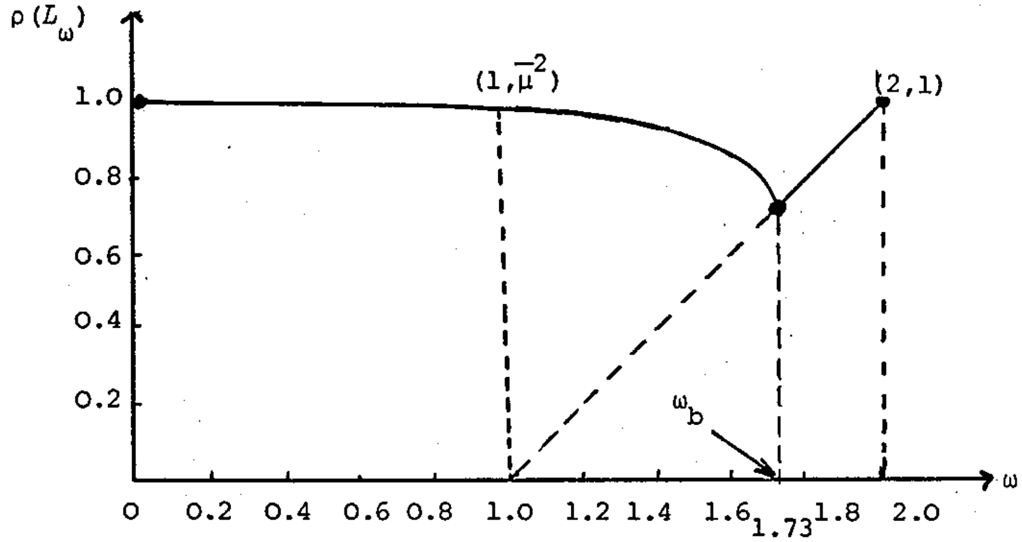


FIGURE (3.9.1)

In a small number of rather specialised cases $\rho(B)$ can be calculated quite easily, for example, if A is an $(m \times m)$ tridiagonal matrix whose Jacobi matrix is

$$B = \begin{bmatrix} 0 & b & & & \\ a & 0 & b & & \bigcirc \\ & a & 0 & b & \\ & & \ddots & \ddots & b \\ \bigcirc & & & a & 0 \end{bmatrix}$$

it is known that,

$$\rho(B) = 2\sqrt{ab} \cos \frac{\pi}{(m+1)} \quad . \tag{3.9.14}$$

Direct comparisons of the Jacobi, Gauss-Seidel and S.O.R. iterative methods can be made when the matrix A has Property (A) and is consistently ordered. From equation (3.9.7) and by using the definition of the rate of convergence given in (3.7.7), we obtain,

$$R(L) = 2R(B) \quad . \tag{3.9.15}$$

Thus, the rate of convergence of the GS method is twice that of the

Jacobi method. By use of equation (3.9.5) it can be verified that, asymptotically as $\rho(B)$ tends to zero,

$$R(\mathcal{L}_{\omega_b}) \simeq 2\sqrt{R(\mathcal{L})}\ . \tag{3.9.16}$$

For the model problem of the Laplace equation in the unit square with m^2 internal mesh points, (Varga, (1962), p.203-204) gives,

$$\left.\begin{aligned}
&\text{(i)} \quad R(B) \approx \frac{\pi^2}{2(m+1)^2} \approx \frac{1}{2}\pi^2 h^2\ , \\
&\text{(ii)} \quad R(\mathcal{L}) \simeq \frac{\pi^2}{(m+1)^2} \simeq \pi^2 h^2\ , \\
&\text{(iii)} \quad R(\mathcal{L}_{\omega_b}) \simeq \frac{2\pi}{(m+1)} \simeq 2\pi h,
\end{aligned}\right\} \begin{matrix} m\to\infty \\ (h\to 0) \end{matrix} \tag{3.9.17}$$

therefore,

$$\frac{R(\mathcal{L}_{\omega_b})}{R(B)} \simeq \frac{4(m+1)}{\pi} \simeq \frac{4}{\pi h}\ ,\ \begin{matrix} m\to\infty \\ (h\to 0) \end{matrix} \tag{3.9.18}$$

where, $\quad h = \dfrac{1}{m+1}$.

Thus, for m large, the S.O.R. iterative method with optimum relaxation factor is an order of magnitude faster than the Jacobi iterative method for the model problem, and,

$$\frac{R(\mathcal{L}_{\omega_b})}{R(\mathcal{L})} \simeq \frac{2(m+1)}{\pi} \simeq \frac{2}{\pi h}\ ,\ h\to 0\ . \tag{3.9.19}$$

This shows the superiority of the S.O.R. iterative method over the GS iterative method by an order of magnitude in h^{-1}.

3.10 OTHER POINT ITERATIVE METHODS

In this section we will briefly discuss some other iterative methods for solving the system,

$$A\underline{u} = \underline{b} \quad , \tag{3.10.1}$$

which have been developed and successfully applied.

3.10.1 Simultaneous Displacement Method (Gradient Method)

The Jacobi iterative method described in Section (3.5) can be extrapolated to give a faster rate of convergence when the coefficient matrix A (of order m) in (3.10.1) is real and positive definite. We assume, without loss of generality, that all the diagonal elements of A are unity. Hence, the Jacobi iterative method,

$$D\underline{u}^{(k+1)} = (E+F)\underline{u}^{(k)} + \underline{b} \ , \tag{3.10.2}$$

can be written as,

$$\underline{u}^{(k+1)} = H\underline{u}^{(k)} + \underline{b} \ , \ (\text{since } D^{-1}=I) \ , \tag{3.10.3}$$

where D,E and F as defined in Section (3.5) and by assuming that $H=D^{-1}(E+F)=E+F$. Alternatively, equation (3.10.3) can be written as,

$$\begin{aligned} \underline{u}^{(k+1)} &= (I-A)\underline{u}^{(k)} + \underline{b} \ , \\ &= \underline{u}^{(k)} + \underline{q}^{(k)} \ , \end{aligned} \tag{3.10.4}$$

which shows that the change in the solution vector is equal to the kth residual vector $\underline{q}^{(k)}=\underline{b}-A\underline{u}^{(k)}$. Equation (3.10.4) can then be extrapolated by multiplying $\underline{q}^{(k)}$ by a positive constant α, to be determined later, giving rise to the iterative method,

$$\underline{u}^{(k+1)} = (I-\alpha A)\underline{u}^{(k)} + \alpha\underline{b} \ , \tag{3.10.5}$$

so that the error vector after k+1 iterations $\underline{e}^{(k+1)}=\underline{u}^{(k+1)}-A^{-1}\underline{b}$ satisfies the recurrence relationship,

$$\underline{e}^{(k+1)} = (I-\alpha A)\underline{e}^{(k)} \ , \tag{3.10.6}$$

which is a linear stationary iteration with iteration matrix $G=I-\alpha A$. Hence, for convergence, we require,

$$\rho(I-\alpha A) < 1 \ . \qquad (3.10.7)$$

If we denote the eigenvalues of A by λ_i, $1\leq i\leq m$, then the condition (3.10.7) can be written as,

$$|1-\alpha\lambda_i| < 1 \ , \ 1\leq i\leq m, \qquad (3.10.8)$$

which gives the permissible range of values of α to be,

$$0 < \alpha < \frac{2}{\max_i \lambda_i} \ . \qquad (3.10.9)$$

However, to obtain the fastest possible convergence, we must choose α so that,

$$\max_i |1-\alpha\lambda_i| = \min \ . \qquad (3.10.10)$$

Very often we know that the λ_i lie in an interval $a\leq\lambda_i\leq c$ for all i ($0<a<c<\infty$). For every α the function $|1-\alpha\lambda|$ takes its maximum at one of the end points $\lambda=a$ or $\lambda=c$. So the best choice of α is that one for which,

$$\max(|1-\alpha a|,|1-\alpha c|) \text{ is smallest.}$$

Clearly, it is the one for which,

$$1-\alpha a = -(1-\alpha c),$$

i.e.,

$$\alpha = \frac{2}{a+c} \ . \qquad (3.10.11)$$

With this choice of α we have, for all i,

$$|1-\alpha\lambda_i| \leq \frac{c-a}{c+a} = \frac{P-1}{P+1} < 1 \ , \qquad (3.10.12)$$

where $P = \frac{c}{a}$ is the *P-condition number* of A, defined by Todd (1950), as the ratio of the maximum eigenvalue to the minimum eigenvalue of a positive definite matrix.

3.10.2 Richardson's Method

Richardson (1910) presented a method for solving the linear system (3.10.1) when A is a positive definite matrix. In this method the relaxation factor α of equation (3.10.5) may depend upon the iteration number k, so that $\alpha=\alpha_k$, giving rise to the iterative method,

$$\underline{u}^{(k+1)} = (I-\alpha_k A)\underline{u}^{(k)} + \alpha_k \underline{b} . \tag{3.10.13}$$

The iteration matrix $(I-\alpha_k A)$ changes at each iteration and so the rate of convergence of the method cannot be found directly. By (3.10.6) we see that,

$$\underline{e}^{(k+1)} = (I-\alpha_k A)\underline{e}^{(k)} .$$

We have, upon successive application, the expression,

$$\underline{e}^{(k+1)} = [\prod_{i=0}^{k} (I-\alpha_i A)]\underline{e}^{(0)}$$

$$= Q_{k+1}(A)\underline{e}^{(0)}, \quad \text{(say)} , \tag{3.10.14}$$

where,
$$Q_{k+1} = \prod_{i=0}^{k} (I-\alpha_i A) , \tag{3.10.15}$$

is a polynomial of degree (k+1) in A such that $Q_{k+1}(0)=1$ and $Q_{k+1}(\frac{1}{\alpha_i})=0$.

This process is non-stationary, since the error operator changes for each iteration.

If the eigenvalues of A are λ_i, $1\leq i\leq m$ as before, with corresponding eigenvectors $\underline{v}_i$, then the eigenvalues and the eigenvectors of $Q_{k+1}(A)$ are $Q_{k+1}(\lambda_i)$ and $\underline{v}_i$ respectively. Let us represent $\underline{e}^{(0)}$ in the basis of eigenvectors $\underline{v}_i$ of A then,

$$\underline{e}^{(0)} = \sum_{i=1}^{m} \gamma_i \underline{v}_i , \tag{3.10.16}$$

the γ_i being arbitrary, hence, from equation (3.10.14)

$$\underline{e}^{(k+1)} = \sum_{i=1}^{m} \gamma_i Q_{k+1}(\lambda_i)\underline{v}_i . \tag{3.10.17}$$

For the error vector $\underline{e}^{(k+1)}$ to be small, $Q_{k+1}(x)$ must be small for x through the interval $0<a\leq x\leq c<\infty$. The precise problem may be formulated in the following manner:

Find $Q_{k+1}(x)$ defined for $a\leq x\leq c$ with $Q_{k+1}(0)=1$, such that,

$$\max_{a\leq x\leq b} |Q_{k+1}(x)| = \min . \tag{3.10.18}$$

Such a polynomial has been given by (Markoff, (1916)), and is defined by,

$$Q_{k+1}(x) = \frac{T_{k+1}[\frac{c+a-2x}{c-a}]}{T_{k+1}[\frac{c+a}{c-a}]} , \tag{3.10.19}$$

where T_k is a Chebyshev polynomial of degree k, defined by,

$$\begin{aligned} T_k(x) &= \cos[k\cos^{-1}x] \\ &= \tfrac{1}{2}[(x+\sqrt{x^2-1})^k+(x-\sqrt{x^2-1})^k] \end{aligned} \tag{3.10.20}$$

adjusted to the interval [-1,1].

Hence, $Q_{k+1}(x)$ is a Chebyshev polynomial of degree (k+1) over the interval [a,c], and scaled so that $Q_{k+1}(0)=1$. Equation (3.10.19) can be written as,

$$Q_{k+1} = \frac{T_{k+1}(y)}{T_{k+1}(y_0)} , \tag{3.10.21}$$

where $y = \frac{c+a-2x}{c-a}$ and $y_0 = \frac{c+a}{c-a} = \frac{P+1}{P-1} > 1$ (y_0 is the value of y when x=0)

$\Rightarrow$ y=1 for x=a and y=-1 for x=c.

Now, since the maximum absolute value of the numerator of equation (3.10.21) is unity, equation (3.10.19) can be written as,

$$\max_{a\le x\le c} |Q_{k+1}(x)| = \frac{1}{T_{k+1}(y_0)} = [T_{k+1}(y_0)]^{-1} , \tag{3.10.22}$$

and,

$$T_{k+1}(y) = \cos[(k+1)\theta] , \tag{3.10.23}$$

where $\cos\theta=y$, and θ is a complex number. Hence,

$$\begin{aligned} 2T_{k+1}(y) &= e^{i(k+1)\theta} + e^{-i(k+1)\theta} \\ &= (\cos\theta+i\sin\theta)^{k+1} + (\cos\theta-i\sin\theta)^{k+1} \end{aligned}$$

since $\cos\theta=y \Rightarrow i\sin\theta=\sqrt{y^2-1}$,

$$2T_{k+1}(y) = (y + \sqrt{y^2-1})^{k+1} + (y - \sqrt{y^2-1})^{k+1} ,y>1.$$

From (3.10.22) as $k \to \infty$

$$\begin{aligned} \max_{a\le x\le c} |Q_{k+1}(x)| &= \frac{2}{(y_0+\sqrt{y_0^2-1})^{k+1}+(y_0-\sqrt{y_0^2-1})^{k+1}} \le \frac{2}{(y_0+\sqrt{y_0^2-1})^{k+1}} \\ &= 2(y_0-\sqrt{y_0^2-1})^{k+1} , \end{aligned} \tag{3.10.24}$$

and it follows that the eigenvalues $|Q_{k+1}(\lambda_i)|$ of $Q_{k+1}(A)$ are uniformly bounded by $2(y_0-\sqrt{y_0^2-1})^{k+1}$ as $k\to\infty$, and the asymptotic bound to the average rate of convergence is given by $\log(y_0+\sqrt{y_0^2-1})$.

For substantially large problems,

$$y_0 \approx 1 + \frac{2}{P} \quad , \tag{3.10.25}$$

which gives the rate of convergence,

$$R(\text{Richardson's method}) \approx \frac{2}{\sqrt{P}} \quad . \tag{3.10.26}$$

(see Forsythe and Wasow, (1960), p.229).

Now, the relaxation factors α_i are the reciprocals of the zeros of the Chebyshev polynomial $Q_{k+1}(x)$, or equivalently, the reciprocal of the zeros of $T_{k+1}(y)$, where the zeros of $T_{k+1}(y)$ are given by,

$$y_i^{(k+1)} = \cos\left[\frac{(2i-1)\pi}{2k}\right] \quad , \ 1\leq i\leq k \ , \tag{3.10.27}$$

since, $y = \frac{c+a-2x}{c-a}$, then $x = \frac{c+a-y(c-a)}{2}$, and,

$$\alpha_i = \frac{2}{c+a-y^{(k+1)}(c-a)} \quad , \tag{3.10.28}$$

hence, from (3.10.27) and (3.10.28),

$$\alpha_i = \frac{2}{(c+a)-(c-a)\cos\frac{(2i-1)\pi}{2k}} \ , \ 1\leq i\leq k, \tag{3.10.29}$$

which defines the iteration method (3.10.13) for a given a and c.

This choice would ensure the fastest convergence in the absence of rounding errors, but in practice the iteration is very sensitive to rounding errors when $\frac{a}{c}$ is very small (Young and Warlick, (1953)).

3.10.3 Second Order Methods

Consider the following linear stationary iteration of second degree, known as the second order Richardson's iterative method,

$$\underline{u}^{(k+1)} = \underline{u}^{(k)}+\alpha(\underline{b}-A\underline{u}^{(k)})+\beta(\underline{u}^{(k)}-\underline{u}^{(k-1)}) \ , \tag{3.10.30}$$

where the parameters α and β remain constant throughout the iteration and are chosen to provide maximum convergence for this method. The error vector for (3.10.30) satisfies,

$$\underline{e}^{(k+1)} = [(1+\beta)I-\alpha A]\underline{e}^{(k)}-\beta\underline{e}^{(k-1)} \quad , \tag{3.10.31}$$

thus, for each error mode we have,

$$\underline{e}_i^{(k+1)} = [1+\beta-\alpha\lambda_i]\underline{e}_i^{(k)}-\beta\underline{e}^{(k-1)} \quad , \tag{3.10.32}$$

where λ_i are the eigenvalues of A. Let γ_i be the eigenvalues of the matrix associated with the iterative process then,

$$\underline{e}_i^{(k+1)} = \gamma_i\underline{e}_i^{(k)} = \gamma_i^2\underline{e}_i^{(k-1)} \quad , \tag{3.10.33}$$

so from (3.10.32) and (3.10.33) we obtain,

$$\underline{e}_i^{(k+1)} = [1+\beta-\alpha\lambda_i - \frac{\beta}{\gamma_i}]\underline{e}_i^{(k)} \quad , \tag{3.10.34}$$

and the combination of (3.10.33) and (3.10.34) leads to,

$$\gamma_i^2-(1+\beta-\alpha\lambda_i)\gamma_i + \beta = 0 \; , \tag{3.10.35}$$

which yields,

$$\gamma_i = \frac{\delta_i \pm\sqrt{\delta_i^2-4\beta}}{2} \quad , \tag{3.10.36}$$

where $\delta_i = 1+\beta-\alpha\lambda_i$.

Choosing α and β so that $(\delta_i^2-4\beta)<0$ for all λ_i (which means that γ_i will be complex) all $|\gamma_i|$ will be identical. Then, the choices,

$$1+\beta-\alpha a = 2\sqrt{\beta} \quad , \tag{3.10.37a}$$

$$1+\beta-\alpha c = -2\sqrt{\beta} \; , \tag{3.10.37b}$$

lead to,

$$\alpha = \left[\frac{2}{\sqrt{a}+\sqrt{c}}\right]^2 \quad \text{and} \quad \beta = \left[\frac{\sqrt{c}-\sqrt{a}}{\sqrt{c}+\sqrt{a}}\right]^2 \quad , \tag{3.10.38}$$

(See Frankel, (1950)). The spectral radius of the method is $\sqrt{\beta}$ and the rate of convergence

$$R(\text{second order Richardson}) \simeq \frac{2}{\sqrt{\frac{c}{a}}} = \frac{2}{\sqrt{P}} \quad . \tag{3.10.39}$$

Note that in this iteration process we need to store two vector iterants for each iteration and this storage requirement could be severe for large systems of equations.

Another second order method (non-stationary), is the *'Chebyshev* *second order method'* defined by,

$$\underline{u}^{(k+1)} = \underline{u}^{(k)}+\alpha_k(\underline{b}-A\underline{u}^{(k)})+\beta_k(\underline{u}^{(k)}-\underline{u}^{(k-1)}) , \quad (3.10.40)$$

where α_k and β_k are defined by (Stiefel, (1958)) as follows,

$$\alpha_k = \frac{4T_k(\frac{c+a}{c-a})}{(c-a)T_{k+1}(\frac{c+a}{c-a})} \quad \text{and} \quad \beta_k = \frac{T_{k-1}(\frac{c+a}{c-a})}{T_{k+1}(\frac{c+a}{c-a})} . \quad (3.10.41)$$

The error vector is given by,

$$\underline{e}^{(k+1)} = \frac{T_{k+1}(\frac{c+a-2x}{c-a})}{T_{k+1}(\frac{c+a}{c-a})}\underline{e}^{(0)} , \quad (3.10.42)$$

which leads to identical results to those obtained using Richardson's method for the rate of convergence.

Since α_k and β_k are less than unity, round off-errors do not appear and the method is preferred to Richardson's method with the only inconvenient matter raised being the estimation of the extreme eigenvalues a and c.

3.10.4 Semi-Iterative Methods

Consider the completely consistent linear stationary iterative method,

$$\underline{u}^{(k+1)} = G\underline{u}^{(k)} + \underline{r} , \quad (3.10.43)$$

where $\underline{r}=(I-G)A^{-1}\underline{b}$ and (I-G) is non-singular.

Given the sequence $u^{(0)},u^{(1)},u^{(2)},\ldots$ we can often develop another sequence $y^{(1)},y^{(2)},\ldots$ by the theory of summability so that either the new sequence converges when the old one does not, or else the new one converges faster than the old one if the old one converges. Assume that there is given a set of coefficients $\alpha_{k,\ell}$, $\ell=0,1,2,\ldots,k$ such that,

$$\sum_{\ell=0}^{k}\alpha_{k,\ell} = 1 , \quad k=0,1,2,\ldots , \quad (3.10.44)$$

so the new sequence can be considered as a linear combination of the old one. Then the relation,

$$\underline{y}^{(k)} = \sum_{\ell=0}^{k} \alpha_{k,\ell}\underline{u}^{(\ell)} \quad , \; k=0,1,2,\ldots \tag{3.10.45}$$

defines a *semi-iterative method* with respect to the iterative method of equation (3.10.43). If $\alpha_{k,\ell}=0$ for all $\ell<k$ and $\alpha_{k,k}=1$, the semi-iterative method reduces to the original method.

Our aim here is to determine $\alpha_{k,\ell}$ such that the rate of convergence of (3.10.45) is greater than that of (3.10.43). Such a determination is described in (Young and Frank, (1963)).

Let the error vector of the k^{th} iterant be,

$$\underline{e}^{(k)} = \underline{u}^{(k)} - \underline{u} \quad , \tag{3.10.46a}$$

and,

$$\underline{\tilde{e}}^{(k)} = \underline{y}^{(k)} - \underline{u} \; , \tag{3.10.46b}$$

where $\underline{u}$ is the unique vector solution of (3.10.1), hence from equation (3.10.45),

$$\begin{aligned}\underline{\tilde{e}}^{(k)} &= \sum_{\ell=0}^{k} \alpha_{k,\ell}\underline{u}^{(\ell)} - \underline{u} \\ &= \sum_{\ell=0}^{k} \alpha_{k,\ell}u^{(\ell)} - \underline{u}\sum_{\ell=0}^{k}\alpha_{k,\ell} \\ &= \sum_{\ell=0}^{k} \alpha_{k,\ell}\underline{e}^{(\ell)} \quad .\end{aligned} \tag{3.10.47}$$

But from equation (3.6.5),

$$\underline{e}^{(\ell)} = G^{\ell}\underline{e}^{(0)} \quad ,$$

hence,

$$\underline{\tilde{e}}^{(\ell)} = (\sum_{\ell=0}^{k} \alpha_{k,\ell}G^{\ell})\underline{e}^{(0)} \quad . \tag{3.10.48}$$

If we now define a polynomial,

$$P_k(\underline{u}) = \sum_{\ell=0}^{k} \alpha_{k,\ell}\underline{u}^{\ell} \quad , \tag{3.10.49}$$

then, equation (3.10.48) is expressible as,

$$\underline{\tilde{e}}^{(\ell)} = P_k(G)\underline{e}^{(0)} = P_k(G)\underline{\tilde{e}}^{(0)} \quad . \tag{3.10.50}$$

Suppose that the iteration matrix G of the iterative system (3.10.36) has real eigenvalues μ and they lie in the range,

$$a \leqslant \mu \leqslant c < 1 \; , \tag{3.10.51}$$

where a and c are real numbers, then, from equation (3.10.43) we have,

$$||\tilde{\underline{e}}^{(\ell)}|| \leq ||P_k(G)||\ ||\underline{e}^{(0)}|| \quad , \qquad (3.10.52)$$

and since $\rho(P_k(G)) = \max\limits_{a\leq\mu\leq c} |P_k(\mu)|$, it follows that the problem becomes that of minimising,

$$\max_{a\leq\mu\leq c} |P_k(\mu)| \quad , \qquad (3.10.53)$$

such that $\quad P_k(1) = 1$, for all k.

Let us introduce a new variable $\gamma=\gamma(\mu)$ in which for $a\leq\mu\leq c$ we have, $-1\leq\gamma\leq 1$, (i.e. $\gamma(a)=-1, \gamma(c)=1$). So this will define the transformation,

$$\gamma(\mu) = \frac{2\mu-(c+a)}{c-a} \quad , \qquad (3.10.54a)$$

and,

$$\mu = \tfrac{1}{2}[(c-a)\gamma+(c+a)] \quad , \qquad (3.10.54b)$$

so,

$$\gamma(1) = \frac{2-(c+a)}{c-a} = z \text{ (say)} \quad ,$$

then $\quad z>1$.

Let,

$$Q_k(\gamma)_i = P_k\left(\frac{(c-a)\gamma_i+(c+a)}{2}\right) \quad , \qquad (3.10.55)$$

hence,

$$\max_{\alpha\leq\mu\leq\beta}|P_k(\mu)| = \max_{-1\leq\gamma\leq 1} |Q_k(\gamma)| \quad , \qquad (3.10.56)$$

then, the problem is to find a polynomial $Q_k(\gamma)$ such that $Q_k(z)=P_k(1)=1$ and,

$$\max_{-1\leq\gamma\leq 1} |Q_k(\gamma)| = \min \qquad (3.10.57)$$

The answer to this problem is, as before, supplied by the work of (Markoff, (1916)) and is given by,

$$Q_k(\gamma) = \frac{T_k(\gamma)}{T_k(z)} \quad , \qquad (3.10.58)$$

where,

$$T_k(u) = \begin{cases} \cos(k.\cos^{-1}u), & |u|\leq 1 \\ \cosh(k\cosh^{-1}u), & |u|>1 . \end{cases} \qquad (3.10.59)$$

Moreover,

$$\max_{-1\leq\gamma\leq 1} |Q_k(\gamma)| = \frac{1}{T_k(z)} = \frac{1}{T_k[\frac{2-(c+a)}{c-a}]} \quad , \qquad (3.10.60)$$

therefore we have,

$$P_k(\mu) = Q_k\left[\frac{2\mu-(c+a)}{c-a}\right]$$

$$= \frac{T_k\left[\frac{2\mu-(c+a)}{c-a}\right]}{T_k\left[\frac{2-(c+a)}{c-a}\right]} \qquad (3.10.61)$$

Now, since $z>1$ and $T_k(z)>1$, then the method is convergent even though the basic method (3.10.43) may not converge.

Since $T_0(x)=1$, $T_1(x)=x$ and $T_{k+1}(x)=2xT_k(x)-T_{k-1}(x)$ for $k\geq 1$. So, $T_2(x)=2x^2-1$, and we have,

$$P_0(\mu) = 1 \ , \qquad (3.10.62)$$

and $\alpha_{0,0}=1$,

$$P_1(\mu) = \frac{2\mu-(c+a)}{2-(c+a)} \ , \qquad (3.10.63)$$

and,

$$\alpha_{1,0} = -\frac{c+a}{2-(c+a)} \ , \quad \alpha_{1,1} = \frac{2}{2-(c+a)} \ ;$$

$$P_2(\mu) = [2(\frac{2\mu-(c+a)}{c-a})^2-1]/[2(\frac{2-(c+a)}{c-a})^2-1]$$

$$= \alpha_{2,2}\mu^2+\alpha_{2,1}\mu+\alpha_{2,0} \ , \qquad (3.10.64)$$

and,

$$\alpha_{2,0} = \frac{(a+c)^2+4ac}{(a+c)^2+8(1-a-c)+4ac}$$

$$\alpha_{2,1} = \frac{-8(a+c)}{(a+c)^2+8(1-a-c)+4ac}$$

$$\alpha_{2,2} = \frac{8}{(a+c)^2+8(1-a-c)+4ac} \ .$$

The $\{\alpha_{k,\ell}\}$ can be determined by this way for any value of k. However, it is more convenient to develop a relation between $\underline{y}^{(k+1)}, y^{(k)}$, and $y^{(k-1)}$ and determine $y^{(k)}$ directly without finding $u^{(k)}$ first. See (Young, 1971, p.349). We state the final result,

$$\underline{y}^{(k+1)} = 2\left[\frac{2}{b-a}G - (\frac{c+a}{c-a})I\right]\frac{T_k(z)}{T_{k+1}(z)}y^{(k)} - (\frac{T_{k-1}(z)}{T_{k+1}(z)})y^{(k-1)} + \frac{4}{c-a}\frac{T_k(z)}{T_{k+1}(z)}r \qquad (3.10.65)$$

for $k\geq 1$.

Application of a semi-iterative method is similar to S.O.R. with a variable relaxation factor. The additional storage of a vector ($y^{(k)}$ and $y^{(k-1)}$) could be an important consideration limiting the utility of this method in the solution of large problems.

3.11 BLOCK ITERATIVE METHODS

In our previous discussion of the iterative methods for solving a system of linear equations,

$$\sum_{j=1}^{m} a_{i,j}u_j = b_i, \quad (i=1,2,\ldots,m) \tag{3.11.1}$$

or $\qquad A\underline{u} = \underline{b}$,

we dealt with point iterative methods, that is, at each step the approximate solution is modified at a single point of the domain. An extension of these methods are *the block (group) iterative methods* in which several unknowns are connected together in the iteration formula in such a way that a linear system must be solved before any one of them can be determined.

In the system (3.11.1), assume that the equations and the unknows u_i are partitioned into ℓ groups such that u_i, $i=1,2,\ldots,m_1$, constitute the first group; $i=m_1+1,m_1+2,\ldots,m_2$, constitute the second group and in general, u_i, $m_{s-1}<i\leq m_s$, constitute the s^{th} group and $m_\ell=m$. The following definition is due to Young (1971), p.435.

Definition 3.11.1

An *ordered grouping* π of $W=\{1,2,\ldots,m\}$ is a subdivision of W into disjoint subsets $G_1,G_2,\ldots,G_\ell$ such that $G_1 \cup G_2 \cup \ldots G_\ell=W$. Two ordered groupings π and π' defined by $G_1,G_2,\ldots,G_\ell$ and $G'_1,G'_2,\ldots,G'_\ell$, respectively are identical if $\ell=\ell'$ and if $G_1=G'_1$, $G_2=G'_2,\ldots,G_\ell=G'_\ell$.

Evidently, this partitioning π imposes a partitioning of the matrix A into blocks (groups) of the form,

$$A = \begin{bmatrix} A_{1,1} & A_{1,2} & \cdots & A_{1,\ell} \\ A_{2,1} & A_{2,2} & \cdots & A_{2,\ell} \\ \vdots & \vdots & & \vdots \\ A_{\ell,1} & A_{\ell,2} & \cdots & A_{\ell,\ell} \end{bmatrix} , \tag{3.11.2}$$

where the diagonal blocks $A_{i,i}$, $1 \leq i \leq \ell$, are square, non-singular matrices. From this partitioning of the matrix A, we define the matrices,

$$D = \begin{bmatrix} A_{1,1} & & & \\ & A_{2,2} & & O \\ & & \ddots & \\ O & & & A_{\ell,\ell} \end{bmatrix}, \quad E = -\begin{bmatrix} O & & & & O \\ A_{2,1} & O & & & \\ \vdots & \ddots & \ddots & & \\ \vdots & & \ddots & \ddots & \\ A_{\ell,1} & \cdots & \cdots & A_{\ell,\ell-1} & O \end{bmatrix}$$

and,

$$F = -\begin{bmatrix} O & A_{1,2} & \cdots & \cdots & A_{1,\ell} \\ & O & \ddots & & \vdots \\ & & \ddots & \ddots & \vdots \\ & O & & \ddots & A_{\ell-1,\ell} \\ & & & & O \end{bmatrix} \tag{3.11.3}$$

where D is a block diagonal matrix and the matrices E and F are strictly lower and upper block triangular respectively, and

$$A = D-E-F \ . \tag{3.11.4}$$

Now, for the column vector $\underline{u}$ in equation (3.11.1) we define column vectors $\underline{U}_1, \underline{U}_2, \ldots, \underline{U}_\ell$ where $\underline{U}_s$ is formed from $\underline{u}$ by deleting all elements of $\underline{u}$ except those corresponding to group s. Similarly, define column vectors $\underline{C}_1, \underline{C}_2, \ldots, \underline{C}_\ell$ for the given vector $\underline{b}$, we also define the submatrices $A_{i,j}$, for $i,j=1,2,\ldots,\ell$, such that each $A_{i,j}$ are formed from the matrix A by deleting all rows except those corresponding to G_i and all columns except those corresponding to G_j. The system (3.11.1) evidently can be written now in the equivalent form,

$$\sum_{j=1}^{\ell} A_{i,j}\underline{U}_j = \underline{C}_i \quad , \ i=1,2,\ldots,\ell \ . \tag{3.11.5}$$

We shall assume from now on that all submatrices $A_{i,i}$ are non-singular. Then, the *block Jacobi iterative method* is defined by,

$$A_{i,i}\underline{U}_i^{(k+1)} = -\sum_{\substack{j=1 \\ j\neq i}}^{\ell} A_{i,j}\underline{U}_j^{(k)} + \underline{C}_i, \quad 1 \leq i \leq \ell \tag{3.11.6}$$

or equivalently,

$$\underline{U}_i^{(k+1)} = \sum_{\substack{j=1\\ j\neq i}}^{\ell} B_{i,j}\underline{U}_j^{(k)} + \underline{V}_i \quad , \; 1\leq i\leq \ell , \tag{3.11.7}$$

where,

$$\left.\begin{aligned} B_{i,j} &= \begin{cases} -A_{i,i}^{-1}A_{i,j} & , \text{ if } i\neq j \\ 0 & , \text{ if } i=j \end{cases} \\ \text{and} \quad \underline{V}_i &= A_{i,i}^{-1}\underline{C}_i \quad , \; 1\leq i\leq \ell . \end{aligned}\right\} \tag{3.11.8}$$

We may write (3.11.7) in the matrix form,

$$\underline{u}^{(k+1)} = B^{(\pi)}\underline{u}^{(k)} + \underline{v}^{(\pi)} \quad , \tag{3.11.9}$$

where,
$$B^{(\pi)} = (D^{(\pi)})^{-1}(E^{(\pi)} + F^{(\pi)}) \; , \; \textit{"The block Jacobi matrix"} \tag{3.11.10a}$$

and
$$\underline{v}^{(\pi)} = (D^{(\pi)})^{-1}\underline{b} \; , \tag{3.11.10b}$$

$$D^{(\pi)} = \text{diag}_{(\pi)}A,$$

$E^{(\pi)}$ and $F^{(\pi)}$ are again strictly lower and upper triangular matrices.

For the *block Gauss-Seidel iterative method* we have,

$$A_{i,i}\underline{U}_i^{(k+1)} = -\sum_{j=1}^{i-1} A_{i,j}\underline{U}_j^{(k+1)} - \sum_{j=i+1}^{\ell} A_{i,j}\underline{U}_j^{(k)} + C_i , \; 1\leq i\leq \ell \tag{3.11.11}$$

or equivalently,

$$\underline{U}_i^{(k+1)} = \sum_{j=1}^{i-1} B_{i,j}\underline{U}_j^{(k+1)} + \sum_{j=i+1}^{\ell} B_{i,j}\underline{U}_j^{(k)} + \underline{V}_i , \; 1\leq i\leq \ell \tag{3.11.12}$$

where $B_{i,j}$ and $\underline{V}_i$ are as given in (3.11.8). This can also be written in the matrix form,

$$\underline{u}^{(k+1)} = L^{(\pi)}\underline{u}^{(k)} + (D^{(\pi)} - E^{(\pi)})^{-1}\underline{b} \; , \tag{3.11.13}$$

where,
$$L^{(\pi)} = (D^{(\pi)} - E^{(\pi)})^{-1}F^{(\pi)} \; , \textit{"The block GS matrix"} \tag{3.11.14}$$

The matrix form of the block J.O.R. (BJOR) and the block S.O.R. (BSOR) iterative methods are given by (3.11.15) and (3.11.16) respectively,

$$\underline{u}^{(k+1)} = B_\omega^{(\pi)}\underline{u}^{(k)} + \underline{r}^{(\pi)} \; , \tag{3.11.15}$$

$$\underline{u}^{(k+1)} = L_\omega^{(\pi)}\underline{u}^{(k)} + \omega(I-\omega L^{(\pi)})^{-1}\underline{v}^{(\pi)} \; , \tag{3.11.16}$$

where,
$$B_\omega^{(\pi)} = \omega B^{(\pi)} + (1-\omega)I \; , \tag{3.11.17a}$$

$$\underline{r}^{(\pi)} = \omega(D^{(\pi)})^{-1}\underline{b} , \tag{3.11.17b}$$

$$L_\omega^{(\pi)} = (I-\omega L^{(\pi)})^{-1}(\omega R^{(\pi)}+(1-\omega)I) . \tag{3.11.18a}$$

$$L^{(\pi)} = (D^{(\pi)})^{-1}E^{(\pi)} , \tag{3.11.18b}$$

and

$$R^{(\pi)} = (D^{(\pi)})^{-1}F^{(\pi)} . \tag{3.11.18c}$$

$B_\omega^{(\pi)}$ and $L_\omega^{(\pi)}$ are the B.J.O.R. and the B.S.O.R. iteration matrices respectively.

For the convergence of the methods and subsequent analysis we follow the generalization of Young's definition of Property A, consistently ordered matrices, ordering vectors as well as giving some theorems concerning the prospect of the convergence criteria.

First, we define the $(\ell\times\ell)$ matrix $Z = (z_{i,j})$ by,

$$\left.\begin{aligned} z_{i,j} &= 1 \quad , \text{ if } A_{i,j}\neq 0 \\ \text{and} \qquad z_{i,j} &= 0 \quad , \text{ if } A_{i,j}=0, \end{aligned}\right\} \tag{3.11.19}$$

where the matrix A and an ordered grouping π, with ℓ groups, are given.

Definition 3.11.2

The matrix A has *Property* $A^{(\pi)}$ if Z has Property A.

Definition 3.11.3

The matrix A is a π-*consistently ordered matrix* if Z is consistently ordered.

Arms, Gates and Zondek (1956), gave the following definition of Property $A^{(\pi)}$ as a generalization of Young's definition (1954).

Definition 3.11.4

The matrix A has *Property* $A^{(\pi)}$ for a given partition π if there exist two disjoint subsets S_1 and S_2 of W, the set of the first ℓ integers, such that $S_1 \cup S_2 = W$, and if $A_{i,j}\neq 0$, then either $i=j$, or $i \in S_1$ and

and $j \in S_2$ or $i \in S_2$ and $j \in S_1$.

As with Property $A^{(\pi)}$, the following definition is a generalization of Young's definition of an "ordering vector".

Definition 3.11.5

An *ordering ℓ-tuple for A* will be an ℓ-tuple $\underline{v}^{\pi}=(v_1^{\pi},v_2^{\pi},\ldots,v_{\ell}^{\pi})$, where each v_s^{π} is an integer, such that, if $A_{i,j}\neq 0$, and $i\neq j$, then $|v_i^{\pi}-v_j^{\pi}|=1$.

Theorem 3.11.1

A matrix A has Property $A^{(\pi)}$ if and only if there exists an ordering ℓ-tuple for A.

The proof is the same as the proof of Theorem (2.7.2) which is given in Young (1971), p.148.

Theorem 3.11.2

If A is symmetric matrix and $D^{(\pi)}$ is positive definite, then $\rho(L_{\omega}^{(\pi)})<1$ if and only if A is positive definite and $0<\omega<2$.

Theorem 3.11.3

If A has Property $A^{(\pi)}$ and is consistently ordered, with $0<\omega<2$, and if λ is a non-zero eigenvalue of $L_{\omega}^{(\pi)}$, and if μ satisfies,

$$(\lambda+\omega-1)^2 = \lambda\omega^2\mu^2 \quad , \tag{3.11.20}$$

then μ is an eigenvalue of $B^{(\pi)}$. Conversely, if μ is an eigenvalue of $B^{(\pi)}$ and if λ satisfies (3.11.20), then λ is an eigenvalue of $L_{\omega}^{(\pi)}$.

Equation (3.11.20) leads to an explicit formula for the optimal relaxation factor ω_b in terms of $\bar{\mu}$ (the spectral radius of $B^{(\pi)}$) where the matrix A is symmetric, positive definite and has Property $A^{(\pi)}$. The value of ω_b is optimal in the sense that the spectral radius $\bar{\lambda}$ of $L_{\omega}^{(\pi)}$

is minimal so that the convergence rate is greatest. The relations are given by,

$$\omega_b = \frac{2}{1+\sqrt{1-\bar{\mu}^2}} , \tag{3.11.21}$$

and

$$\bar{\lambda} = \rho(L_\omega^{(\pi)}) = \omega_b - 1 . \tag{3.11.22}$$

Further we have,

$$\rho(L_{\omega_b}^{(\pi)}) < \rho(L_\omega^{(\pi)}) , \quad \omega \neq \omega_b , \tag{3.11.23}$$

and, asymptotically, as $\bar{\mu} \to 1$, we obtain the relation,

$$R(L_{\omega_b}^{(\pi)}) \simeq 2\sqrt{R(L^{(\pi)})} = 2\sqrt{2R(B^{(\pi)})} . \tag{3.11.24}$$

To illustrate the foregoing, we consider blocks each consisting of all points on a column (or row), or on two columns (or rows) on our model problem. Methods based on the use of such blocks are called *line iterative methods* and *two-line iterative methods* respectively, they are special cases of a wider class of methods known as the k-line iterative methods. But we shall only consider the first two types. For convenience, let us assume that our model problem has an even number of columns, so that there are an exact number of two-line blocks. We have ℓ line blocks and $\frac{\ell}{2}$ two-line blocks.

The Line Iterative Method

In this method the points are ordered as in Figure (3.2.2) and we consider all the mesh points of a particular line as a block, then for the five-point difference approximation, the resulting coefficient matrix A is block tridiagonal with each diagonal submatrix as a tridiagonal matrix. Hence,

$$A = \begin{bmatrix} \beta_1 & \gamma_1 & & & O \\ \alpha_2 & \beta_2 & \gamma_2 & & \\ & \ddots & \ddots & \ddots & \\ & & & & \gamma_{\ell-1} \\ O & & & \alpha_\ell & \beta_\ell \end{bmatrix} , \tag{3.11.25}$$

where,

$$\beta_r = \begin{bmatrix} 4 & -1 & & & \\ -1 & 4 & -1 & & \bigcirc \\ & \ddots & \ddots & \ddots & \\ & \bigcirc & & & -1 \\ & & & -1 & 4 \end{bmatrix} \quad , \ 1 \leq r \leq \ell$$

and,

$$\alpha_{r+1} = \gamma_r = \begin{bmatrix} -1 & & & \\ & -1 & & \bigcirc \\ \bigcirc & & \ddots & \\ & & & -1 \end{bmatrix} \quad , \ 1 \leq r \leq \ell-1 \ . \tag{3.11.26}$$

The system (3.11.5) can be written in this case as,

$$\begin{bmatrix} \beta_1 & \gamma_1 & & & \\ \alpha_2 & \beta_2 & \gamma_2 & & \bigcirc \\ & \ddots & \ddots & \ddots & \\ \bigcirc & & \ddots & \ddots & \gamma_{\ell-1} \\ & & & \alpha_\ell & \beta_\ell \end{bmatrix} \begin{bmatrix} \underline{U}_1 \\ \underline{U}_2 \\ \vdots \\ \vdots \\ \underline{U}_\ell \end{bmatrix} = \begin{bmatrix} \underline{C}_1 \\ \underline{C}_2 \\ \vdots \\ \vdots \\ \underline{C}_\ell \end{bmatrix} \ . \tag{3.11.27}$$

It is clear that we must solve sub-systems of equations of the form,

$$\alpha_r \underline{U}_{r-1} + \beta_r \underline{U}_r + \gamma_r \underline{U}_{r+1} = \underline{C}_r \quad , \tag{3.11.28}$$

for $\beta_r \underline{U}_r$, then solve,

$$\beta_r \underline{U}_r = \underline{t}_r \ \text{(say)} \qquad , \tag{3.11.29}$$

for $\underline{U}_r$. Where $\alpha_r, \beta_r, \gamma_r$, $1 \leq r \leq \ell$, as in (3.11.26) with $\alpha_1 = \gamma_\ell = 0$ and $\underline{U}_r$ is the column vector of values on the r^{th} line. The solution of equation (3.11.28) by the block S.O.R. iterative method is obtained by solving,

$$\alpha_r \underline{U}_{r-1}^{(k+1)} + \beta_r \hat{\underline{U}}_r^{(k+1)} + \gamma_r \underline{U}_{r+1}^{(k)} = \underline{C}_r \ , \tag{3.11.30}$$

for $\beta_r \hat{U}_r^{(k+1)}$, then solving,

$$\beta_r \hat{\underline{U}}_r^{(k+1)} = \underline{t}_r \ , \tag{3.11.31}$$

for $\hat{\underline{U}}_r^{(k+1)}$, and extrapolating,

$$\underline{U}_r^{(k+1)} = \omega(\hat{\underline{U}}_r^{(k+1)} - \underline{U}_r^{(k)}) + \underline{U}_r^{(k)} \ , \qquad (3.11.32)$$

to obtain the final solution. This particular block iterative method is often called the *successive line overrelaxation method* (*S.L.O.R.*).

Efficient algorithms exist for solving equation (3.11.31), where the matrices β_r have the general form,

$$\beta_r = \begin{bmatrix} b_1 & c_1 & & & \bigcirc \\ a_2 & b_2 & c_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & c_{m-1} \\ \bigcirc & & & a_m & b_m \end{bmatrix} , \ 1 \leq r \leq \ell. \qquad (3.11.33)$$

A well-known algorithm for the problem (based on Gaussian elimination) is as follows:

Compute the initial values, $c_1^* = \dfrac{c_1}{b_1}$,

$$t_1^* = \frac{t_1}{b_1} \ , \quad b_1 \neq 0 \ .$$

Compute recursively,

$$c_i^* = \frac{c_i}{b_i - a_i c_{i-1}^*} \ , \ i=2,3,\ldots,m,$$

$$t_i^* = \frac{t_i - a_i t_{i-1}^*}{b_i - a_i c_{i-1}^*} \ , \ i=2,3,\ldots,m, \qquad (3.11.34a)$$

which transforms the matrix equation to,

$$\begin{bmatrix} 1 & c_1^* & & & \bigcirc \\ & 1 & c_2^* & & \\ & & \ddots & \ddots & \\ & & & 1 & c_{m-1}^* \\ \bigcirc & & & & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ \vdots \\ u_m \end{bmatrix} = \begin{bmatrix} t_1^* \\ t_2^* \\ \vdots \\ \vdots \\ t_m^* \end{bmatrix} . \qquad (3.11.34b)$$

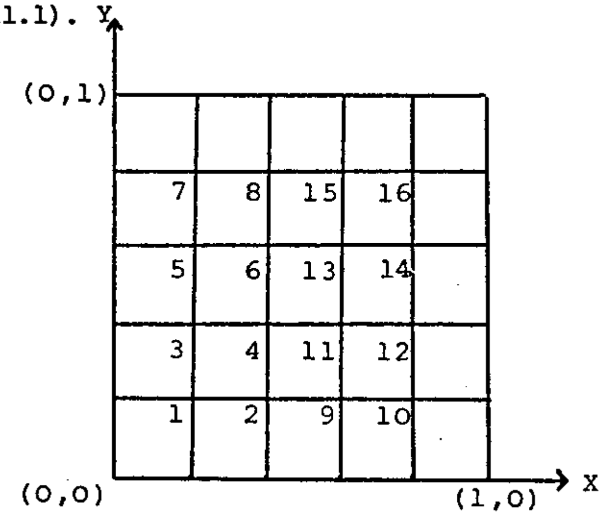
The components u_i of the solution vector $\underline{u}$ can then be computed by the back substitution,

$$u_m = t_m^* \ ,$$

$$u_i = t_i^* - c_i^* u_{i+1} \quad , \ i=m-1,m-2,\ldots,1. \qquad (3.11.34c)$$

<u>The Two-Line Iterative Method</u>

In this method we consider the block as the mesh points lying on two grid lines and the points may be re-ordered among the two lines as shown in Figure (3.11.1).



<u>FIGURE (3.11.1)</u>

Then, for the five-point difference approximation, the resulting coefficient matrix A has a block structure shown in (3.11.25) but we will have $\frac{\ell}{2}$ blocks rather than ℓ blocks, and each block submatrix β_r, $1 \le r \le \frac{\ell}{2}$, is quindiagonal matrix of the form,

$$\beta_r = \begin{bmatrix} 4 & -1 & -1 & & & & & & \\ -1 & 4 & 0 & -1 & & & & \text{O} & \\ -1 & 0 & 4 & -1 & -1 & & & & \\ & -1 & -1 & 4 & 0 & -1 & & & \\ & & \ddots & \ddots & \ddots & \ddots & \ddots & & \\ & \text{O} & & -1 & -1 & 4 & 0 & -1 & \\ & & & & & -1 & 0 & 4 & -1 \\ & & & & & & -1 & -1 & 4 \end{bmatrix} , \quad 1 \le r \le \frac{\ell}{2}$$

$$\alpha_r = \begin{bmatrix} 0 & -1 & & & & & \\ & 0 & 0 & & & & \\ & & \ddots & -1 & & \text{O} & \\ & & & \ddots & \ddots & & \\ & \text{O} & & & \ddots & 0 & \\ & & & & & 0 & -1 \\ & & & & & & 0 \end{bmatrix} , \quad 2 \le r \le \frac{\ell}{2} ,$$

and,

$$\gamma_r = \begin{bmatrix} 0 & & & & & & \\ -1 & 0 & & & & & \\ & 0 & \ddots & & & \text{O} & \\ & & -1 & \ddots & & & \\ & & & \ddots & \ddots & & \\ & \text{O} & & & 0 & 0 & \\ & & & & & -1 & 0 \end{bmatrix} , \quad 1 \le r \le \frac{\ell}{2} - 1 , \qquad (3.11.35)$$

Similar to the line iterative method we can write (3.11.5) in the form of (3.11.27) with ℓ replaced by $\frac{\ell}{2}$, and by following the same steps, equation (3.11.31) can be solved for $\underline{\hat{U}}_r^{(k+1)}$, $1 \le r \le \frac{\ell}{2}$, with β_r a quindiagonal matrix, then by extrapolating we obtain the final solution. This iterative method is known as the *successive two-line overrelaxation method (S.2.L.O.R.)*.

For solving equation (3.11.31), where the matrices β_r have the general form,

$$\beta_r = \begin{bmatrix} c_1 & d_1 & e_1 & & & & & \\ b_2 & c_2 & d_2 & e_2 & & & \text{O} & \\ a_3 & b_3 & c_3 & d_3 & e_3 & & & \\ & a_4 & b_4 & c_4 & d_4 & e_4 & & \\ & & \ddots & \ddots & \ddots & \ddots & \ddots & \\ \text{O} & & a_{m-2} & b_{m-2} & c_{m-2} & d_{m-2} & e_{m-2} & \\ & & & a_{m-1} & b_{m-1} & c_{m-1} & d_{m-1} & \\ & & & & a_m & b_m & c_m & \end{bmatrix} , \quad 1 \le r \le \frac{\ell}{2} , \qquad (3.11.36)$$

we can use the following algorithm (Conte and de Boor, (1972), p.380).

Compute the initial values,

$$d_1^* = \frac{d_1}{c_1} \quad , \; d_m=0, \; b_1=0 \text{ and } c_1 \neq 0 \; ,$$

$$e_1^* = \frac{e_1}{c_1} \quad , \; e_m=e_{m+1}=0$$

$$c_2^* = c_2-b_2d_1^* \; ,$$

$$d_2^* = \frac{d_2-b_2e_1^*}{c_2^*} \; ,$$

$$e_2^* = \frac{e_2}{c_2^*} \quad , \quad c_2^* \neq 0 \quad , \tag{3.11.37a}$$

then compute recursively,

$$b_i^* = b_i-a_id_{i-2}^* \; ,$$

$$c_i^* = c_i-a_ie_{i-2}^*-b_i^*d_{i-1}^* \quad , \quad i=3,4,\ldots,m$$

$$d_i^* = \frac{d_i-b_i^*e_{i-1}^*}{c_i^*} \; ,$$

and

$$e_i^* = \frac{e_i}{c_i^*} \quad , \quad c_i^* \neq 0 \; , \; i=3,4,\ldots,m. \tag{3.11.37b}$$

Compute,

$$t_1^* = \frac{t_1}{c_1} \quad ,$$

$$t_2^* = \frac{t_2-b_2\,t_1^*}{c_2^*} \; ,$$

$$t_i^* = \frac{t_i-a_i\,t_{i-2}^*-b_i^*\,t_{i-1}^*}{c_i^*} \; , \; i=3,4,\ldots,m. \tag{3.11.37c}$$

The components u_i of the solution vector $\underline{u}$ can then be computed by the back substitution,

$$u_m = t_m^*$$

$$u_{m-1} = t_{m-1}^*-d_{m-1}^*u_m \; ,$$

$$u_i = t_i^*-d_i^*u_{i+1}-e_i^*u_{i+2} \quad , \; i=m-2,m-3,\ldots,1 \tag{3.11.37d}$$

It can be shown that in passing from point to block iterative methods we can obtain improved rates of convergence. Concerning the S.L.O.R. method, Cuthill and Varga (1959) have shown that it is possible to normalize the equation in such a way that the S.O.R. and S.L.O.R. both require exactly the same arithmetic computations per mesh point. Also, it has been shown that this normalized method is stable with respect to the growth of rounding errors. Varga (1962), pp.204-205, shows for the model problem,

$$\rho(B^{(\text{line})}) = \frac{\cos\pi h}{2-\cos\pi h}\ , \tag{3.11.38}$$

From this, we can easily calculate that,

$$\left.\begin{aligned} &R(B^{(\text{line})}) \approx \pi^2 h^2\ , \\ \text{and} \qquad &R(\mathcal{L}^{(\text{line})}) \approx 2\pi^2 h^2\ , \end{aligned}\right\} h\to 0\ . \tag{3.11.39}$$

Hence, by comparing these results with (3.9.13), we note that the line Jacobi method converges at the same rate as the point G.S. method.

Consequently, for the S.L.O.R. and S.2.L.O.R. methods we have,

$$R(\mathcal{L}_{\omega_b}^{(\text{line})}) \approx 2\sqrt{2}\pi h,\quad h\to 0\ , \tag{3.11.40}$$

$$R(\mathcal{L}_{\omega_b}^{(2\ \text{line})}) \approx 4\pi h\ ,\ h\to 0\ , \tag{3.11.41}$$

Hence,

$$\frac{R(\mathcal{L}_{\omega_b}^{(\text{line})})}{R(\mathcal{L}_{\omega_b})} \approx \frac{2\sqrt{2}\pi h}{2\pi h} \approx \sqrt{2}\ , \tag{3.11.42a}$$

and,

$$\frac{R(\mathcal{L}_{\omega_b}^{(2\ \text{line})})}{R(\mathcal{L}_{\omega_b}^{(\text{line})})} \approx \frac{4\pi h}{2\sqrt{2}\pi h} \approx \sqrt{2} \tag{3.11.42b}$$

which shows that the rate of convergence of the S.2.L.O.R. method is approximately twice that of the S.O.R. method. Parter (1961) showed that the k-line S.O.R. method with optimum ω converges approximately $\sqrt{2k}$ as fast as S.O.R. method. But, one must solve more involved systems of equations at each iteration. Parter (1981) gave a formula for evaluating the spectral radius ρ of the k-line Jacobi scheme and is given by,

$$\rho(B^{(k\ \text{line})}) \approx 1-k\pi^2h^2 \ . \tag{3.11.43}$$

A "new" ordering has been investigated by Benson (1969), Evans (1972) and Benson and Evans (1972). For a two-dimensional problem with Dirichlet, Neuman or mixed conditions on a closed rectangular boundary they used the usual five-point finite difference equations and ordered the points in a "peripheral" fashion shown in Figure (3.11.2).

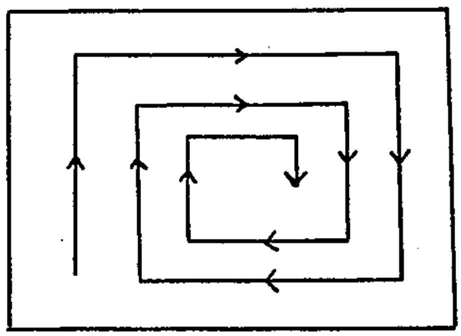


FIGURE (3.11.2)

The resulting matrix is consistently-ordered in block form, so that all the benefits of the S.O.R. method can be obtained.

3.12 ALTERNATING DIRECTION IMPLICIT METHODS

Alternating-Direction Implicit (A.D.I.) methods are somewhat similar to the line Jacobi iterative methods but with alternating directions. There are two basic A.D.I. methods due to Peaceman and Rachford (1955) and Douglas and Rachford (1956). In this section we will consider the former, which is known as the Peaceman-Rachford (PR) method for solving the system

$$A\underline{u} = \underline{b} \ , \tag{3.12.1}$$

where the $(m\times m)$ matrix A is nonsingular and can be represented as the sum of three $(m\times m)$ matrices,

$$A = H + V + \Sigma \ . \tag{3.12.2}$$

We can make the following assertions about the matrix A, which can be verified using the theorems of Sections (2.2), (2.4) and (2.5).

(a) A is an irreducible Stieltjes matrix, i.e. A is a real, symmetric, and positive definite irreducible matrix with non-positive off-diagonal entries.

(b) H and V are real, symmetric, diagonally dominant matrices with positive diagonal entries and non-positive off-diagonal entries.

(c) Σ is a non-negative diagonal matrix.

In a typical situation H and V would be tridiagonal or could be made so by a permutation of the rows and corresponding columns.

By using (3.12.2) we can write the matrix equation (3.12.1) as a pair of matrix equations,

$$\left.\begin{aligned} (H+\tfrac{1}{2}\Sigma+rI)\underline{u} &= \underline{b}-(V+\tfrac{1}{2}\Sigma-rI)\underline{u} \ , \\ (V+\tfrac{1}{2}\Sigma+rI)\underline{u} &= \underline{b}-(H+\tfrac{1}{2}\Sigma-rI)\underline{u} \ , \end{aligned}\right\} \tag{3.12.3}$$

for any positive scalar r. If we let

$$H_1 = H+\tfrac{1}{2}\Sigma \ , \quad V_1 = V+\tfrac{1}{2}\Sigma \ , \tag{3.12.4}$$

then the *Peaceman-Rachford alternating-direction implicit method* is defined by

$$\left.\begin{aligned}(H_1+r_{k+1}I)\underline{u}^{(k+\frac{1}{2})} &= \underline{b} - (V_1-r_{k+1}I)\underline{u}^{(k)} \ , \\ (V_1+r_{k+1}I)\underline{u}^{(k+1)} &= \underline{b} - (H_1-r_{k+1}I)\underline{u}^{(k+\frac{1}{2})} ,\end{aligned}\right\} \qquad (3.12.5)$$

where the r_k's are positive *acceleration parameters* chosen to make the process converge rapidly, and $\underline{u}^{(0)}$ is an arbitrary initial vector approximation to the unique solution of (3.12.1).

Since both H_1 and V_1 are, by suitable rearrangement of their rows and corresponding columns, tridiagonal matrices, the iterative method (3.12.5) can be carried out directly using the algorithm given in Section (3.11) on page 126. The vector $\underline{u}^{(k+\frac{1}{2})}$ is treated as an auxiliary vector which is discarded as soon as it has been used in the calculation of $\underline{u}^{(k+1)}$.

The two equations of (3.12.5) are now combined to give,

$$\underline{u}^{(k+1)} = T_{r_{k+1}}\underline{u}^{(k)} + g_{r_{k+1}}(\underline{b}) \ , \quad k\geq 0 \qquad (3.12.6)$$

where,

$$T_r = (V_1+rI)^{-1}(H_1-rI)(H_1+rI)^{-1}(V_1-rI) \ , \qquad (3.12.7a)$$

and

$$g_r(\underline{b}) = (V_1+rI)^{-1}\{I-(H_1-rI)(H_1+rI)^{-1}\}\underline{b} \ . \qquad (3.12.7b)$$

It will be noticed that (3.12.6) is of the same form as (3.5.3).

We consider the case, where all the constants r_j are equal to the fixed constant $r>0$. The following convergence theorem will be stated here without proof (see Varga (1962), p.213).

Theorem (3.12.1)

Let H_1 and V_1 be $m\times m$ symmetric non-negative definite matrices, where at least one of the matrices H_1 and V_1 is positive definite. Then, for any $r>0$, the PR matrix T_r of (3.12.7a) is convergent.

For the model problem (Section 3.2), if the vector $\underline{\alpha}^{(p,q)}$ is defined so that its component for the i^{th} column and j^{th} row of the mesh is (for $h=1/(m+1)$)

$$\alpha_{i,j}^{(p,q)} = \beta_{p,q} \sin\left(\frac{p\pi i}{m+1}\right) . \sin\left(\frac{q\pi j}{m+1}\right) , \quad 1\leq i,j\leq m, \; 1\leq p,q\leq m \tag{3.12.8}$$

it follows that,

$$\begin{aligned} H\underline{\alpha}^{(p,q)} &= 4 \sin^2\left(\frac{p\pi}{2(m+1)}\right)\underline{\alpha}^{(p,q)} , \\ V\underline{\alpha}^{(p,q)} &= 4 \sin^2\left(\frac{q\pi}{2(m+1)}\right)\underline{\alpha}^{(p,q)} . \end{aligned} \quad \text{for all } 1\leq p,q\leq m \tag{3.12.9}$$

Since for this problem $\Sigma=0$, hence from (3.12.9) and (3.12.7a) we have,

$$T_r\underline{\alpha}^{(p,q)} = \left[\frac{r-4\sin^2\left(\frac{q\pi}{2(m+1)}\right)}{r+4\sin^2\left(\frac{q\pi}{2(m+1)}\right)}\right] \left[\frac{r-4\sin^2\left(\frac{p\pi}{2(m+1)}\right)}{r+4\sin^2\left(\frac{p\pi}{2(m+1)}\right)}\right] \underline{\alpha}^{(p,q)} , \quad 1\leq p,q\leq m. \tag{3.12.10}$$

We therefore conclude that,

$$\rho(T_r) = \left\{\max_{1\leq q\leq m} \left|\frac{r-4\sin^2\left(\frac{q\pi}{2(m+1)}\right)}{r+4\sin^2\left(\frac{q\pi}{2(m+1)}\right)}\right|\right\}^2 . \tag{3.12.11}$$

To minimize this as a function of r, we consider the simple function,

$$g(x;r) = \frac{r-x}{r+x} , \quad r>0 \tag{3.12.12}$$

where $0<x_1\leq x\leq x_2$. The derivative of $g(x;r)$ with respect to x is negative for all $x\geq 0$, so that,

$$\max_{x_1\leq x\leq x_2} |g(x;r)| = \max\left\{\left|\frac{r-x_1}{r+x_1}\right| , \left|\frac{r-x_2}{r+x_2}\right|\right\} . \tag{3.12.13}$$

From this, it is easily shown that,

$$\max_{x_1\leq x\leq x_2} |g(x;r)| = \begin{cases} \dfrac{x_2-r}{x_2+r} , & 0<r\leq\sqrt{x_1x_2} , \\[2ex] \dfrac{r-x_1}{r+x_1} , & r\geq\sqrt{x_1x_2} , \end{cases} \tag{3.12.14}$$

hence,

$$\min_{r>0} \{ \max_{x_1\leq x\leq x_2} |g(x;r)|\} = g(x_1;\sqrt{x_1x_2}) = \frac{\sqrt{x_2}-\sqrt{x_1}}{\sqrt{x_2}+\sqrt{x_1}} . \tag{3.12.15}$$

Thus, by setting $x_1=4\sin^2(\frac{\pi}{2(m+1)})$ and $x_2=4\cos^2(\frac{\pi}{2(m+1)})$, we have $r=\sqrt{x_1 x_2}$ is optimum in the sense that the spectral radius $\rho(T_r)$ is minimized. It can further be shown that the spectral radius (and, consequently, the rate of convergence) of this method is asymptotically the same as that of the point S.O.R. with optimum relaxation parameter for all $h>0$ for the model problem.

A substantial improvement in the convergence of the Peaceman-Rachford method for solving Laplace's equation in a square can be obtained by the use of a sequence of iteration parameters r_{k+1} (see equation (3.12.5)). In this case the parameters are used successively in a cyclic order.

At least two methods have been given for choosing the iteration parameters, which though not optimal, nevertheless are nearly as good as the optimal parameters in many cases. The first set of parameters, given by Peaceman and Rachford (1955) are,

$$r_i^{(p)} = \bar{b}(\bar{a}/\bar{b})^{(2i-1)/2n} \quad , \quad i=1,2,\ldots,n \tag{3.12.16}$$

and the second set are the Wachspress parameters (Wachspress, 1957), which are given by,

$$r_i^{(w)} = \bar{b}(\bar{a}/\bar{b})^{(i-1)/(n-1)}, \quad n\geq 2, \quad i=1,2,\ldots,n \tag{3.12.17}$$

where, $\bar{a} = \min(a,\alpha)$, $\bar{b} = \max(b,\beta)$,

a and b, α and β are the least and greatest eigenvalues of H_1 and V_1 of equations (3.12.5) respectively.

CHAPTER 4

ALTERNATIVE GROUP ITERATIVE METHODS

4.1 INTRODUCTION

In Section (3.11) block iterative methods were discussed and in particular the one line and two-line iterative methods. This chapter presents the novel approach of using small groups of fixed size, i.e., groups of a certain number of individual equations (mesh points) and this group is treated similar to the way a single point is treated in the point iterative method.

Our main concern here is to construct new grouping of the mesh points into small size groups of 2,4,6,9,12,16 and 25 points and to investigate their advantages as well as choosing the most efficient group for solving elliptic type of p.d.e.'s in two-space dimensions.

In Section (4.5) and by a similar manner to the two dimensional case, we present an explicit 8-point group to solve the elliptic p.d.e. in three-space dimensions. The method is developed and analysed theoretically and experimentally and compared with results from the point S.O.R. iterative method.

4.2 THE 2-SPACE DIMENSIONAL GROUP ITERATIVE METHODS

4.2.1 The 2-Point Group

Consider the linear system (3.11.1) defined in the unit square, $0 \leq x, y \leq 1$, with m^2 internal mesh points in the region shown in Figure (4.2.1).

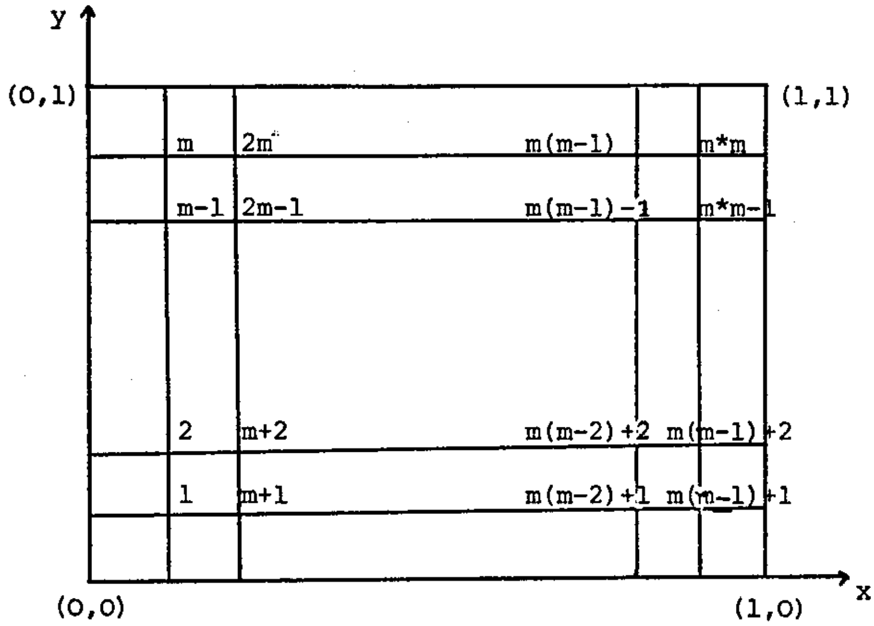


FIGURE (4.2.1)

Suppose that the mesh points are ordered in *groups of two* as shown in Figure (4.2.2).

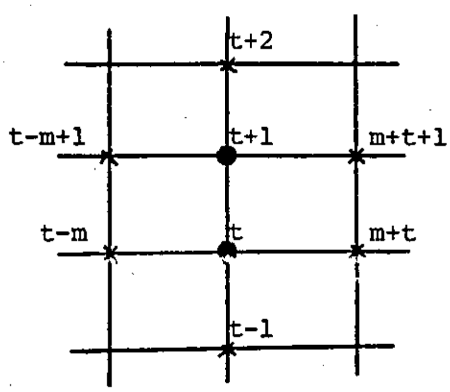


FIGURE (4.2.2)

where $t=(rm+1)$, step 2, $(r+1)m-1$ and $r=0,1,2,\ldots,m-1$. We take m to be an even number, such that each subset (group) G_ℓ of Definition (3.11.1), $\ell=1,2,\ldots,m^2/2$, consists of two elements $\{t,t+1\}$.

For the simple case of the unit square and $\Delta x=\Delta y=h=1/5$, assume that the groups are ordered in red-black ordering as shown in Figure (4.2.3).

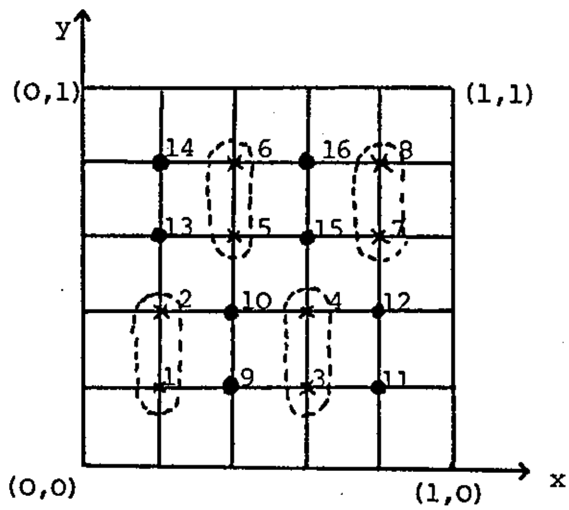


FIGURE (4.2.3)

By using the five point approximation scheme, the finite difference equation at the point P (see Figure (4.2.4)) has the form,

$$u_P+\alpha_1 u_{B,P}+\alpha_2 u_{R,P}+\alpha_3 u_{T,P}+\alpha_4 u_{L,P} = b_P \,, \tag{4.2.1}$$

where B,R,T and L denote the Bottom, Right, Top and Left of the representative point P respectively.

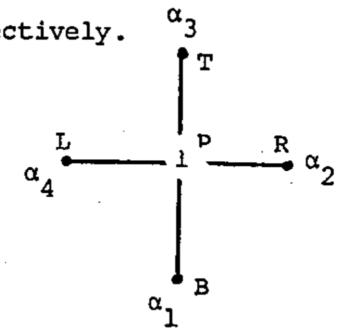


FIGURE (4.2.4)

This can be applied to each of the mesh points of Figure (4.2.3) in turn. The resulting coefficient matrix then has the block structure,

$$A = \left[\begin{array}{cccc|cccc}
R_0 & & & & R_2 & 0 & R_3 & 0 \\
 & R_0 & \bigcirc & & R_4 & R_2 & 0 & R_3 \\
 & & R_0 & & R_1 & 0 & R_4 & R_2 \\
 & \bigcirc & & R_0 & 0 & R_1 & 0 & R_4 \\
\hline
R_4 & R_2 & R_3 & 0 & R_0 & & & \\
0 & R_4 & 0 & R_3 & & R_0 & \bigcirc & \\
R_1 & 0 & R_2 & 0 & \bigcirc & & R_0 & \\
0 & R_1 & R_4 & R_2 & & & & R_0
\end{array}\right], \qquad (4.2.2a)$$

where,

$$R_0 = \begin{pmatrix} 1 & \alpha_3 \\ \alpha_1 & 1 \end{pmatrix}, \quad R_1 = \begin{pmatrix} 0 & \alpha_1 \\ 0 & 0 \end{pmatrix}, \quad R_2 = \begin{pmatrix} \alpha_2 & 0 \\ 0 & \alpha_2 \end{pmatrix},$$

$$R_3 = \begin{pmatrix} 0 & 0 \\ \alpha_3 & 0 \end{pmatrix} \text{ and } R_4 = \begin{pmatrix} \alpha_4 & 0 \\ 0 & \alpha_4 \end{pmatrix} \qquad (4.2.2b)$$

This matrix has Property $A^{(\pi)}$ and is π-consistently ordered, see Definitions (3.11.3) and (3.11.4). If the groups are taken in natural row ordering, then the coefficient matrix A has the structure,

$$A = \left[\begin{array}{cccc|cccc}
R_0 & R_2 & 0 & 0 & R_3 & & & \\
R_4 & R_0 & R_2 & 0 & & R_3 & \bigcirc & \\
0 & R_4 & R_0 & R_2 & \bigcirc & & R_3 & \\
0 & 0 & R_4 & R_0 & & & & R_3 \\
\hline
R_1 & & & & R_0 & R_2 & 0 & 0 \\
 & R_1 & \bigcirc & & R_4 & R_0 & R_2 & 0 \\
\bigcirc & & R_1 & & 0 & R_4 & R_0 & R_2 \\
 & & & R_1 & 0 & 0 & R_4 & R_0
\end{array}\right], \qquad (4.2.3)$$

which is also π-consistently ordered and has Property $A^{(\pi)}$. Hence, in both cases the theory of block S.O.R. is valid.

Now, to derive the explicit block S.O.R. method for this problem, we calculate the transformed matrix A^E and the modified vector $\underline{b}^E$, where,

$$A^E = [\text{diag}\{R_0\}]^{-1}A\ , \tag{4.2.4}$$

and,

$$\underline{b}^E = [\text{diag}\{R_0\}]^{-1}\underline{b}\ . \tag{4.2.5}$$

The matrix $[\text{diag}\{R_0\}]^{-1}$ is simply $\text{diag}\{R_0^{-1}\}$ and the inverse of R_0 is given by,

$$R_0^{-1} = \frac{1}{1-\alpha_1\alpha_3}\begin{bmatrix} 1 & -\alpha_3 \\ -\alpha_1 & 1 \end{bmatrix}\ . \tag{4.2.6}$$

Hence,

$$A^E = \begin{bmatrix} I & C \\ B & I \end{bmatrix}\ , \tag{4.2.7a}$$

where,

$$B=\frac{1}{1-\alpha_1\alpha_3}\left[\begin{array}{cc|cc|cc|cc}
\alpha_4 & -\alpha_3\alpha_4 & \alpha_2 & -\alpha_2\alpha_3 & -\alpha_3^2 & 0 & 0 & 0 \\
-\alpha_1\alpha_4 & \alpha_4 & -\alpha_1\alpha_2 & \alpha_2 & \alpha_3 & 0 & 0 & 0 \\
\hline
0 & 0 & \alpha_4 & -\alpha_3\alpha_4 & 0 & 0 & -\alpha_3^2 & 0 \\
0 & 0 & -\alpha_1\alpha_4 & \alpha_4 & 0 & 0 & \alpha_3 & 0 \\
\hline
0 & \alpha_1 & 0 & 0 & \alpha_2 & -\alpha_2\alpha_3 & 0 & 0 \\
0 & -\alpha_1^2 & 0 & 0 & -\alpha_1\alpha_2 & \alpha_2 & 0 & 0 \\
\hline
0 & 0 & 0 & \alpha_1 & \alpha_4 & -\alpha_3\alpha_4 & \alpha_2 & -\alpha_2\alpha_3 \\
0 & 0 & 0 & -\alpha_1^2 & -\alpha_1\alpha_4 & \alpha_4 & -\alpha_1\alpha_2 & \alpha_2
\end{array}\right]\ , \tag{4.2.7b}$$

and,

$$C=\frac{1}{1-\alpha_1\alpha_3}\left[\begin{array}{cc|cc|cc|cc}
\alpha_2 & -\alpha_2\alpha_3 & 0 & 0 & -\alpha_3^2 & 0 & 0 & 0 \\
-\alpha_1\alpha_2 & \alpha_2 & 0 & 0 & \alpha_3 & 0 & 0 & 0 \\
\hline
\alpha_4 & -\alpha_3\alpha_4 & \alpha_2 & -\alpha_2\alpha_3 & 0 & 0 & -\alpha_3^2 & 0 \\
-\alpha_1\alpha_4 & \alpha_4 & -\alpha_1\alpha_2 & \alpha_2 & 0 & 0 & \alpha_3 & 0 \\
\hline
0 & \alpha_1 & 0 & 0 & \alpha_4 & -\alpha_3\alpha_4 & \alpha_2 & -\alpha_2\alpha_3 \\
0 & -\alpha_1^2 & 0 & 0 & -\alpha_1\alpha_4 & \alpha_4 & -\alpha_1\alpha_2 & \alpha_2 \\
\hline
0 & 0 & 0 & \alpha_1 & 0 & 0 & \alpha_4 & -\alpha_3\alpha_4 \\
0 & 0 & 0 & -\alpha_1^2 & 0 & 0 & -\alpha_1\alpha_4 & \alpha_4
\end{array}\right] . \tag{4.2.7c}$$

The block structure of the matrix A^E is the same as that of A with the submatrices R_0 replaced by identity matrices and the submatrices R_i, i=1,2,3,4 replaced by $R_0^{-1}R_i$. In addition, where R_i, i=1,2,3,4 has a column of zeros, so does $R_0^{-1}R_i$, and where an element α_i occurs as the $(p,q)^{th}$ element of R_i, the q^{th} column of $R_0^{-1}R_i$ is the p^{th} column of R_0^{-1}, multiplied by α_i. For example,

$$R_0^{-1}R_1 = \frac{1}{1-\alpha_1\alpha_3}\begin{pmatrix} 0 & \alpha_1 \\ 0 & -\alpha_1^2 \end{pmatrix} . \tag{4.2.8}$$

For the model problem (3.2.1) and a square grid,

$$\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = -\frac{1}{4} , \tag{4.2.9}$$

so that from (4.2.6),

$$R_0^{-1} = \frac{4}{15}\begin{pmatrix} 4 & 1 \\ 1 & 4 \end{pmatrix} , \tag{4.2.10}$$

and for example,

$$R_0^{-1}R_1 = -\frac{1}{15}\begin{pmatrix} 0 & 4 \\ 0 & 1 \end{pmatrix} . \tag{4.2.11}$$

Similarly, we can obtain $R_0^{-1}R_2$, $R_0^{-1}R_3$ and $R_0^{-1}R_4$, from which we can establish the computational molecule at the point P to be,

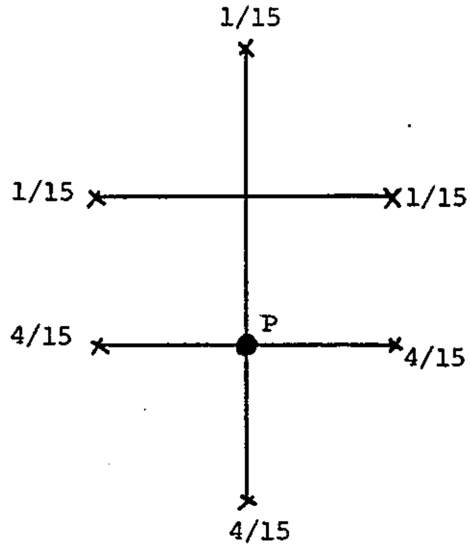


FIGURE (4.2.5)

Hence we have the explicit group Jacobi method given by,

$$\left.\begin{aligned} u_t^{(k+1)} &= \frac{1}{15}[4(u_{t-m}^{(k)}+u_{t-1}^{(k)}+u_{m+t}^{(k)})+u_{m+t+1}^{(k)}+u_{t+2}^{(k)}+u_{t-m+1}^{(k)}] \ , \\ u_{t+1}^{(k+1)} &= \frac{1}{15}[4(u_{m+t+1}^{(k)}+u_{t+2}^{(k)}+u_{t-m+1}^{(k)})+u_{t-m}^{(k)}+u_{t-1}^{(k)}+u_{m+t}^{(k)}] \ , \end{aligned}\right\} \qquad (4.2.12)$$

where $t=(rm+1),(2),(r+1)m-1$ and $r=0,1,2,\ldots,m-1$. Note that, if any of $u_\ell^{(k)}$ in the right hand sides of (4.2.12) is a point of the boundary ∂R then we compute the appropriate value of u at this point by (3.2.2). Clearly similar sets of equations can be written down for the 2-point explicit Gauss-Seidel, S.O.R. and J.O.R. iterative methods.

4.2.2 The 4-Point Group

Similar to the 2-point group an explicit set of equations for the *4-point group* can be derived (Evans and Biggins, (1982)) in which each group is formed from 4 points of the net region Figure (4.2.1) in accordance with Figure (4.2.6),

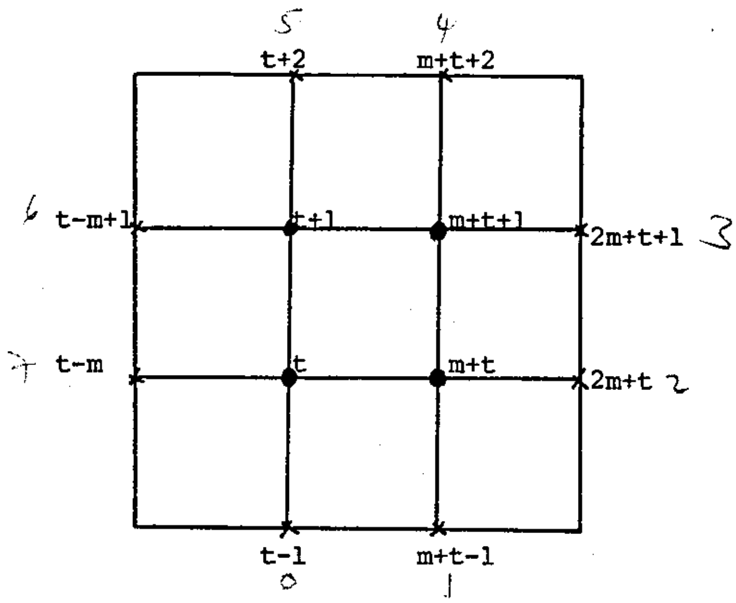


FIGURE (4.2.6)

where t=(rm+1),(2),(r+1)m-1 and r=0,(2),m-2. Like the 2-point group, m must be an even number, and G_ℓ , $\ell=1,2,\ldots,\frac{m^2}{4}$ consists of 4 elements {t,t+1,m+t,m+t+1}. If we take the unit square and $\Delta x=\Delta y=1/7$, and with a red-black ordering of the 4 point groups, then by applying equation (4.2.1) to each of the mesh points in turn, the resulting coefficient matrix has the block structure,

$$A = \left[\begin{array}{ccccc:cccc}
R_0 & & & & & R_2 & R_3 & 0 & 0 \\
& R_0 & & O & & R_4 & 0 & R_3 & 0 \\
& & R_0 & & & R_1 & R_4 & R_2 & R_3 \\
& O & & R_0 & & 0 & R_1 & 0 & R_2 \\
& & & & R_0 & 0 & 0 & R_1 & R_4 \\
\hdashline
R_4 & R_2 & R_3 & 0 & 0 & R_0 & & & \\
R_1 & 0 & R_2 & R_3 & 0 & & R_0 & O & \\
0 & R_1 & R_4 & 0 & R_3 & & O & R_0 & \\
0 & 0 & R_1 & R_4 & R_2 & & & & R_0
\end{array}\right] , \qquad (4.2.13a)$$

where,

$$R_0 = \left[\begin{array}{cc|cc} 1 & \alpha_3 & \alpha_2 & 0 \\ \alpha_1 & 1 & 0 & \alpha_2 \\ \hline \alpha_4 & 0 & 1 & \alpha_3 \\ 0 & \alpha_4 & \alpha_1 & 1 \end{array}\right], \quad R_1 = \left[\begin{array}{c|c} \begin{matrix} 0 & \alpha_1 \\ 0 & 0 \end{matrix} & O \\ \hline O & \begin{matrix} 0 & \alpha_1 \\ 0 & 0 \end{matrix} \end{array}\right], \quad R_2 = \left[\begin{array}{c|c} O & O \\ \hline \begin{matrix} \alpha_2 & 0 \\ 0 & \alpha_2 \end{matrix} & O \end{array}\right]$$

$$R_3 = \left[\begin{array}{c|c} \begin{matrix} 0 & 0 \\ \alpha_3 & 0 \end{matrix} & O \\ \hline O & \begin{matrix} 0 & 0 \\ \alpha_3 & 0 \end{matrix} \end{array}\right] \quad \text{and} \quad R_4 = \left[\begin{array}{c|c} O & \begin{matrix} \alpha_4 & 0 \\ 0 & \alpha_4 \end{matrix} \\ \hline O & O \end{array}\right] \tag{4.2.13b}$$

If the groups are taken in natural row ordering, the coefficient matrix A will have the same structure as given in (4.2.3), and the theory of block S.O.R. is valid, since the matrix A is π-consistently ordered and has Property $A^{(\pi)}$.

To derive the explicit block S.O.R. method we follow the same steps as the 2 point group, hence we have to determine the inverse of R_0 which is given by,

$$R_0^{-1} = \frac{1}{d}\begin{bmatrix} \alpha_5 & \alpha_3\alpha_6 & \alpha_2\alpha_7 & 2\alpha_2\alpha_3 \\ \alpha_1\alpha_6 & \alpha_5 & 2\alpha_1\alpha_2 & \alpha_2\alpha_7 \\ \alpha_4\alpha_7 & 2\alpha_3\alpha_4 & \alpha_5 & \alpha_3\alpha_6 \\ 2\alpha_1\alpha_4 & \alpha_4\alpha_7 & \alpha_1\alpha_6 & \alpha_5 \end{bmatrix}, \tag{4.2.14a}$$

where,

$$d = (\alpha_6+1)^2+2\alpha_5-1, \tag{4.2.14b}$$

$$\alpha_5 = 1-\alpha_1\alpha_3-\alpha_2\alpha_4, \quad \alpha_6 = \alpha_1\alpha_3-\alpha_2\alpha_4-1 \quad \text{and} \quad \alpha_7 = \alpha_2\alpha_4-\alpha_1\alpha_3-1. \tag{4.2.14c}$$

Hence,

$$R_0^{-1}R_1 = \frac{1}{d}\begin{bmatrix} 0 & \alpha_1\alpha_5 & 0 & \alpha_1\alpha_2\alpha_7 \\ 0 & \alpha_1^2\alpha_6 & 0 & 2\alpha_1^2\alpha_2 \\ 0 & \alpha_1\alpha_4\alpha_7 & 0 & \alpha_1\alpha_5 \\ 0 & 2\alpha_1^2\alpha_4 & 0 & \alpha_1^2\alpha_6 \end{bmatrix}. \tag{4.2.15}$$

For the model problem (3.2.1) and by using (4.2.9),

$$R_O^{-1} = \frac{1}{6}\begin{bmatrix} 7 & 2 & 2 & 1 \\ 2 & 7 & 1 & 2 \\ 2 & 1 & 7 & 2 \\ 1 & 2 & 2 & 7 \end{bmatrix} \quad . \qquad (4.2.16)$$

and hence,

$$R_O^{-1}R_1 = -\frac{1}{24}\begin{bmatrix} 0 & 7 & 0 & 2 \\ 0 & 2 & 0 & 1 \\ 0 & 2 & 0 & 7 \\ 0 & 1 & 0 & 2 \end{bmatrix} \quad . \qquad (4.2.17)$$

From which we can set up the computational molecule at the point P (say) to be,

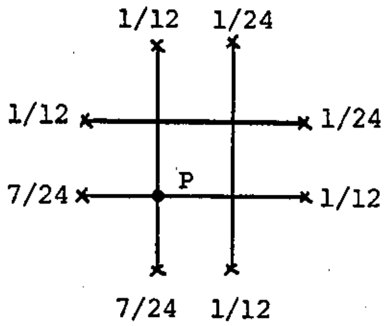


FIGURE (4.2.7)

The explicit group Jacobi method can be derived by using this molecule as,

$$\left.\begin{aligned} u_t^{(k+1)} &= \frac{1}{24}[7(u_{t-m}^{(k)}+u_{t-1}^{(k)})+2(u_{m+t-1}^{(k)}+u_{2m+t}^{(k)}+u_{t+2}^{(k)}+u_{t-m+1}^{(k)})+ \\ &\qquad u_{2m+t+1}^{(k)}+u_{m+t+2}^{(k)}] \;, \\ u_{t+1}^{(k+1)} &= \frac{1}{24}[7(u_{t+2}^{(k)}+u_{t-m+1}^{(k)})+2(u_{t-m}^{(k)}+u_{t-1}^{(k)}+u_{2m+t+1}^{(k)}+u_{m+t+2}^{(k)})+ \\ &\qquad u_{m+t-1}^{(k)}+u_{2m+t}^{(k)}] \;, \end{aligned}\right|$$

$$\left.\begin{aligned}
u_{m+t}^{(k+1)} &= \frac{1}{24}[7(u_{m+t-1}^{(k)}+u_{2m+t}^{(k)})+2(u_{t-m}^{(k)}+u_{t-1}^{(k)}+u_{2m+t+1}^{(k)}+u_{m+t+2}^{(k)})+ \\
&\qquad u_{t+2}^{(k)}+u_{t-m+1}^{(k)}] , \\
u_{m+t+1}^{(k+1)} &= \frac{1}{24}[7(u_{2m+t+1}^{(k)}+u_{m+t+2}^{(k)})+2(u_{m+t-1}^{(k)}+u_{2m+t}^{(k)}+u_{t+2}^{(k)}+u_{t-m+1}^{(k)})+ \\
&\qquad u_{t-m}^{(k)}+u_{t-1}^{(k)}] ,
\end{aligned}\right\} \tag{4.2.18}$$

where t=(rm+1),(2),(r+1)m-1 and r=0,(2),m-2.

The same discussion for the 2 point group can be applied to the 4 point group.

4.2.3 The 6-Point Group

The next grouping of mesh points which will be considered is the *6-point group*, where each group is formed from 6 points of the net region Figure (4.2.1) in accordance with Figure (4.2.8).

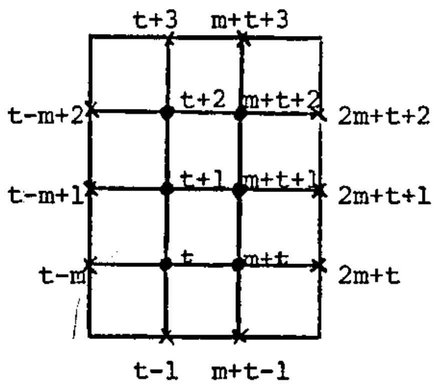


FIGURE (4.2.8)

where t=(rm+1),(3),(r+1)m-2 and r=0,(2),m-2, each subset G_ℓ, ℓ=1,2,..., $\frac{m^2}{6}$, of Definition (3.11.1) consists of 6 elements. For a square region, m must be divisible by 6.

Consider the linear system (3.11.1) defined in a unit square with

$\Delta x=\Delta y=1/7$ and assume that the groups are ordered in red-black ordering, then by applying equation (4.2.1) to each of the mesh points in turn, the resulting coefficient matrix A has the block structure,

$$A = \left[\begin{array}{ccc|ccc} R_0 & & \bigcirc & R_2 & R_3 & 0 \\ & R_0 & & R_4 & 0 & R_3 \\ \bigcirc & & R_0 & R_1 & R_4 & R_2 \\ \hline R_4 & R_2 & R_3 & R_0 & & \bigcirc \\ R_1 & 0 & R_2 & & R_0 & \\ 0 & R_1 & R_4 & \bigcirc & & R_0 \end{array}\right], \tag{4.2.19a}$$

where,

$$R_0 = \left[\begin{array}{ccc|ccc} 1 & \alpha_3 & 0 & \alpha_2 & & \bigcirc \\ \alpha_1 & 1 & \alpha_3 & & \alpha_2 & \\ 0 & \alpha_1 & 1 & \bigcirc & & \alpha_2 \\ \hline \alpha_4 & & \bigcirc & 1 & \alpha_3 & 0 \\ & \alpha_4 & & \alpha_1 & 1 & \alpha_3 \\ \bigcirc & & \alpha_4 & 0 & \alpha_1 & 1 \end{array}\right], \quad R_1 = \left[\begin{array}{ccc|ccc} 0 & 0 & \alpha_1 & & & \\ 0 & 0 & 0 & & \bigcirc & \\ 0 & 0 & 0 & & & \\ \hline & & & 0 & 0 & \alpha_1 \\ & \bigcirc & & 0 & 0 & 0 \\ & & & 0 & 0 & 0 \end{array}\right],$$

$$R_2 = \left[\begin{array}{ccc|ccc} & & & & & \\ & \bigcirc & & & \bigcirc & \\ & & & & & \\ \hline \alpha_2 & 0 & 0 & & & \\ 0 & \alpha_2 & 0 & & \bigcirc & \\ 0 & 0 & \alpha_2 & & & \end{array}\right], \quad R_3 = \left[\begin{array}{ccc|ccc} 0 & 0 & 0 & & & \\ 0 & 0 & 0 & & \bigcirc & \\ \alpha_3 & 0 & 0 & & & \\ \hline & & & 0 & 0 & 0 \\ & \bigcirc & & 0 & 0 & 0 \\ & & & \alpha_3 & 0 & 0 \end{array}\right],$$

and,

$$R_4 = \left[\begin{array}{ccc|ccc} & & & \alpha_4 & 0 & 0 \\ & \bigcirc & & 0 & \alpha_4 & 0 \\ & & & 0 & 0 & \alpha_4 \\ \hline & \bigcirc & & & \bigcirc & \end{array}\right] \tag{4.2.19b}$$

Similar to the previous two cases, the theory of block S.O.R. can be applied to this case since A is π-consistently ordered and has Property $A^{(\pi)}$.

In order to form the transformed matrix A^E of equation (4.2.4), the matrix R_O again, has to be inverted and its inverse is given by,

$$R_O^{-1} = \frac{1}{d}\begin{pmatrix} \beta_1^2-\alpha_1\alpha_3(\beta_2+3) & -\alpha_3\beta_1(\beta_2+1) & \alpha_3^2\beta_3 & \alpha_2(\alpha_1\alpha_3\beta_3-\beta_1^2) & 2\alpha_2\alpha_3\beta_1 & -\alpha_2\alpha_3^2(\beta_2+3) \\ -\alpha_1\beta_1(\beta_2+1) & \beta_1(\beta_1-2\alpha_1\alpha_3) & -\alpha_3\beta_1(\beta_2+1) & 2\alpha_1\alpha_2\beta_1 & \alpha_2\beta_1(\beta_2-1) & 2\alpha_2\alpha_3\beta_1 \\ \alpha_1^2\beta_3 & -\alpha_1\beta_1(\beta_2+1) & \beta_1^2-\alpha_1\alpha_3(\beta_2+3) & -\alpha_1^2\alpha_2(\beta_2+3) & 2\alpha_1\alpha_2\beta_1 & \alpha_2(\alpha_1\alpha_3\beta_3-\beta_1^2) \\ \alpha_4(\alpha_1\alpha_3\beta_3-\beta_1^2) & 2\alpha_3\alpha_4\beta_1 & -\alpha_3^2\alpha_4(\beta_2+3) & \beta_1^2-\alpha_1\alpha_3(\beta_2+3) & -\alpha_3\beta_1(\beta_2+1) & \alpha_3^2\beta_3 \\ 2\alpha_1\alpha_4\beta_1 & \alpha_4\beta_1(\beta_2-1) & 2\alpha_3\alpha_4\beta_1 & -\alpha_1\beta_1(\beta_2+1) & \beta_1(\beta_1-2\alpha_1\alpha_3) & -\alpha_3\beta_1(\beta_2+1) \\ -\alpha_1^2\alpha_4(\beta_2+3) & 2\alpha_1\alpha_4\beta_1 & \alpha_4(\alpha_1\alpha_3\beta_3-\beta_1^2) & \alpha_1^2\beta_3 & -\alpha_1\beta_1(\beta_2+1) & \beta_1^2-\alpha_1\alpha_3(\beta_2+3) \end{pmatrix}$$

(4.2.20a)

where,

$$d = \beta_1[\beta_1^2-4\alpha_1\alpha_3(1+\alpha_2\alpha_4-\alpha_1\alpha_3)] \ , \tag{4.2.20b}$$

and,

$$\left.\begin{aligned} \beta_1 &= 1-\alpha_2\alpha_4 \ , \\ \beta_2 &= \alpha_2\alpha_4-2\alpha_1\alpha_3 \ , \\ \text{and} \quad \beta_3 &= 1-2\alpha_1\alpha_3+3\alpha_2\alpha_4 \ . \end{aligned}\right\} \tag{4.2.20c}$$

By applying (4.2.4) we can form the matrix A^E which has the same block structure as that of A with the submatrices R_O replaced by identity matrices and R_i, i=1,2,3,4 replaced by $R_O^{-1}R_i$, i.e.,

$$R_O^{-1}R_1=\frac{1}{d}\begin{pmatrix} 0 & 0 & \alpha_1\beta_1^2-\alpha_1^2\alpha_3(\beta_2+3) & 0 & 0 & \alpha_1\alpha_2(\alpha_1\alpha_3\beta_3-\beta_1^2) \\ 0 & 0 & -\alpha_1^2\beta_1(\beta_2+1) & 0 & 0 & 2\alpha_1^2\alpha_2\beta_1 \\ 0 & 0 & \alpha_1^3\beta_3 & 0 & 0 & -\alpha_1^3\alpha_2(\beta_2+3) \\ 0 & 0 & \alpha_1\alpha_4(\alpha_1\alpha_3\beta_3-\beta_1^2) & 0 & 0 & \alpha_1\beta_1^2-\alpha_1^2\alpha_3(\beta_2+3) \\ 0 & 0 & 2\alpha_1^2\alpha_4\beta_1 & 0 & 0 & -\alpha_1^2\beta_1(\beta_2+1) \\ 0 & 0 & -\alpha_1^3\alpha_4(\beta_2+3) & 0 & 0 & \alpha_1^3\beta_3 \end{pmatrix} . \tag{4.2.21}$$

For the model problem (3.2.1) and a square mesh $\alpha_i = -\frac{1}{4}$, i=1,2,3,4, and we have,

$$R_O^{-1} = \frac{4}{2415}\begin{pmatrix} 712 & 225 & 68 & 208 & 120 & 47 \\ 225 & 780 & 225 & 120 & 255 & 120 \\ 68 & 225 & 712 & 47 & 120 & 208 \\ 208 & 120 & 47 & 712 & 225 & 68 \\ 120 & 255 & 120 & 225 & 780 & 225 \\ 47 & 120 & 208 & 68 & 225 & 712 \end{pmatrix} , \tag{4.2.22}$$

$$R_0^{-1}R_1 = -\frac{1}{2415}\begin{bmatrix} 0 & 0 & 712 & 0 & 0 & 208 \\ 0 & 0 & 225 & 0 & 0 & 120 \\ 0 & 0 & 68 & 0 & 0 & 47 \\ 0 & 0 & 208 & 0 & 0 & 712 \\ 0 & 0 & 120 & 0 & 0 & 225 \\ 0 & 0 & 47 & 0 & 0 & 68 \end{bmatrix}, \qquad (4.2.23)$$

and

$$R_0^{-1}R_2 = -\frac{1}{2415}\begin{bmatrix} 208 & 120 & 47 & 0 & 0 & 0 \\ 120 & 255 & 120 & 0 & 0 & 0 \\ 47 & 120 & 208 & 0 & 0 & 0 \\ 712 & 225 & 68 & 0 & 0 & 0 \\ 225 & 780 & 225 & 0 & 0 & 0 \\ 68 & 225 & 712 & 0 & 0 & 0 \end{bmatrix}. \qquad (4.2.24)$$

Similarly, $R_0^{-1}R_3$ and $R_0^{-1}R_4$ can be obtained, which enable us to establish the computational molecule at the points P_1 and P_2 to be,

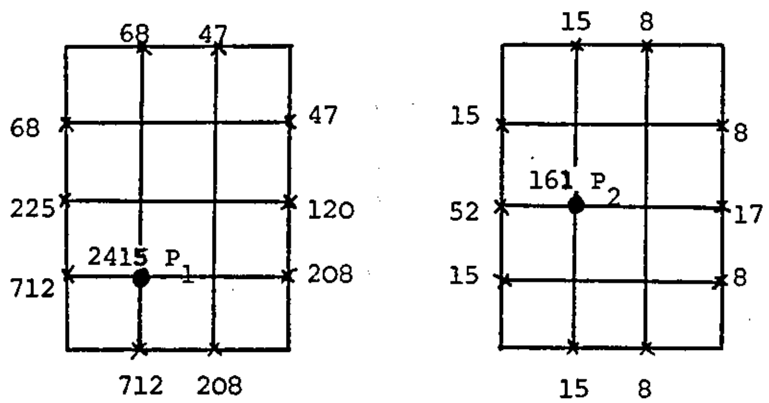


FIGURE (4.2.9)

Hence, an explicit group Jacobi method can be derived from this molecule and is given by,

$$u_t^{(k+1)} = \frac{1}{2415}\,[712(u_{t-m}^{(k)}+u_{t-1}^{(k)})+225u_{t-m+1}^{(k)}+208(u_{m+t-1}^{(k)}+u_{2m+t}^{(k)})+$$
$$120u_{2m+t+1}^{(k)}+68(u_{t+3}^{(k)}+u_{t-m+2}^{(k)})+47(u_{2m+t+2}^{(k)}+u_{m+t+3}^{(k)})] \; ,$$

$$u_{t+1}^{(k+1)} = \frac{1}{161}[52u_{t-m+1}^{(k)}+17u_{2m+t+1}^{(k)}+15(u_{t-m}^{(k)}+u_{t-1}^{(k)}+u_{t+3}^{(k)}+u_{t-m+2}^{(k)})+$$
$$8(u_{m+t-1}^{(k)}+u_{2m+t}^{(k)}+u_{2m+t+2}^{(k)}+u_{m+t+3}^{(k)})] ,$$

$$u_{t+2}^{(k+1)} = \frac{1}{2415}[712(u_{t+3}^{(k)}+u_{t-m+2}^{(k)})+225u_{t-m+1}^{(k)}+208(u_{2m+t+2}^{(k)}+u_{m+t+3}^{(k)})+$$
$$120u_{2m+t+1}^{(k)}+68(u_{t-m}^{(k)}+u_{t-1}^{(k)})+47(u_{m+t-1}^{(k)}+u_{2m+t}^{(k)})] ,$$

$$u_{m+t}^{(k+1)} = \frac{1}{2415}[712(u_{m+t-1}^{(k)}+u_{2m+t}^{(k)})+225u_{2m+t+1}^{(k)}+208(u_{t-m}^{(k)}+u_{t-1}^{(k)})+$$
$$120u_{t-m+1}^{(k)}+68(u_{2m+t+2}^{(k)}+u_{m+t+3}^{(k)})+47(u_{t+3}^{(k)}+u_{t-m+2}^{(k)})] ,$$

$$u_{m+t+1}^{(k+1)} = \frac{1}{161}[52u_{2m+t+1}^{(k)}+17u_{t-m+1}^{(k)}+15(u_{m+t-1}^{(k)}+u_{2m+t}^{(k)}+u_{2m+t+2}^{(k)}+u_{m+t+3}^{(k)}+$$
$$8(u_{t-m}^{(k)}+u_{t-1}^{(k)}+u_{t+3}^{(k)}+u_{t-m+2}^{(k)})] ,$$

and

$$u_{m+t+2}^{(k+1)} = \frac{1}{2415}[712(u_{2m+t+2}^{(k)}+u_{m+t+3}^{(k)})+225u_{2m+t+1}^{(k)}+208(u_{t+3}^{(k)}+u_{t-m+2}^{(k)})+$$
$$120u_{t-m+1}^{(k)}+68(u_{m+t-1}^{(k)}+u_{2m+t}^{(k)})+47(u_{t-m}^{(k)}+u_{t-1}^{(k)})] , \quad (4.2.25)$$

where $t=(rm+1),(3),(r+1)m-2$ and $r=0,(2),m-2$.

Similar set of equations can be written down for the 6-point group Gauss-Seidel, J.O.R. and S.O.R. iterative methods.

4.2.4 The 9-Point Group

In this grouping, each group is formed from the 9 points as shown in Figure (4.2.10).

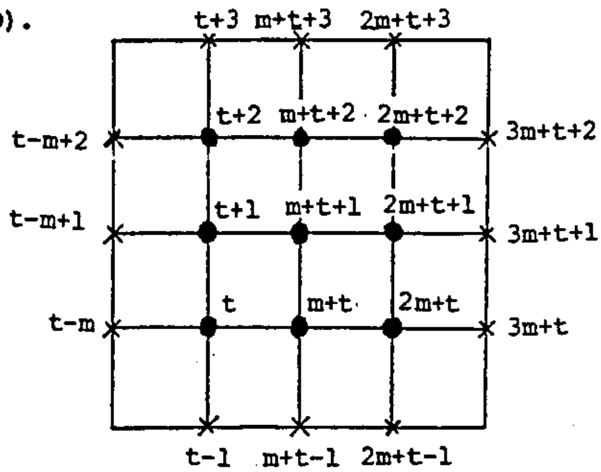


FIGURE (4.2.10)

where t=(rm+1),(3),(r+1)m-2 and r=0,(3),m-3. m must be divisible by 3, and each subset G_ℓ, $\ell=1,2,\ldots,\frac{m^2}{9}$ consists of 9 elements. By considering the linear system (3.11.1) and $\Delta x=\Delta y=1/7$, by applying the same strategy as previously with the 6 point group, the coefficient matrix A has the following block structure,

$$A = \left[\begin{array}{cc|cc} R_0 & 0 & R_2 & R_3 \\ 0 & R_0 & R_1 & R_4 \\ \hline R_4 & R_3 & R_0 & 0 \\ R_1 & R_2 & 0 & R_0 \end{array}\right] \tag{4.2.26a}$$

where,

$$R_0=\left[\begin{array}{ccc|ccc|ccc} 1 & \alpha_3 & 0 & \alpha_2 & & 0 & & & \\ \alpha_1 & 1 & \alpha_3 & & \alpha_2 & & & 0 & \\ 0 & \alpha_1 & 1 & 0 & & \alpha_2 & & & \\ \hline \alpha_4 & & 0 & 1 & \alpha_3 & 0 & \alpha_2 & & 0 \\ & \alpha_4 & & \alpha_1 & 1 & \alpha_3 & & \alpha_2 & \\ 0 & & \alpha_4 & 0 & \alpha_1 & 1 & 0 & & \alpha_2 \\ \hline & & & \alpha_4 & & 0 & 1 & \alpha_3 & 0 \\ & 0 & & & \alpha_4 & & \alpha_1 & 1 & \alpha_3 \\ & & & 0 & & \alpha_4 & 0 & \alpha_1 & 1 \end{array}\right], \quad R_1=\left[\begin{array}{ccc|ccc|ccc} 0 & 0 & \alpha_1 & & & & & & \\ 0 & 0 & 0 & & 0 & & & 0 & \\ 0 & 0 & 0 & & & & & & \\ \hline & & & 0 & 0 & \alpha_1 & & & \\ & 0 & & 0 & 0 & 0 & & 0 & \\ & & & 0 & 0 & 0 & & & \\ \hline & & & & & & 0 & 0 & \alpha_1 \\ & 0 & & & 0 & & 0 & 0 & 0 \\ & & & & & & 0 & 0 & 0 \end{array}\right],$$

$$R_2=\left[\begin{array}{ccc|ccc|ccc} & & & & & & & & \\ & 0 & & & 0 & & & 0 & \\ & & & & & & & & \\ \hline & & & & & & & & \\ & 0 & & & 0 & & & 0 & \\ & & & & & & & & \\ \hline \alpha_2 & 0 & 0 & & & & & & \\ 0 & \alpha_2 & 0 & & 0 & & & 0 & \\ 0 & 0 & \alpha_2 & & & & & & \end{array}\right], \quad R_3=\left[\begin{array}{ccc|ccc|ccc} 0 & 0 & 0 & & & & & & \\ 0 & 0 & 0 & & 0 & & & 0 & \\ \alpha_3 & 0 & 0 & & & & & & \\ \hline & & & 0 & 0 & 0 & & & \\ & 0 & & 0 & 0 & 0 & & 0 & \\ & & & \alpha_3 & 0 & 0 & & & \\ \hline & & & & & & 0 & 0 & 0 \\ & 0 & & & 0 & & 0 & 0 & 0 \\ & & & & & & \alpha_3 & 0 & 0 \end{array}\right]$$

and

$$R_4 = \left[\begin{array}{c:c:c} O & O & \begin{matrix} \alpha_4 & 0 & 0 \\ 0 & \alpha_4 & 0 \\ 0 & 0 & \alpha_4 \end{matrix} \\ \hdashline O & O & O \\ \hdashline O & O & O \end{array}\right] . \qquad (4.2.26b)$$

The inverse of the matrix R_0 can be obtained by the use of the system "REDUCE" which was described in Section (2.9). It is given by,

$$R_0^{-1} = \frac{1}{d} \left[\begin{array}{ccc:ccc:ccc}
\gamma_1 & \alpha_3\beta_1 & \alpha_3^2\beta_2 & \alpha_2\beta_3 & \alpha_2\alpha_3\beta_4 & \alpha_2\alpha_3^2\beta_5 & \alpha_2^2\beta_6 & \alpha_2^2\alpha_3\beta_7 & \alpha_2^2\alpha_3^2\beta_8 \\
\alpha_1\beta_1 & \gamma_2 & \alpha_3\beta_1 & \alpha_1\alpha_2\beta_4 & \alpha_2\beta_9 & \alpha_2\alpha_3\beta_4 & \alpha_1\alpha_2^2\beta_7 & \alpha_2^2\beta_{10} & \alpha_2^2\alpha_3\beta_7 \\
\alpha_1^2\beta_2 & \alpha_1\beta_1 & \gamma_1 & \alpha_1^2\alpha_2\beta_5 & \alpha_1\alpha_2\beta_4 & \alpha_2\beta_3 & \alpha_1^2\alpha_2^2\beta_8 & \alpha_1\alpha_2^2\beta_7 & \alpha_2^2\beta_6 \\ \hdashline
\alpha_4\beta_3 & \alpha_3\alpha_4\beta_4 & \alpha_3^2\alpha_4\beta_5 & \gamma_3 & \alpha_3\beta_{11} & \alpha_3^2\beta_{12} & \alpha_2\beta_3 & \alpha_2\alpha_3\beta_4 & \alpha_2\alpha_3^2\beta_5 \\
\alpha_1\alpha_4\beta_4 & \alpha_4\beta_9 & \alpha_3\alpha_4\beta_4 & \alpha_1\beta_{11} & \gamma_4 & \alpha_3\beta_{11} & \alpha_1\alpha_2\beta_4 & \alpha_2\beta_9 & \alpha_2\alpha_3\beta_4 \\
\alpha_1^2\alpha_4\beta_5 & \alpha_1\alpha_4\beta_4 & \alpha_4\beta_3 & \alpha_1^2\beta_{12} & \alpha_1\beta_{11} & \gamma_3 & \alpha_1^2\alpha_2\beta_5 & \alpha_1\alpha_2\beta_4 & \alpha_2\beta_3 \\ \hdashline
\alpha_4^2\beta_6 & \alpha_3\alpha_4^2\beta_7 & \alpha_3^2\alpha_4^2\beta_8 & \alpha_4\beta_3 & \alpha_3\alpha_4\beta_4 & \alpha_3^2\alpha_4\beta_5 & \gamma_1 & \alpha_3\beta_1 & \alpha_3^2\beta_2 \\
\alpha_1\alpha_4^2\beta_7 & \alpha_4^2\beta_{10} & \alpha_3\alpha_4^2\beta_7 & \alpha_1\alpha_4\beta_4 & \alpha_4\beta_9 & \alpha_3\alpha_4\beta_4 & \alpha_1\beta_1 & \gamma_2 & \alpha_3\beta_1 \\
\alpha_1^2\alpha_4^2\beta_8 & \alpha_1\alpha_4^2\beta_7 & \alpha_4^2\beta_6 & \alpha_1^2\alpha_4\beta_5 & \alpha_1\alpha_4\beta_4 & \alpha_4\beta_3 & \alpha_1^2\beta_2 & \alpha_1\beta_1 & \gamma_1
\end{array}\right] \qquad (4.2.27a)$$

where,

$$d = 4\alpha_1\alpha_2\alpha_3\alpha_4[4(\alpha_1\alpha_3-\alpha_2\alpha_4)^2-2(\alpha_1\alpha_3+\alpha_2\alpha_4)+3]-2\alpha_1\alpha_3(4\alpha_1^2\alpha_3^2-6\alpha_1\alpha_3+3)$$

$$-2\alpha_2\alpha_4(4\alpha_2^2\alpha_4^2-6\alpha_2\alpha_4+3)+1 . \qquad (4.2.27b)$$

$$\beta_1 = 2\alpha_1\alpha_2\alpha_3\alpha_4(4\alpha_1\alpha_3-6\alpha_2\alpha_4-1)-4\alpha_1\alpha_3(\alpha_1\alpha_3-1)+\alpha_2\alpha_4(4\alpha_2^2\alpha_4^2-4\alpha_2\alpha_4+3)-1,$$

$$\beta_2 = 2\alpha_1\alpha_2\alpha_3\alpha_4(4\alpha_2\alpha_4-2\alpha_1\alpha_3-3)+4\alpha_1\alpha_3(\alpha_1\alpha_3-1)-2\alpha_2^2\alpha_4^2(2\alpha_2\alpha_4-3)+1,$$

$$\beta_3 = 2\alpha_1\alpha_2\alpha_3\alpha_4(4\alpha_2\alpha_4-6\alpha_1\alpha_3-1)-4\alpha_2\alpha_4(\alpha_2\alpha_4-1)+\alpha_1\alpha_3(4\alpha_1^2\alpha_3^2-4\alpha_1\alpha_3+3)-1,$$

$$\beta_4 = 2[(2\alpha_1\alpha_3-1)(2\alpha_2\alpha_4-1)],$$

$$\beta_5 = 4\alpha_1\alpha_3(\alpha_2\alpha_4-\alpha_1\alpha_3+2)-2\alpha_2\alpha_4-3,$$

$$\beta_6 = 2\alpha_1\alpha_2\alpha_3\alpha_4(4\alpha_1\alpha_3-2\alpha_2\alpha_4-3)+4\alpha_2\alpha_4(\alpha_2\alpha_4-1)-2\alpha_2^2\alpha_4^2(2\alpha_1\alpha_3-1)+1\ ,$$

$$\beta_7 = 4\alpha_2\alpha_4(\alpha_1\alpha_3-\alpha_2\alpha_4+2)-2\alpha_1\alpha_3-3\ , \tag{4.2.27c}$$

$$\beta_8 = 2[\alpha_1\alpha_3(2\alpha_1\alpha_3-4\alpha_2\alpha_4-3)+\alpha_2\alpha_4(2\alpha_2\alpha_4-3)+3]\ ,$$

$$\beta_9 = -4\alpha_1\alpha_2\alpha_3\alpha_4(2\alpha_1\alpha_3-2\alpha_2\alpha_4+1)+4\alpha_1^2\alpha_3^2-4\alpha_2\alpha_4(\alpha_2\alpha_4-1)-1\ ,$$

$$\beta_{10} = 4\alpha_2\alpha_4(\alpha_2\alpha_4-3\alpha_1\alpha_3-1)+6\alpha_1\alpha_3+1.$$

$$\beta_{11} = 4\alpha_1\alpha_2\alpha_3\alpha_4(2\alpha_1\alpha_3-2\alpha_2\alpha_4-1)-4\alpha_1\alpha_3(\alpha_1\alpha_3-1)+4\alpha_2^2\alpha_4^2-1\ ,$$

$$\beta_{12} = 4\alpha_1\alpha_3(\alpha_1\alpha_3-3\alpha_2\alpha_4-1)+6\alpha_2\alpha_4+1\ ,$$

$$\gamma_1 = 2\alpha_1\alpha_2\alpha_3\alpha_4[2(\alpha_1\alpha_3-\alpha_2\alpha_4)^2-\alpha_1\alpha_3-\alpha_2\alpha_4+3]-\alpha_1\alpha_3(4\alpha_1^2\alpha_3^2-8\alpha_1\alpha_3+5)-\alpha_2\alpha_4 (4\alpha_2^2\alpha_4^2-8\alpha_2\alpha_4+5)+1\ ,$$

$$\gamma_2 = -2\alpha_1\alpha_2\alpha_3\alpha_4(4\alpha_1\alpha_3-2\alpha_2\alpha_4-3)+4\alpha_1\alpha_3(\alpha_1\alpha_3-1)-\alpha_2\alpha_4(4\alpha_2^2\alpha_4^2-8\alpha_2\alpha_4+5)+1\ ,$$

$$\gamma_3 = 4\alpha_1^2\alpha_3^2(\alpha_2\alpha_4-\alpha_1\alpha_3+2)-\alpha_1\alpha_3(8\alpha_2^2\alpha_4^2-6\alpha_2\alpha_4+5)+4\alpha_2\alpha_4(\alpha_2\alpha_4-1)+1\ ,$$

and

$$\gamma_4 = 4\alpha_1\alpha_2\alpha_3\alpha_4(3-2\alpha_1\alpha_3-2\alpha_2\alpha_4)+4\alpha_1\alpha_3(\alpha_1\alpha_3-1)+4\alpha_2\alpha_4(\alpha_2\alpha_4-1)+1\ .$$

Hence the matrix A^E of (4.2.4) can be formed, which has the same block structure as that of A with the submatrices R_O replaced by the identity matrices I and R_i, i=1,2,3,4, replaced by $R_O^{-1}R_i$, which can be easily determined, for example,

$$R_O^{-1}R_2=\frac{1}{d}\begin{pmatrix}
\alpha_2^3\beta_6 & \alpha_2^3\alpha_3\beta_7 & \alpha_2^3\alpha_3^2\beta_8 & 0 & 0 & 0 & 0 & 0 & 0\\
\alpha_1\alpha_2^3\beta_7 & \alpha_2^3\beta_{10} & \alpha_2^3\alpha_3\beta_7 & 0 & & & & & 0\\
\alpha_1^2\alpha_2^3\beta_8 & \alpha_1\alpha_2^3\beta_7 & \alpha_2^3\beta_6 & 0 & & & & & 0\\
\alpha_2^2\beta_3 & \alpha_2^2\alpha_3\beta_4 & \alpha_2^2\alpha_3^2\beta_5 & 0 & & & & & 0\\
\alpha_1\alpha_2^2\beta_4 & \alpha_2^2\beta_9 & \alpha_2^2\alpha_3\beta_4 & 0 & & & O & & 0\\
\alpha_1^2\alpha_2^2\beta_5 & \alpha_1\alpha_2^2\beta_4 & \alpha_2^2\beta_3 & 0 & & & & & 0\\
\alpha_2\gamma_1 & \alpha_2\alpha_3\beta_1 & \alpha_2\alpha_3^2\beta_2 & 0 & & & & & 0\\
\alpha_1\alpha_2\beta_1 & \alpha_2\gamma_2 & \alpha_2\alpha_3\beta_1 & 0 & & & & & 0\\
\alpha_1^2\alpha_2\beta_2 & \alpha_1\alpha_2\beta_1 & \alpha_2\gamma_1 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}, \quad (4.2.28)$$

whilst $R_O^{-1}R_1, R_O^{-1}R_3$ and $R_O^{-1}R_4$ can also be obtained.

For the model problem (3.2.1), $\alpha_1=\alpha_2=\alpha_3=\alpha_4=-\frac{1}{4}$, and we have,

$$R_O^{-1}=\frac{1}{56}\left[\begin{array}{ccc|ccc|ccc}
67 & 22 & 7 & 22 & 14 & 6 & 7 & 6 & 3\\
22 & 74 & 22 & 14 & 28 & 14 & 6 & 10 & 6\\
7 & 22 & 67 & 6 & 14 & 22 & 3 & 6 & 7\\
22 & 14 & 6 & 74 & 28 & 10 & 22 & 14 & 6\\
14 & 28 & 14 & 28 & 84 & 28 & 14 & 28 & 14\\
6 & 14 & 22 & 10 & 28 & 74 & 6 & 14 & 22\\
7 & 6 & 3 & 22 & 14 & 6 & 67 & 22 & 7\\
6 & 10 & 6 & 14 & 28 & 14 & 22 & 74 & 22\\
3 & 6 & 7 & 6 & 14 & 22 & 7 & 22 & 67
\end{array}\right]. \quad (4.2.29)$$

Hence,

$$R_O^{-1}R_1=-\frac{1}{224}\begin{bmatrix}
0 & 0 & 67 & 0 & 0 & 22 & 0 & 0 & 7\\
0 & 0 & 22 & 0 & 0 & 14 & 0 & 0 & 6\\
0 & 0 & 7 & 0 & 0 & 6 & 0 & 0 & 3\\
0 & 0 & 22 & 0 & 0 & 74 & 0 & 0 & 22\\
0 & 0 & 14 & 0 & 0 & 28 & 0 & 0 & 14\\
0 & 0 & 6 & 0 & 0 & 10 & 0 & 0 & 6\\
0 & 0 & 7 & 0 & 0 & 22 & 0 & 0 & 67\\
0 & 0 & 6 & 0 & 0 & 14 & 0 & 0 & 22\\
0 & 0 & 3 & 0 & 0 & 6 & 0 & 0 & 7
\end{bmatrix}, \quad (4.2.30a)$$

$$R_O^{-1}R_2 = -\frac{1}{224}\begin{bmatrix} 7 & 6 & 3 & \\ 6 & 10 & 6 & \\ 3 & 6 & 7 & \\ 22 & 14 & 6 & \\ 14 & 28 & 14 & O \\ 6 & 14 & 22 & \\ 67 & 22 & 7 & \\ 22 & 74 & 22 & \\ 7 & 22 & 67 & \end{bmatrix} . \qquad (4.2.30b)$$

Similarly, $R_O^{-1}R_3$ and $R_O^{-1}R_4$ can be obtained, and hence the computational molecule can be established at the points P_1, P_2 and P_3 to be,

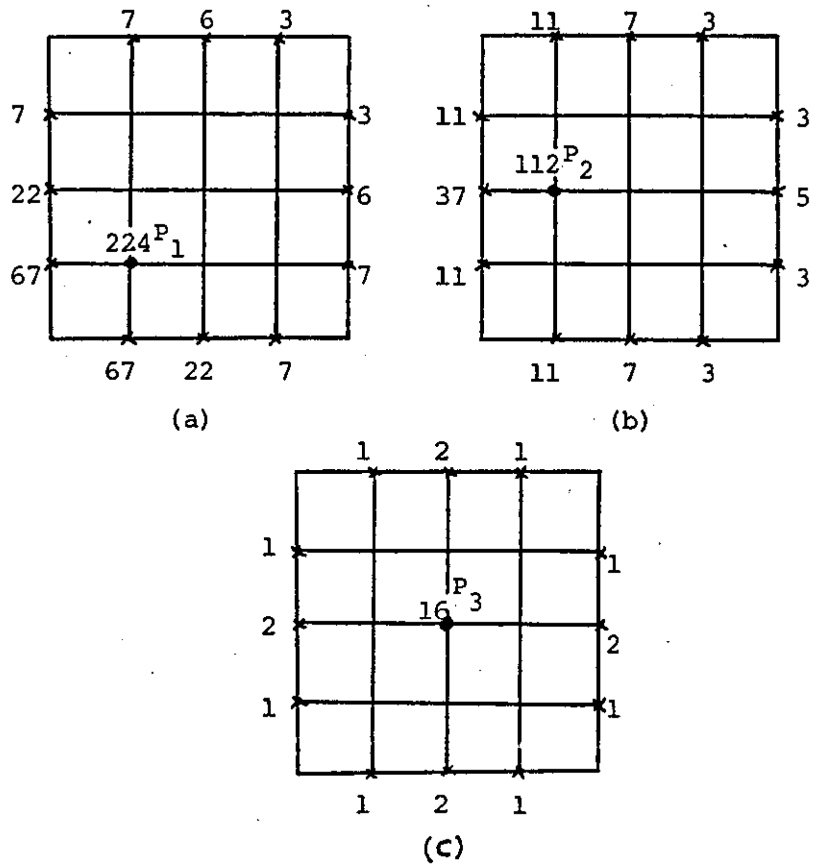


FIGURE (4.2.11)

From the explicit equations derived from the new molecule the explicit group Jacobi method can be obtained and is given by,

$$\begin{aligned}
u_{t}^{(k+1)} &= \frac{1}{224}[67(u_{t-m}^{(k)}+u_{t-1}^{(k)})+22(u_{m+t-1}^{(k)}+u_{t-m+1}^{(k)})+7(u_{2m+t-1}^{(k)}+u_{3m+t}^{(k)}+\\
&\qquad u_{t+3}^{(k)}+u_{t-m+2}^{(k)})+6(u_{3m+t+1}^{(k)}+u_{m+t+3}^{(k)})+3(u_{3m+t+2}^{(k)}+u_{2m+t+3}^{(k)})]\ ,\\
u_{t+1}^{(k+1)} &= \frac{1}{112}[37u_{t-m+1}^{(k)}+11(u_{t-m}^{(k)}+u_{t-1}^{(k)}+u_{t+3}^{(k)}+u_{t-m+2}^{(k)})+7(u_{m+t-1}^{(k)}+u_{m+t+3}^{(k)})+\\
&\qquad 5u_{3m+t+1}^{(k)}+3(u_{2m+t-1}^{(k)}+u_{3m+t}^{(k)}+u_{3m+t+2}^{(k)}+u_{2m+t+3}^{(k)})]\ ,\\
u_{t+2}^{(k+1)} &= \frac{1}{224}[67(u_{t+3}^{(k)}+u_{t-m+2}^{(k)})+22(u_{m+t+3}^{(k)}+u_{t-m+1}^{(k)})+7(u_{3m+t+2}^{(k)}+u_{2m+t+3}^{(k)}+\\
&\qquad u_{t-m}^{(k)}+u_{t-1}^{(k)})+6(u_{m+t-1}^{(k)}+u_{3m+t+1}^{(k)})+3(u_{2m+t-1}^{(k)}+u_{3m+t}^{(k)})]\ ,\\
u_{m+t}^{(k+1)} &= \frac{1}{112}[37u_{m+t-1}^{(k)}+11(u_{t-m}^{(k)}+u_{t-1}^{(k)}+u_{2m+t-1}^{(k)}+u_{3m+t}^{(k)})+7(u_{3m+t+1}^{(k)}+u_{t-m+1}^{(k)})+\\
&\qquad 5u_{m+t+3}^{(k)}+3(u_{3m+t+2}^{(k)}+u_{2m+t+3}^{(k)}+u_{t+3}^{(k)}+u_{t-m+2}^{(k)})]\ ,\\
u_{m+t+1}^{(k+1)} &= \frac{1}{16}[2(u_{m+t-1}^{(k)}+u_{3m+t+1}^{(k)}+u_{m+t+3}^{(k)}+u_{t-m+1}^{(k)})+u_{t-m}^{(k)}+u_{t-1}^{(k)}+u_{2m+t-1}^{(k)}+\\
&\qquad u_{3m+t}^{(k)}+u_{3m+t+2}^{(k)}+u_{2m+t+3}^{(k)}+u_{t+3}^{(k)}+u_{t-m+2}^{(k)}]\ ,\\
u_{m+t+2}^{(k+1)} &= \frac{1}{112}[37u_{m+t+3}^{(k)}+11(u_{3m+t+2}^{(k)}+u_{2m+t+3}^{(k)}+u_{t+3}^{(k)}+u_{t-m+2}^{(k)})+7(u_{3m+t+1}^{(k)}+\\
&\qquad u_{t-m+1}^{(k)})+5u_{m+t-1}^{(k)}+3(u_{t-m}^{(k)}+u_{t-1}^{(k)}+u_{2m+t-1}^{(k)}+u_{3m+t}^{(k)})]\ ,\\
u_{2m+t}^{(k+1)} &= \frac{1}{224}[67(u_{2m+t-1}^{(k)}+u_{3m+t}^{(k)})+22(u_{m+t-1}^{(k)}+u_{3m+t+1}^{(k)})+7(u_{t-m}^{(k)}+u_{t-1}^{(k)}+\\
&\qquad u_{3m+t+2}^{(k)}+u_{2m+t+3}^{(k)})+6(u_{m+t+3}^{(k)}+u_{t-m+1}^{(k)})+3(u_{t+3}^{(k)}+u_{t-m+2}^{(k)})]\ ,\\
u_{2m+t+1}^{(k+1)} &= \frac{1}{112}[37u_{3m+t+1}^{(k)}+11(u_{2m+t-1}^{(k)}+u_{3m+t}^{(k)}+u_{3m+t+2}^{(k)}+u_{2m+t+3}^{(k)})+7(u_{m+t-1}^{(k)}+\\
&\qquad u_{m+t+3}^{(k)})+5u_{t-m-1}^{(k)}+3(u_{t-m}^{(k)}+u_{t-1}^{(k)}+u_{t+3}^{(k)}+u_{t-m+2}^{(k)})]\ ,
\end{aligned}$$

+1

$$u^{(k+1)}_{2m+t+2} = \frac{1}{224}[67(u^{(k)}_{3m+t+2}+u^{(k)}_{2m+t+3})+22(u^{(k)}_{3m+t+1}+u^{(k)}_{m+t+3})+7(u^{(k)}_{2m+t-1}+ u^{(k)}_{3m+t}+u^{(k)}_{t+3}+u^{(k)}_{t-m+2})+6(u^{(k)}_{t+m-1}+u^{(k)}_{t-m+1})+3(u^{(k)}_{t-m}+u^{(k)}_{t-1})] \;, \tag{4.2.31}$$

where $t=(rm+1),(3),(r+1)m-2$ and $r=0,(3),m-3$.

A similar set of equations can be derived for the 9-point Gauss-Seidel, J.O.R. and S.O.R. iterative methods.

4.2.5 The 12-Point Group

Assume that the mesh points in the region of Figure (4.2.1) are ordered in groups of 12 as shown in Figure (4.2.12).

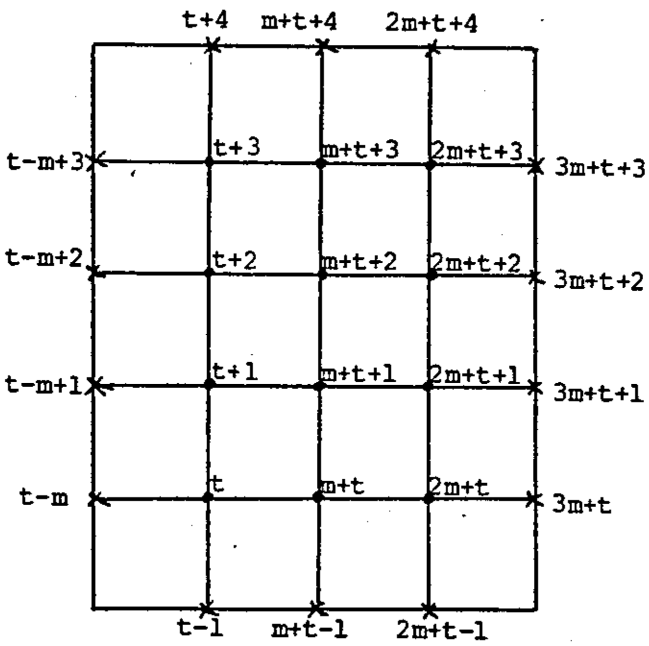


FIGURE (4.2.12)

where $t=rm+1,(4),(r+1)m-3$ and $r=0,(3),m-3$. In this case G_ℓ, $\ell=1,2,\ldots \frac{m^2}{12}$, consists of 12 elements, and m must be divisible by 12. By considering the linear system (3.11.1) defined in a unit square with $\Delta x=\Delta y=\frac{1}{13}$, and assuming that the groups are ordered in red-black ordering, then by applying equation (4.2.1) to each of the mesh points in turn, the resulting coefficient matrix A has the block structure,

$$A = \left[\begin{array}{cccccc|cccccc}
R_0 & & & & & & R_2 & 0 & R_3 & & & \\
 & R_0 & & \bigcirc & & & R_4 & R_2 & 0 & R_3 & \bigcirc & \\
 & & R_0 & & & & R_1 & 0 & R_4 & R_2 & R_3 & \\
 & & & R_0 & & & & R_1 & 0 & R_4 & 0 & R_3 \\
 & \bigcirc & & & R_0 & & \bigcirc & & R_1 & 0 & R_2 & 0 \\
 & & & & & R_0 & & & & R_1 & R_4 & R_2 \\
\hline
R_4 & R_2 & R_3 & & & & R_0 & & & & & \\
0 & R_4 & 0 & R_3 & \bigcirc & & & R_0 & & & \bigcirc & \\
R_1 & 0 & R_2 & 0 & R_3 & & & & R_0 & & & \\
 & R_1 & R_4 & R_2 & 0 & R_3 & & & & R_0 & & \\
\bigcirc & & R_1 & 0 & R_4 & R_2 & & \bigcirc & & & R_0 & \\
 & & & R_1 & 0 & R_4 & & & & & & R_0
\end{array}\right] \quad (4.2.32a)$$

where,

$$R_0 = \left[\begin{array}{cccc|cccc|cccc}
1 & \alpha_3 & \bigcirc & & \alpha_2 & & & & & & & \\
\alpha_1 & 1 & \alpha_3 & & & \alpha_2 & \bigcirc & & & & & \\
 & \alpha_1 & 1 & \alpha_3 & & \bigcirc & \alpha_2 & & & \bigcirc & & \\
\bigcirc & & \alpha_1 & 1 & & & & \alpha_2 & & & & \\
\hline
\alpha_4 & & & & 1 & \alpha_3 & \bigcirc & & \alpha_2 & & \bigcirc & \\
 & \alpha_4 & \bigcirc & & \alpha_1 & 1 & \alpha_3 & & & \alpha_2 & & \\
\bigcirc & & \alpha_4 & & \bigcirc & \alpha_1 & 1 & \alpha_3 & \bigcirc & & \alpha_2 & \\
 & & & \alpha_4 & & & \alpha_1 & 1 & & & & \alpha_2 \\
\hline
 & & & & \alpha_4 & & \bigcirc & & 1 & \alpha_3 & \bigcirc & \\
 & \bigcirc & & & & \alpha_4 & & & \alpha_1 & 1 & \alpha_3 & \\
 & & & & & & \alpha_4 & & & \alpha_1 & 1 & \alpha_3 \\
 & & & & \bigcirc & & & \alpha_4 & \bigcirc & & \alpha_1 & 1
\end{array}\right] , \quad (4.2.32b)$$

$$R_1=\left[\begin{array}{c|c|c} \begin{smallmatrix}0&0&0&\alpha_1\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{smallmatrix} & O & O \\ \hline O & \overset{\alpha_1}{O} & O \\ \hline O & O & \overset{\alpha_1}{O}\end{array}\right], \quad R_2=\left[\begin{array}{c|c|c} O & O & O \\ \hline O & O & O \\ \hline \begin{smallmatrix}\alpha_2 & & & O\\ & \alpha_2 & & \\ & & \alpha_2 & \\ O & & & \alpha_2\end{smallmatrix} & O & O\end{array}\right]$$

$$R_3=\left[\begin{array}{c|c|c} \begin{smallmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\\alpha_3&0&0&0\end{smallmatrix} & O & O \\ \hline O & \underset{\alpha_3}{O} & O \\ \hline O & O & \underset{\alpha_3}{O}\end{array}\right], \text{ and } R_4=\left[\begin{array}{c|c|c} O & O & \begin{smallmatrix}\alpha_4 & & & O\\ & \alpha_4 & & \\ & & \alpha_4 & \\ O & & & \alpha_4\end{smallmatrix} \\ \hline O & O & O \\ \hline O & O & O\end{array}\right].$$

By using REDUCE, the inverse of the matrix R_0 can be obtained, and it is given in Appendix A. The matrix A^E of (4.2.4) can now be formed, which has the same block structure as that of A with the submatrices R_0 replaced by the identity matrices I and R_i, i=1,2,3,4, replaced by $R_0^{-1}R_i$, which can be easily determined, since if R_i, i=1,2,3,4, has a column or row of zeros so does $R_0^{-1}R_i$, and where an element α_i occurs as the $(p,q)^{th}$ element of R_i, the q^{th} column of $R_0^{-1}R_i$ is the p^{th}

column of R_0^{-1}, multiplied by α_i.

For the model problem (3.2.1), $\alpha_1=\alpha_2=\alpha_3=\alpha_4=-1/4$, we have R_0^{-1} symmetric and is given by,

$$R_0^{-1} = \frac{1}{1380027}\begin{bmatrix}
1655792 & & & & & & & & & & & \\
554644 & 1857424 & & & & & & & & & & \\
201632 & 623444 & 1857424 & & & & & & & & & \\
68800 & 201632 & 554644 & 1655792 & & & & \text{symmetric} & & & & \\
548416 & 361152 & 183084 & 73568 & 1832512 & & & & & & & \\
361152 & 731500 & 434720 & 183084 & 713108 & 2130128 & & & & & & \\
183084 & 434720 & 731500 & 361152 & 297616 & 824296 & 2130128 & & & & & \\
73568 & 183084 & 361152 & 548416 & 111188 & 297616 & 713108 & 1832512 & & & & \\
176720 & 158464 & 95984 & 42388 & 548416 & 361152 & 183084 & 73568 & 1655792 & & & \\
158464 & 272704 & 200852 & 95984 & 361152 & 731500 & 434720 & 183084 & 554644 & 1857424 & & \\
95984 & 200852 & 272704 & 158464 & 183084 & 434720 & 731500 & 361152 & 201632 & 623444 & 1857424 & \\
42388 & 95984 & 158464 & 176720 & 73568 & 183084 & 361152 & 548416 & 68800 & 201632 & 554644 & 1655792
\end{bmatrix}$$

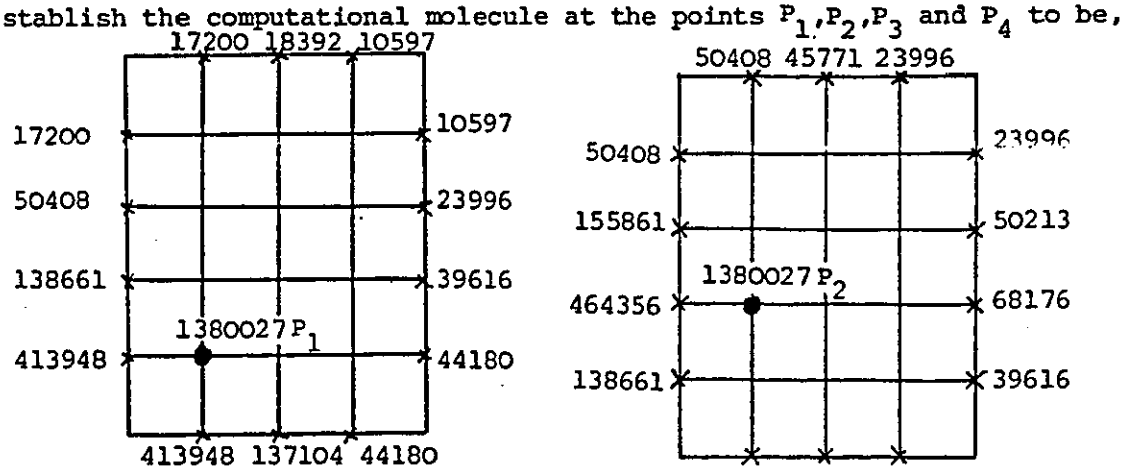
(4.2.33)

Hence,

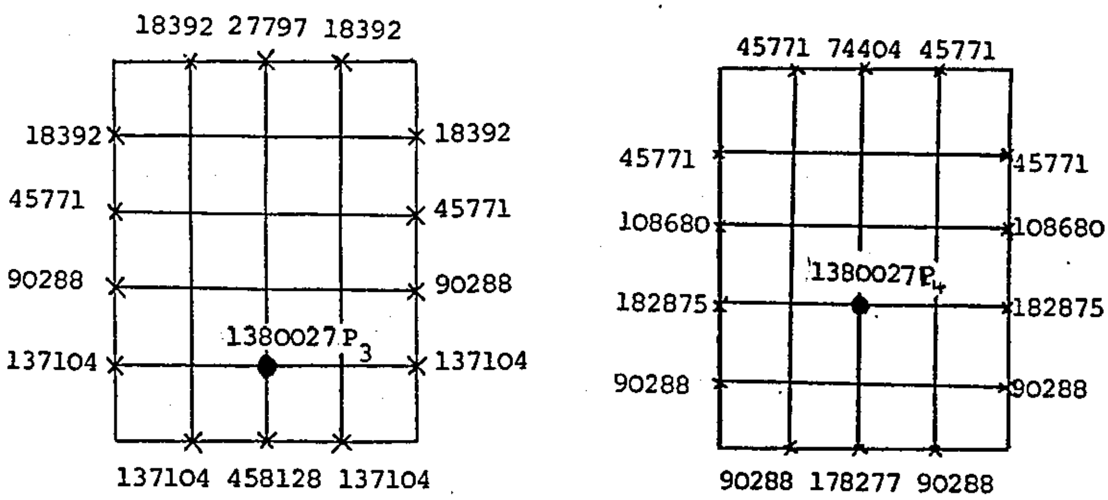
$$R_0^{-1}R_1=-\frac{1}{1380027}\begin{pmatrix}
0&0&0&413948&0&0&0&137104&0&0&0&44180\\
0&0&0&138661&0&0&0&90288&0&0&0&39616\\
0&0&0&50408&0&0&0&45771&0&0&0&23996\\
0&0&0&17200&0&0&0&18392&0&0&0&10597\\
0&0&0&137104&0&0&0&458128&0&0&0&137103\\
0&0&0&90288&0&0&0&178277&0&0&0&90288\\
0&0&0&45771&0&0&0&74404&0&0&0&45771\\
0&0&0&18392&0&0&0&27797&0&0&0&18392\\
0&0&0&44180&0&0&0&137103&0&0&0&413948\\
0&0&0&39616&0&0&0&90288&0&0&0&138661\\
0&0&0&23996&0&0&0&45771&0&0&0&50408\\
0&0&0&10597&0&0&0&18392&0&0&0&17200
\end{pmatrix}, \quad (4.2.34a)$$

$$R_0^{-1}R_4=-\frac{1}{1380027}\left(\begin{array}{c|cccc}
 & 413948 & 138661 & 50408 & 17200\\
 & 138661 & 464356 & 155861 & 50408\\
 & 50408 & 155861 & 464356 & 138661\\
 & 17200 & 50408 & 138661 & 413948\\
 & 137104 & 90288 & 45771 & 18392\\
 & 90288 & 182875 & 108680 & 45771\\
0 & 45771 & 108680 & 182875 & 90288\\
 & 18392 & 45771 & 90288 & 137104\\
 & 44180 & 39616 & 23996 & 10597\\
 & 39616 & 68176 & 50213 & 23996\\
 & 23996 & 50213 & 68176 & 39616\\
 & 10597 & 23996 & 39616 & 44180
\end{array}\right), \quad (4.2.34b)$$

Similarly, $R_0^{-1}R_2$ and $R_0^{-1}R_3$ can be determined, which enables us to establish the computational molecule at the points P_1, P_2, P_3 and P_4 to be,



(a) (b)



(c) (d)

FIGURE (4.2.13)

Hence, an explicit group Gauss-Seidel method can be derived from this molecule and is given by,

$$u_t^{(k+1)} = \frac{1}{1380027}[413948(u_{t-m}^{(k+1)}+u_{t-1}^{(k+1)})+138661u_{t-m+1}^{(k+1)}+137104u_{m+t-1}^{(k+1)}+$$

$$+50408u_{t-m+2}^{(k+1)}+44180(u_{2m+t-1}^{(k+1)}+u_{3m+t}^{(k)})+39616u_{3m+k+1}^{(k)}+23996u_{3m+t+2}^{(k)}+$$

$$+18392u_{m+t+4}^{(k)}+17200(u_{t-m+3}^{(k+1)}+u_{t+4}^{(k)})+10597(u_{2m+t+4}^{(k)}+u_{3m+t+3}^{(k)})] ,$$

$u_{t+3}^{(k+1)}, u_{2m+t+3}^{(k+1)}$ and $u_{2m+t}^{(k+1)}$ are found from completely analogous expressions obtained by rotating the mesh points clockwise as given in Figure (4.2.12).

$$u_{t+1}^{(k+1)} = \frac{1}{1380027}[464356u_{t-m+1}^{(k+1)}+155861u_{t-m+2}^{(k+1)}+138661(u_{t-1}^{(k+1)}+u_{t-m}^{(k+1)})$$

$$+90288u_{m+t-1}^{(k+1)}+68176u_{3m+t+1}^{(k)}+50408(u_{t-m+3}^{(k+1)}+u_{t+4}^{(k)})$$

$$+50213u_{3m+t+2}^{(k)}+45771u_{m+t+4}^{(k)}+39616(u_{2m+t-1}^{(k+1)}+u_{3m+t}^{(k)})$$

$$+23996(u_{2m+t+4}^{(k)}+u_{2m+t+3}^{(k)})] ,$$

$u_{t+2}^{(k+1)}, u_{2m+t+2}^{(k+1)}$ and $u_{2m+t+1}^{(k+1)}$ can be obtained similarly,

$$u_{m+t}^{(k+1)} = \frac{1}{1380027}[458128u_{m+t-1}^{(k+1)}+137104(u_{t-m}^{(k+1)}+u_{t-1}^{(k+1)}+u_{2m+t-1}^{(k+1)}+u_{3m+t}^{(k)})$$

$$+90288(u^{(k+1)}_{t-m+1}+u^{(k)}_{3m+t+1})+45771(u^{(k+1)}_{t-m+2}+u^{(k)}_{3m+t+2})+27797u^{(k)}_{m+t+4}$$

$$+18392(u^{(k)}_{3m+t+3}+u^{(k)}_{2m+t+4}+u^{(k)}_{t+4}+u^{(k+1)}_{t-m+3})] ,$$

similarly for $u^{(k+1)}_{m+t+3}$.

$$u^{(k+1)}_{m+t+1} = \frac{1}{1380027}[182875(u^{(k+1)}_{t-m+1}+u^{(k)}_{3m+t+1})+178277u^{(k+1)}_{m+t-1}+108680(u^{(k+1)}_{t-m+2}$$

$$+u^{(k)}_{3m+t+2})+90288(u^{(k+1)}_{t-m}+u^{(k+1)}_{t-1}+u^{(k+1)}_{2m+t-1}+u^{(k)}_{3m+t})+74404u^{(k)}_{m+t+4}$$

$$+45771(u^{(k)}_{3m+t+3}+u^{(k)}_{2m+t+4}+u^{(k)}_{t+4}+u^{(k+1)}_{t-m+3})] , \tag{4.2.35}$$

similarly, for the point $u^{(k+1)}_{m+t+2}$,

where $t=rm+1,(4),(r+1)m-3$ and $r=0,(3),m-3$.

Clearly a similar set of equations can be written down for the 12-point Jacobi, J.O.R. and S.O.R. iterative methods.

4.2.6 The 16-Point Group

Each group here is formed from the 16 points as shown in Figure (4.2.14),

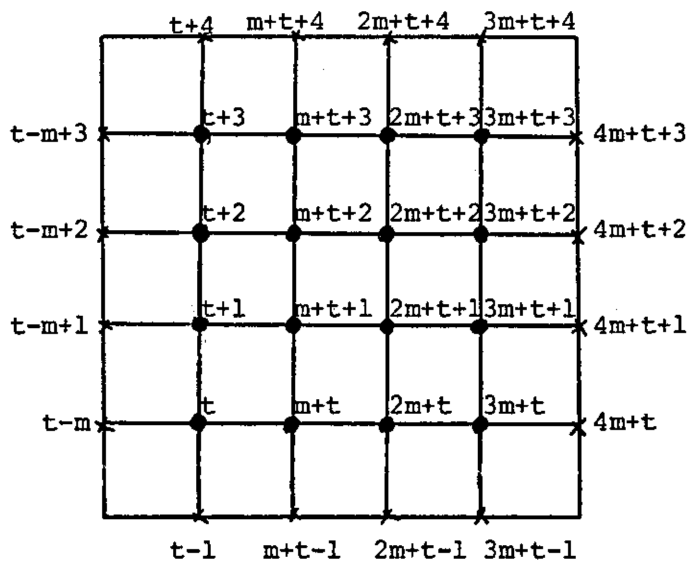


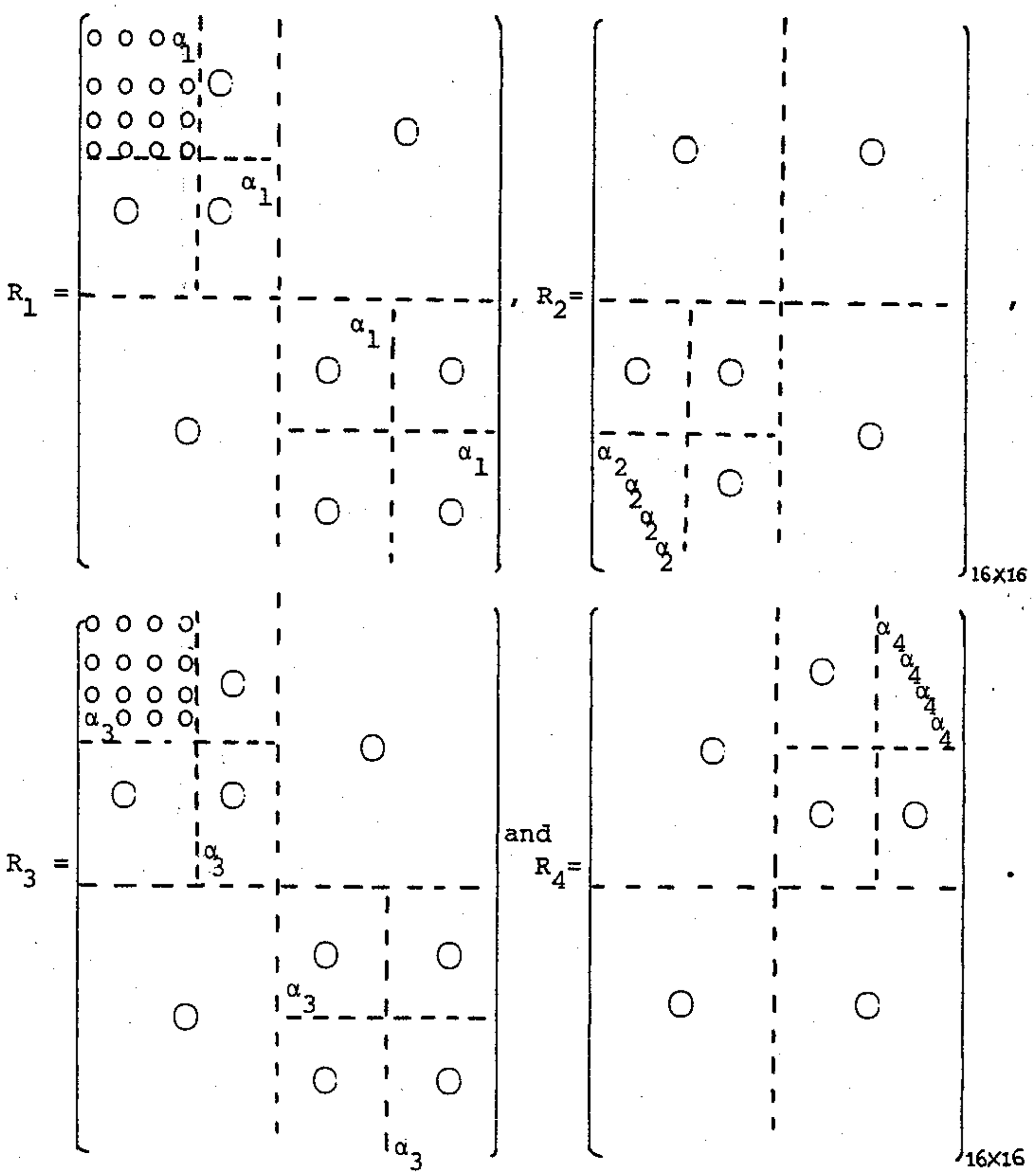
FIGURE (4.2.14)

where $t=rm+1,(4),(r+1)m-3$ and $r=0,(4),m-4$. In this case m must be divisible by 4. By considering the linear system (3.11.1) defined in a unit square with $\Delta x=\Delta y=1/9$, then by applying the same strategy as previously with the 12-point group, the coefficient matrix A has the following block structure,

$$A = \left[\begin{array}{cc|cc} R_0 & O & R_2 & R_3 \\ O & R_0 & R_1 & R_4 \\ \hline R_4 & R_3 & R_0 & O \\ R_1 & R_2 & O & R_0 \end{array}\right] \tag{4.2.36a}$$

where,

$$R_0 = \left[\begin{array}{cccc|cccc|cccc|cccc}
1 & \alpha_3 & 0 & 0 & \alpha_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\alpha_1 & 1 & \alpha_3 & 0 & 0 & \alpha_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \alpha_1 & 1 & \alpha_3 & 0 & 0 & \alpha_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \alpha_1 & 1 & 0 & 0 & 0 & \alpha_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
\alpha_4 & 0 & 0 & 0 & 1 & \alpha_3 & 0 & 0 & \alpha_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \alpha_4 & 0 & 0 & \alpha_1 & 1 & \alpha_3 & 0 & 0 & \alpha_2 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \alpha_4 & 0 & 0 & \alpha_1 & 1 & \alpha_3 & 0 & 0 & \alpha_2 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \alpha_4 & 0 & 0 & \alpha_1 & 1 & 0 & 0 & 0 & \alpha_2 & 0 & 0 & 0 & 0 \\
\hline
0 & 0 & 0 & 0 & \alpha_4 & 0 & 0 & 0 & 1 & \alpha_3 & 0 & 0 & \alpha_2 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \alpha_4 & 0 & 0 & \alpha_1 & 1 & \alpha_3 & 0 & 0 & \alpha_2 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \alpha_4 & 0 & 0 & \alpha_1 & 1 & \alpha_3 & 0 & 0 & \alpha_2 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \alpha_4 & 0 & 0 & \alpha_1 & 1 & 0 & 0 & 0 & \alpha_2 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \alpha_4 & 0 & 0 & 0 & 1 & \alpha_3 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \alpha_4 & 0 & 0 & \alpha_1 & 1 & \alpha_3 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \alpha_4 & 0 & 0 & \alpha_1 & 1 & \alpha_3 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \alpha_4 & 0 & 0 & \alpha_1 & 1
\end{array}\right] \tag{4.2.36b}$$



By assuming that $\alpha_1=\alpha_2=\alpha_3=\alpha_4=\alpha$ in R_0, R_1, R_2, R_3 and R_4, then by using REDUCE, R_0^{-1} can be determined, and it is given in Appendix B, and hence $R_0^{-1}R_i$, i=1,2,3,4, can be easily obtained.

For the model problem (3.2.1), $\alpha=-1/4$, it follows that R_0^{-1} is a symmetric matrix and is given by,

$$R_O^{-1} = \frac{1}{1650}\left[\begin{array}{cccccccccccccccc}
1987 &&&&&&&&&&&&&&& \\
674 & 2238 &&&&&&&&&&&&&& \\
251 & 762 & 2238 &&&&&&&&&&&&& \\
88 & 251 & 674 & 1987 &&&&&&&&&&&& \\
674 & 458 & 242 & 101 & 2238 &&&&&&& \text{symmetric} &&&& \\
458 & 916 & 559 & 242 & 916 & 2647 &&&&&&&&&& \\
242 & 559 & 916 & 458 & 409 & 1078 & 2647 &&&&&&&&& \\
101 & 242 & 458 & 674 & 162 & 409 & 916 & 2238 &&&&&&&& \\
251 & 242 & 158 & 74 & 762 & 559 & 316 & 138 & 2238 &&&&&&& \\
242 & 409 & 316 & 158 & 559 & 1078 & 697 & 316 & 916 & 2647 &&&&&& \\
158 & 316 & 409 & 242 & 316 & 697 & 1078 & 559 & 409 & 1078 & 2647 &&&&& \\
74 & 158 & 242 & 251 & 138 & 316 & 559 & 762 & 162 & 409 & 916 & 2238 &&&& \\
88 & 101 & 74 & 37 & 251 & 242 & 158 & 74 & 674 & 458 & 242 & 101 & 1987 &&& \\
101 & 162 & 138 & 74 & 242 & 409 & 316 & 158 & 458 & 916 & 559 & 242 & 674 & 2238 && \\
74 & 138 & 162 & 101 & 158 & 316 & 409 & 242 & 242 & 559 & 916 & 458 & 251 & 762 & 2238 & \\
37 & 74 & 101 & 88 & 74 & 158 & 242 & 251 & 101 & 242 & 458 & 674 & 88 & 251 & 674 & 1987
\end{array}\right] \tag{4.2.37}$$

Hence,

$$R_0^{-1}R_1 = -\frac{1}{6600}\begin{pmatrix}
0 & 0 & 0 & 1987 & 0 & 0 & 0 & 674 & 0 & 0 & 0 & 251 & 0 & 0 & 0 & 88 \\
 & & & 674 & & & & 458 & & & & 242 & & & & 101 \\
 & & & 251 & & & & 242 & & & & 158 & & & & 74 \\
 & & & 88 & & & & 101 & & & & 74 & & & & 37 \\
 & & & 674 & & & & 2238 & & & & 762 & & & & 251 \\
 & & & 458 & & & & 916 & & & & 559 & & & & 242 \\
 & & & 242 & & & & 409 & & & & 316 & & & & 158 \\
 & \bigcirc & & 101 & & \bigcirc & & 162 & & \bigcirc & & 138 & & \bigcirc & & 74 \\
 & & & 251 & & & & 762 & & & & 2238 & & & & 674 \\
 & & & 242 & & & & 559 & & & & 916 & & & & 458 \\
 & & & 158 & & & & 316 & & & & 409 & & & & 242 \\
 & & & 74 & & & & 138 & & & & 162 & & & & 101 \\
 & & & 88 & & & & 251 & & & & 674 & & & & 1987 \\
 & & & 101 & & & & 242 & & & & 458 & & & & 674 \\
 & & & 74 & & & & 158 & & & & 242 & & & & 251 \\
0 & 0 & 0 & 37 & 0 & 0 & 0 & 74 & 0 & 0 & 0 & 101 & 0 & 0 & 0 & 88
\end{pmatrix}, \quad (4.2.38a)$$

$$R_0^{-1}R_2 = -\frac{1}{6600}\begin{pmatrix}
88 & 101 & 74 & 37 & \\
101 & 162 & 138 & 74 & \\
74 & 138 & 162 & 101 & \\
37 & 74 & 101 & 88 & \\
251 & 242 & 158 & 74 & \\
242 & 409 & 316 & 158 & \\
158 & 316 & 409 & 242 & \\
74 & 158 & 242 & 251 & \\
674 & 458 & 242 & 101 & \bigcirc \\
458 & 916 & 559 & 242 & \\
242 & 559 & 916 & 458 & \\
101 & 242 & 458 & 674 & \\
1987 & 674 & 251 & 88 & \\
674 & 2238 & 762 & 251 & \\
251 & 762 & 2238 & 674 & \\
88 & 251 & 674 & 1987 &
\end{pmatrix}, \quad (4.2.38b)$$

$R_0^{-1}R_3$ and $R_0^{-1}R_4$ can be determined similarly. The computational molecule can now be established, it is given in Figure (4.2.15) at the points P_1, P_2 and P_3.

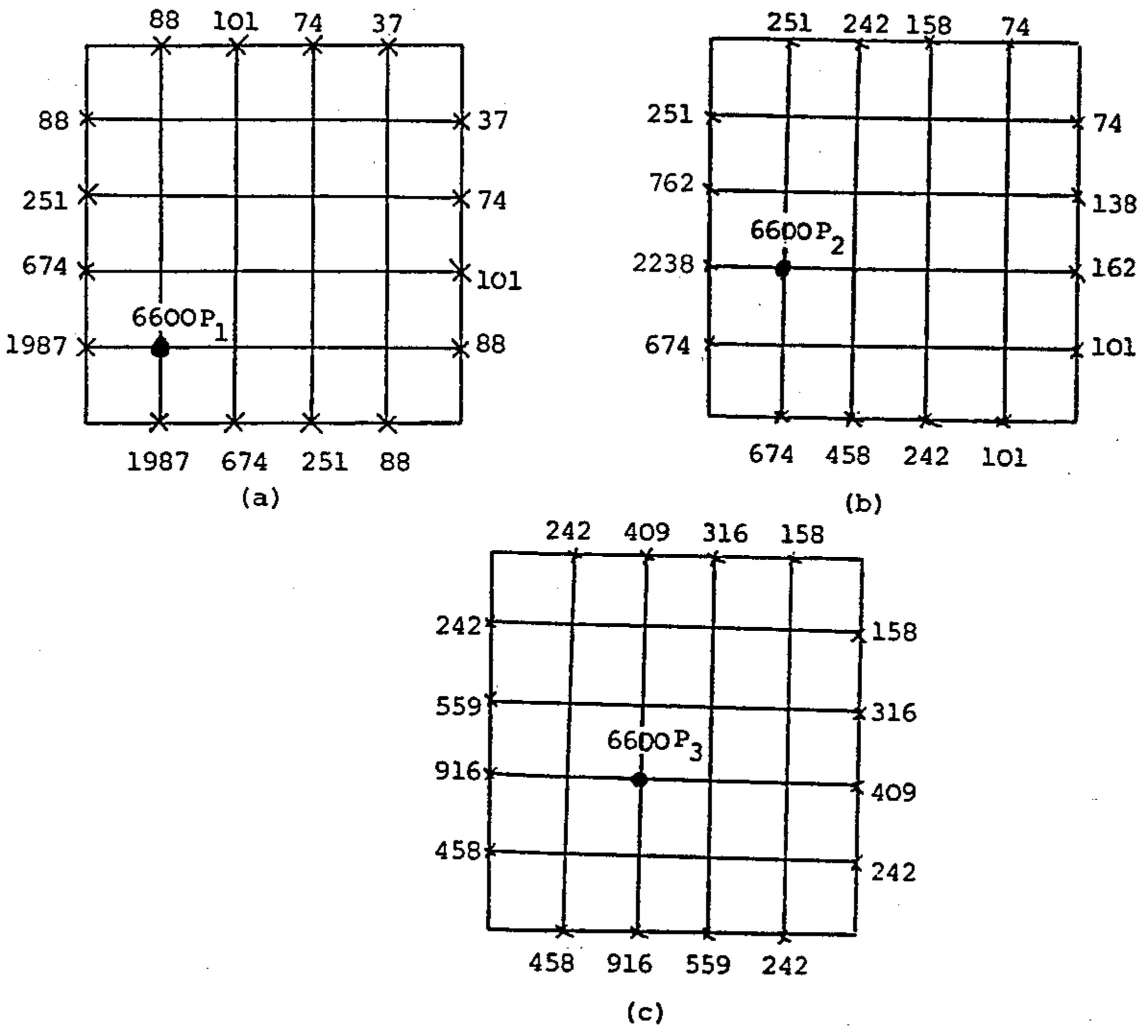


FIGURE (4.2.15)

An explicit 16-point group Jacobi iterative method can be derived from this molecule and is given by,

$$\begin{aligned} u_t^{(k+1)} = \frac{1}{6600}[&1987(u_{t-m}^{(k)}+u_{t-1}^{(k)})+674(u_{t-m+1}^{(k)}+u_{m+t-1}^{(k)})+251(u_{t-m+2}^{(k)}+u_{2m+t-1}^{(k)}) \\ &+101(u_{m+t+4}^{(k)}+u_{4m+t+1}^{(k)})+88(u_{3m+t-1}^{(k)}+u_{4m+t}^{(k)}+u_{t+4}^{(k)}+u_{t-m+3}^{(k)}) \\ &+74(u_{2m+t+4}^{(k)}+u_{4m+t+2}^{(k)})+37(u_{3m+t+4}^{(k)}+u_{4m+t+3}^{(k)})] \;. \end{aligned}$$

By rotating the points of Figure (4.2.15a) a completely analogous expression can be obtained for the points $u_{t+3}^{(k+1)}, u_{3m+t+3}^{(k+1)}$ and $u_{3m+t}^{(k+1)}$.

In a similar manner, we have,

$$u_{t+1}^{(k+1)} = \frac{1}{6600}[2238u_{t-m+1}^{(k)}+762u_{t-m+2}^{(k)}+674(u_{t-m}^{(k)}+u_{t-1}^{(k)})+458u_{m+t-1}^{(k)}$$

$$+251(u_{t+4}^{(k)}+u_{t-m+3}^{(k)})+242(u_{2m+t-1}^{(k)}+u_{m+t+4}^{(k)})+162u_{4m+t+1}^{(k)}+158u_{2m+t+4}^{(k)}$$

$$+138u_{4m+t+2}^{(k)}+101(u_{3m+t-1}^{(k)}+u_{4m+t}^{(k)})+74(u_{3m+t+4}^{(k)}+u_{4m+t+3}^{(k)}) ,$$

while the mesh points, $u_{t+2}^{(k+1)}$, $u_{m+t}^{(k+1)}$, $u_{2m+t}^{(k+1)}$, $u_{3m+t+1}^{(k+1)}$, $u_{3m+t+2}^{(k+1)}$, $u_{m+t+3}^{(k+1)}$ and $u_{2m+t+3}^{(k+1)}$ are found by formulae similar to the one above. Also, we have,

$$u_{m+t+1}^{(k+1)} = \frac{1}{6600}[916(u_{m+t-1}^{(k)}+u_{t-m+1}^{(k)})+559(u_{t-m+2}^{(k)}+u_{2m+t-1}^{(k)})+$$

$$458(u_{t-m}^{(k)}+u_{t-1}^{(k)})+409(u_{m+t+4}^{(k)}+u_{4m+t+1}^{(k)})+316(u_{2m+t+4}^{(k)}+$$

$$u_{4m+t+2}^{(k)})+242(u_{3m+t-1}^{(k)}+u_{4m+t}^{(k)}+u_{t+4}^{(k)}+u_{t-m+3}^{(k)})+158(u_{3m+t+4}^{(k)}+u_{4m+t+3}^{(k)}), \tag{4.2.39}$$

and again by using similar formulae the points $u_{m+t+2}^{(k+1)}$, $u_{2m+t+1}^{(k+1)}$ and $u_{2m+t+2}^{(k+1)}$ can be obtained.

A similar set of equations can be written down for the 16-point group Gauss-Seidel, J.O.R. and S.O.R. methods.

4.2.7 The 25-Point Group

The final grouping of the mesh points which we are concerned with is by considering each group formed from the 25 points as shown in Figure (4.2.16).

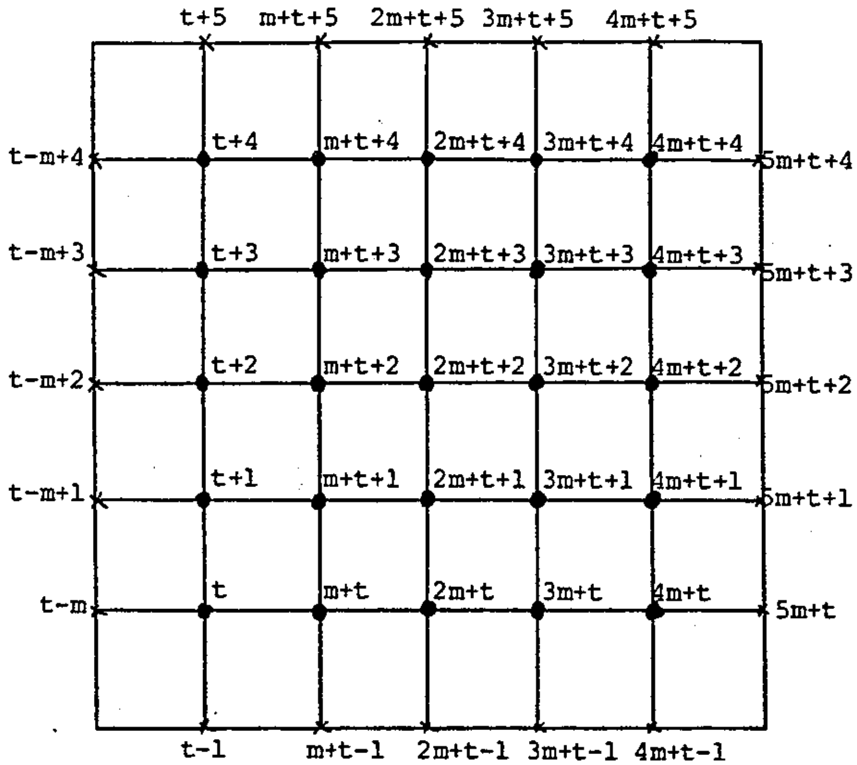
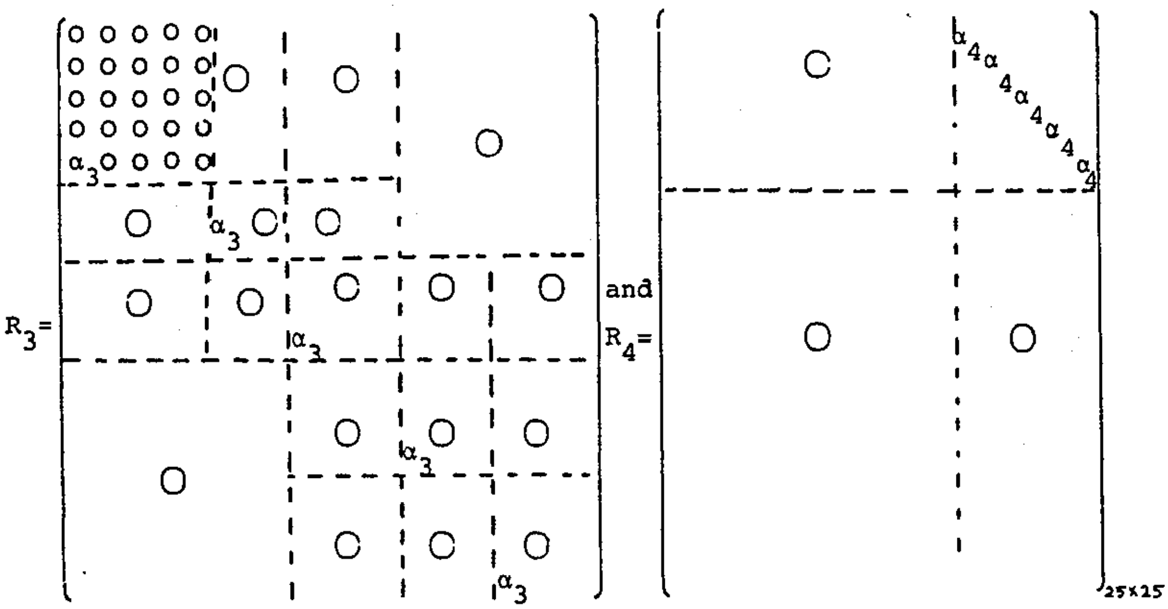


FIGURE (4.2.16)

where $t=rm+1,(5),(r+1)m-4$ and $r=0,(5),m-5$. Each group G_ℓ of Definition (3.11.1), $\ell=1,2,\ldots,\frac{m^2}{25}$, consists of 25 elements, and m must be divisible by 5. Consider the linear system (3.11.1) defined in a unit square with $\Delta x=\Delta y=1/11$, and the groups are ordered in red-black ordering, then by applying equation (4.2.1) to the mesh points in turn, the resulting coefficient matrix A has the same block structure as that given in (4.2.36a), where,

$$R_0=\left(\begin{array}{c|c|c|c|c}
\begin{matrix}1&\alpha_3&&&\\ \alpha_1&1&\alpha_3&&0\\ &\alpha_1&1&\alpha_3&\\ 0&&\alpha_1&1&\alpha_3\\ &&&\alpha_1&1\end{matrix} &
\begin{matrix}\alpha_2&&&&\\ &\alpha_2&&0&\\ &&\alpha_2&&\\ &0&&\alpha_2&\\ &&&&\alpha_2\end{matrix} & 0 & 0 & 0\\ \hline
\begin{matrix}\alpha_4&&&&\\ &\alpha_4&&0&\\ &&\alpha_4&&\\ 0&&&\alpha_4&\\ &&&&\alpha_4\end{matrix} & \ddots\;0 & 0\;\ddots & 0 & 0\\ \hline
0 & \ddots\;0 & \ddots\;0 & 0\;\ddots & 0\\ \hline
0 & 0 & \ddots\;0 & \ddots\;0 & \begin{matrix}\alpha_2&&&&\\ &\alpha_2&&&0\\ &&\alpha_2&&\\ &0&&\alpha_2&\\ &&&&\alpha_2\end{matrix}\\ \hline
0 & 0 & 0 & \begin{matrix}\alpha_4&&&&\\ &\alpha_4&&0&\\ &&\alpha_4&&\\ &0&&\alpha_4&\\ &&&&\alpha_4\end{matrix} & \begin{matrix}1&\alpha_3&&&\\ \alpha_1&1&\alpha_3&&0\\ &\alpha_1&1&\alpha_3&\\ &&\alpha_1&1&\alpha_3\\ &0&&\alpha_1&1\end{matrix}
\end{array}\right)_{25\times 25},$$

$$R_1=\left(\begin{array}{c|c|c|c|c}
\begin{matrix}0&0&0&0&\alpha_1\\ 0&0&0&0&0\;1\\ 0&0&0&0&0\\ 0&0&0&0&0\\ 0&0&0&0&0\end{matrix} & 0 & 0 & 0 & \\ \hline
0 & \begin{matrix}\alpha_1\\ 0\end{matrix} & 0 & & \\ \hline
0 & 0 & \begin{matrix}\alpha_1\\ 0\end{matrix} & 0 & 0\\ \hline
0 & & 0 & \begin{matrix}\alpha_1\\ 0\end{matrix} & 0\\ \hline
 & & 0 & 0 & \begin{matrix}\alpha_1\\ 0\end{matrix}
\end{array}\right)_{25\times 25},\quad
R_2=\left(\begin{array}{c|c}
0 & 0\\ \hline
\begin{matrix}\alpha_2\;\alpha_2\;\alpha_2\;\alpha_2\;\alpha_2 & 0\\ 0 & \end{matrix} & 0
\end{array}\right)_{25\times 25},$$



(4.2.40)

For the model problem (3.2.1), $\alpha_1=\alpha_2=\alpha_3=\alpha_4=-1/4$ and by applying REDUCE we can determine R_0^{-1} which is a symmetric matrix and is given by,

$$
R_0^{-1} = \frac{1}{102960}\left[\begin{array}{ccccccccccccccccccccccccc}
124268 \\
42616 & 140824 \\
16556 & 49560 & 143464 \\
6944 & 19196 & 49560 & 140824 \\
2640 & 6944 & 16556 & 42616 & 124260 \\
42616 & 29640 & 16664 & 8580 & 3616 & 140824 \\
29640 & 59280 & 38220 & 20280 & 8580 & 59280 & 169260 \\
16664 & 38220 & 62896 & 38220 & 16664 & 28436 & 73320 & 175144 \\
8580 & 20280 & 38220 & 59280 & 29640 & 14040 & 34320 & 73320 & 169260 \\
3616 & 8580 & 16664 & 29640 & 42616 & 5884 & 14040 & 28436 & 59280 & 140824 \\
16556 & 16664 & 11880 & 7096 & 3244 & 49560 & 38220 & 23760 & 13260 & 5880 & 143464 \\
16664 & 28436 & 23760 & 15124 & 7096 & 38220 & 73320 & 51480 & 29640 & 13260 & 62896 & 175144 \\
11880 & 23760 & 31680 & 23760 & 11880 & 23760 & 51480 & 7920 & 51480 & 23760 & 31680 & 7920 & 182160 \\
7096 & 15124 & 23760 & 28436 & 16664 & 13260 & 29640 & 51480 & 73320 & 38220 & 16304 & 38696 & 7920 & 175144 \\
3244 & 7096 & 11880 & 16664 & 16556 & 5880 & 13260 & 23760 & 38220 & 49560 & 7016 & 16304 & 31680 & 62896 & 143464 \\
6944 & 8580 & 7096 & 4680 & 2264 & 19196 & 20280 & 15124 & 9360 & 4376 & 49560 & 38220 & 23760 & 13260 & 5880 & 140824 \\
8580 & 14040 & 13260 & 9360 & 4680 & 20280 & 34320 & 29640 & 19500 & 9360 & 38220 & 73320 & 51480 & 29640 & 13260 & 59280 & 169260 \\
7096 & 13260 & 16304 & 13260 & 7096 & 15124 & 29640 & 38696 & 29640 & 15124 & 23760 & 51480 & 7920 & 51480 & 23760 & 28436 & 73320 & 175144 & & & & & \text{symmetric} \\
4680 & 9360 & 13260 & 14040 & 8580 & 9360 & 19500 & 29640 & 34320 & 20280 & 13260 & 29640 & 51480 & 73320 & 38220 & 14040 & 34320 & 73320 & 169260 \\
2264 & 4680 & 7096 & 8580 & 6944 & 4376 & 9360 & 15124 & 20280 & 19196 & 5880 & 13260 & 23760 & 38220 & 49560 & 5884 & 14040 & 28436 & 59280 & 140824 \\
2640 & 3616 & 3244 & 2264 & 1132 & 6944 & 8580 & 7096 & 4680 & 2264 & 16556 & 16664 & 11880 & 7096 & 3244 & 42616 & 29640 & 16664 & 8580 & 3616 & 124268 \\
3616 & 5884 & 5880 & 4376 & 2264 & 8580 & 14040 & 13260 & 9360 & 4680 & 16664 & 28436 & 23760 & 15124 & 7096 & 29640 & 59280 & 38220 & 20280 & 8580 & 42616 & 140824 \\
3244 & 5880 & 7016 & 5880 & 3244 & 7096 & 13260 & 16304 & 13260 & 7096 & 11880 & 23760 & 31680 & 23760 & 11880 & 16664 & 38220 & 62896 & 38220 & 16664 & 16556 & 49560 & 143464 \\
2264 & 4376 & 5880 & 5884 & 3616 & 4680 & 9360 & 13260 & 14040 & 8500 & 7096 & 15124 & 23760 & 28436 & 16664 & 8580 & 20280 & 38220 & 59280 & 29640 & 6944 & 19196 & 49560 & 140824 \\
1132 & 2264 & 3244 & 3616 & 2640 & 2264 & 4680 & 7096 & 8580 & 6944 & 3244 & 7096 & 11800 & 16664 & 16556 & 3616 & 8580 & 16664 & 29640 & 42616 & 2640 & 6944 & 16556 & 42616 & 124268
\end{array}\right]
$$

(4.2.41)

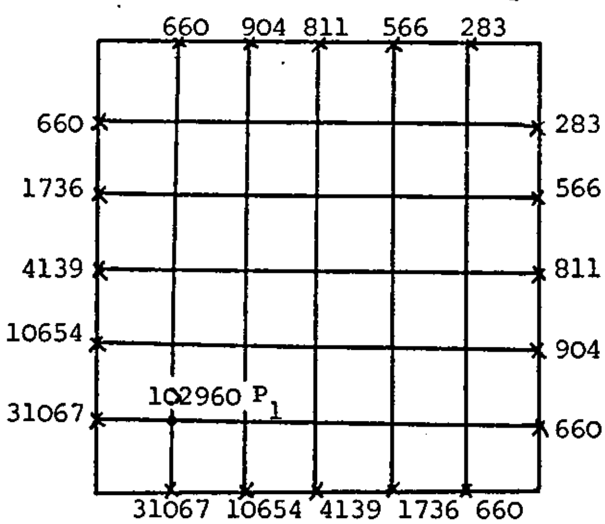
Hence,

$$
R_0^{-1}R_1 = -\frac{1}{102960}\left(\begin{array}{ccccc|cc|cc|cc|cc}
0&0&0&0&31067 & & 10654 & & 4139 & & 1736 & & 660\\
0&0&0&0&10654 & & 7410 & & 4166 & & 2145 & & 904\\
0&0&0&0&4139 & O & 4166 & O & 2970 & O & 1774 & O & 811\\
0&0&0&0&1736 & & 2145 & & 1774 & & 1170 & & 566\\
0&0&0&0&660 & & 904 & & 811 & & 566 & & 283\\
\hline
 & & & & 10654 & & 35206 & & 12390 & & 4799 & & 1736\\
 & & & & 7410 & & 14820 & & 9555 & & 5070 & & 2145\\
\multicolumn{4}{c}{O} & 4166 & O & 7109 & O & 5940 & O & 3781 & O & 1774\\
 & & & & 2145 & & 3510 & & 3315 & & 2340 & & 1170\\
 & & & & 904 & & 1471 & & 1470 & & 1094 & & 566\\
\hline
 & & & & 4139 & & 12390 & & 35866 & & 12390 & & 4139\\
 & & & & 4166 & & 9555 & & 15724 & & 9555 & & 4166\\
\multicolumn{4}{c}{O} & 2970 & O & 5940 & O & 7920 & O & 5940 & O & 2970\\
 & & & & 1774 & & 3315 & & 4076 & & 3315 & & 1774\\
 & & & & 811 & & 1470 & & 1754 & & 1470 & & 811\\
\hline
 & & & & 1736 & & 4799 & & 12390 & & 35181 & & 10654\\
 & & & & 2145 & & 5070 & & 9555 & & 14820 & & 7410\\
\multicolumn{4}{c}{O} & 1774 & O & 3781 & O & 5940 & O & 7109 & O & 4166\\
 & & & & 1170 & & 2340 & & 3315 & & 3510 & & 2145\\
 & & & & 566 & & 1094 & & 1470 & & 1471 & & 904\\
\hline
 & & & & 660 & & 1736 & & 4139 & & 10654 & & 31067\\
 & & & & 904 & & 2145 & & 4166 & & 7410 & & 10654\\
\multicolumn{4}{c}{O} & 811 & O & 1774 & O & 2970 & O & 4166 & O & 4139\\
 & & & & 566 & & 1170 & & 1774 & & 2145 & & 1736\\
 & & & & 283 & & 566 & & 811 & & 904 & & 660
\end{array}\right)
$$

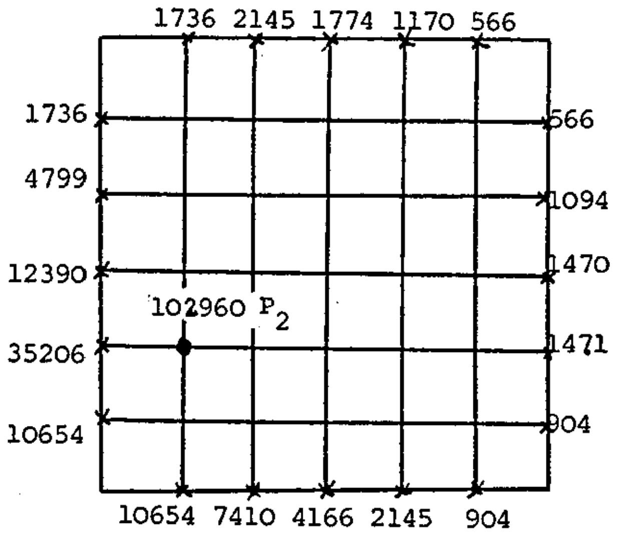
(4.2.42a)

$$R_0^{-1}R_2 = -\frac{1}{102960}\left[\begin{array}{ccccc|c}
660 & 904 & 811 & 566 & 283 & \\
904 & 1471 & 1470 & 1094 & 566 & \\
811 & 1470 & 1754 & 1470 & 811 & O \\
566 & 1094 & 1470 & 1471 & 904 & \\
283 & 566 & 811 & 904 & 660 & \\
\hline
1736 & 2145 & 1774 & 1170 & 566 & \\
2145 & 3510 & 3315 & 2340 & 1170 & \\
1774 & 3315 & 4076 & 3315 & 1774 & O \\
1170 & 2340 & 3315 & 3510 & 2145 & \\
566 & 1170 & 1774 & 2145 & 1736 & \\
\hline
4139 & 4166 & 2970 & 1774 & 811 & \\
4166 & 7109 & 5940 & 3781 & 1774 & \\
2970 & 5940 & 7920 & 5940 & 2970 & O \\
1774 & 3781 & 5940 & 7109 & 4166 & \\
811 & 1774 & 2970 & 4166 & 4139 & \\
\hline
10654 & 7410 & 4166 & 2145 & 904 & \\
7410 & 14820 & 9555 & 5070 & 2145 & \\
4166 & 9555 & 15724 & 9555 & 4166 & O \\
2145 & 5070 & 9555 & 14820 & 7410 & \\
904 & 2145 & 4166 & 7410 & 10654 & \\
\hline
31067 & 10654 & 4139 & 1736 & 660 & \\
10654 & 35206 & 12390 & 4799 & 1736 & \\
4139 & 12390 & 35866 & 12390 & 4139 & O \\
1736 & 4799 & 12390 & 35206 & 10654 & \\
660 & 1736 & 4139 & 10654 & 31067 &
\end{array}\right]_{25\times 25}, \quad (4.2.42b)$$

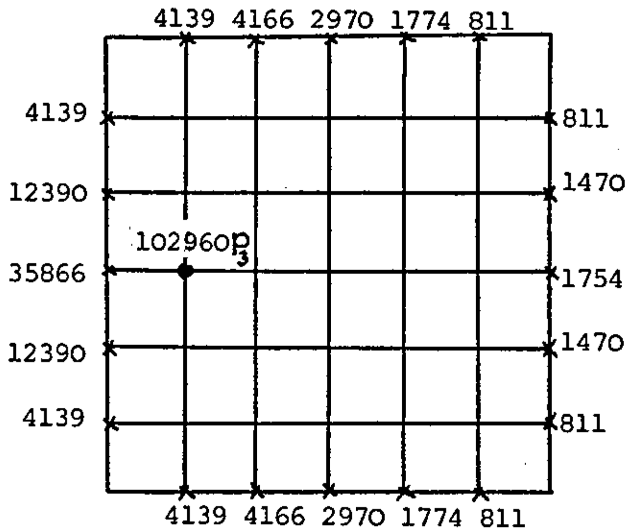
similarly, $R_0^{-1}R_3$ and $R_0^{-1}R_4$ can be determined, which enable us to establish the computational molecule at the points P_1, P_2, P_3, P_4, P_5 and P_6 to be,



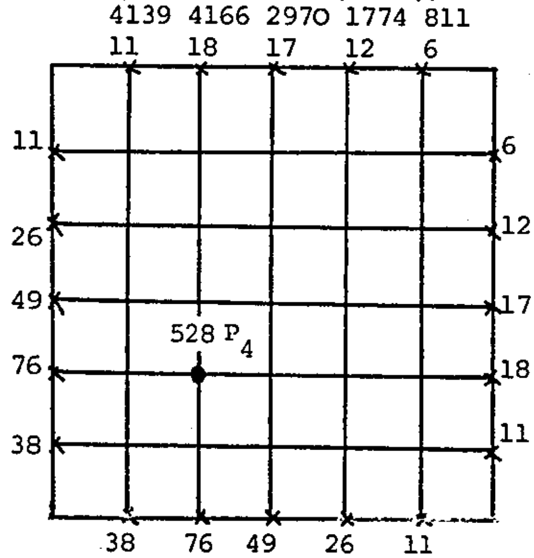
(a)


1736 2145 1774 1170 566
1736
4799
12390
35206
10654
566
1094
1470
1471
904
102960 P_2
10654 7410 4166 2145 904

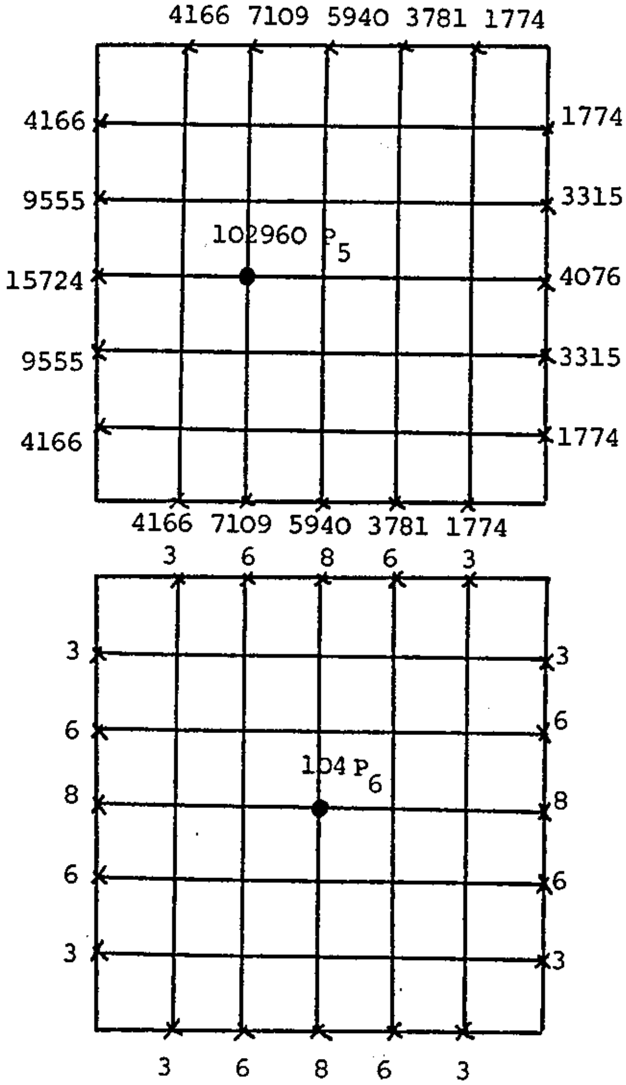
(b)


4139 4166 2970 1774 811
4139
12390
35866
12390
4139
811
1470
1754
1470
811
102960 P_3
4139 4166 2970 1774 811

(c)


11 18 17 12 6
11
26
49
76
38
6
12
17
18
11
528 P_4
38 76 49 26 11

(d)



(e)

(f)

FIGURE (4.2.17)

Hence by using these new computational molecules, an explicit group Gauss-Seidel iterative method can be derived and is given by,

$$u_t^{(k+1)} = \frac{1}{102960}[31067(u_{t-m}^{(k+1)}+u_{t-1}^{(k+1)})+10654(u_{t-m+1}^{(k+1)}+u_{m+t-1}^{(k+1)})+4139(u_{t-m+2}^{(k+1)}+ u_{2m+t-1}^{(k+1)})+1736(u_{t-m+3}^{(k+1)}+u_{3m+t-1}^{(k+1)})+904(u_{m+t+5}^{(k)}+u_{5m+t+1}^{(k)})+ 811(u_{2m+t+5}^{(k)}+u_{5m+t+2}^{(k)})+660(u_{4m+t-1}^{(k+1)}+u_{5m+t}^{(k)}+u_{t+5}^{(k)}+u_{t-m+4}^{(k+1)})+ 566(u_{3m+t+5}^{(k)}+u_{5m+t+3}^{(k)})+283(u_{4m+t+5}^{(k)}+u_{5m+t+4}^{(k)})],$$

$u_{t+4}^{(k+1)}, u_{4m+t+4}^{(k+1)}$ and $u_{4m+t}^{(k+1)}$ are found from completely analogous expressions obtained by rotating the mesh points clockwise of Figure (4.2.17a).

$$u_{t+1}^{(k+1)} = \frac{1}{102960}[35206u_{t-m+1}^{(k+1)}+12390u_{t-m+2}^{(k+1)}+10654(u_{t-m}^{(k+1)}+u_{t-1}^{(k+1)})+7410u_{m+t-1}^{(k+1)}$$
$$+4799u_{t-m+3}^{(k+1)}+4166u_{2m+t-1}^{(k+1)}+2145(u_{3m+t-1}^{(k+1)}+u_{m+t+5}^{(k)})+1774u_{2m+t+5}^{(k)}$$
$$+1736(u_{t-m+4}^{(k+1)}+u_{t+5}^{(k)})+1471u_{5m+t+1}^{(k)}+1470u_{5m+t+2}^{(k)}+1170u_{3m+t+5}^{(k)}+$$
$$1094u_{5m+t+3}^{(k)}+904(u_{4m+t-1}^{(k+1)}+u_{5m+t}^{(k)})+566(u_{4m+t+5}^{(k)}+u_{5m+t+4}^{(k)})],$$

the mesh points $u_{t+3}^{(k+1)}, u_{m+t}^{(k+1)}, u_{m+t+4}^{(k+1)}, u_{3m+t}^{(k+1)}, u_{3m+t+4}^{(k+1)}, u_{4m+t+1}^{(k+1)}$ and $u_{4m+t+3}^{(k+1)}$ are obtained by using formulae similar to the one above.

$$u_{t+2}^{(k+1)} = \frac{1}{102960}[35866u_{t-m+2}^{(k+1)}+12390(u_{t-m+1}^{(k+1)}+u_{t-m+3}^{(k+1)})+4166(u_{m+t-1}^{(k+1)}+u_{m+t+5}^{(k)})+$$
$$4139(u_{t-m}^{(k+1)}+u_{t-1}^{(k+1)}+u_{t-m+4}^{(k+1)}+u_{t+5}^{(k)})+2970(u_{2m+t-1}^{(k+1)}+u_{2m+t+5}^{(k)})+$$
$$1774(u_{3m+t-1}^{(k+1)}+u_{3m+t+5}^{(k)})+1754u_{5m+t+2}^{(k)}+1470(u_{5m+t+1}^{(k)}+u_{5m+t+3}^{(k)})+$$
$$811(u_{4m+t-1}^{(k+1)}+u_{5m+t}^{(k)}+u_{4m+t+5}^{(k)}+u_{5m+t+4}^{(k)})],$$

by using similar formulae, the mesh points $u_{2m+t}^{(k+1)}, u_{2m+t+4}^{(k+1)}$ and $u_{4m+t+2}^{(k+1)}$ are obtained.

$$u_{m+t+1}^{(k+1)} = \frac{1}{528}[76(u_{t-m+1}^{(k+1)}+u_{m+t-1}^{(k+1)})+49(u_{t-m+2}^{(k+1)}+u_{2m+t-1}^{(k+1)})+38(u_{t-m}^{(k+1)}+u_{t-1}^{(k+1)})$$
$$+26(u_{t-m+3}^{(k+1)}+u_{3m+t-1}^{(k+1)})+18(u_{m+t+5}^{(k)}+u_{5m+t+1}^{(k)})+17(u_{2m+t+5}^{(k)}+u_{5m+t+2}^{(k)})+$$
$$12(u_{3m+t+5}^{(k)}+u_{5m+t+3}^{(k)})+11(u_{4m+t-1}^{(k+1)}+u_{5m+t}^{(k)}+u_{t+5}^{(k)}+u_{t-m+4}^{(k+1)})+$$
$$6(u_{4m+t+5}^{(k)}+u_{5m+t+4}^{(k)})],$$

the mesh points $u_{m+t+3}^{(k+1)}, u_{3m+t+3}^{(k+1)}$ and $u_{3m+t+1}^{(k+1)}$ are found from completely analogous expressions obtained by rotating the mesh points clockwise of Figure (4.2.17d).

$$u_{m+t+2}^{(k+1)} = \frac{1}{102960}[15724u_{t-m+2}^{(k+1)}+9555(u_{t-m+1}^{(k+1)}+u_{t-m+3}^{(k+1)})+7109(u_{m+t-1}^{(k+1)}+ u_{m+t+5}^{(k)})+5940(u_{2m+t-1}^{(k+1)}+u_{2m+t+5}^{(k)})+4166(u_{t-m}^{(k+1)}+u_{t-1}^{(k+1)}+u_{t-m+4}^{(k+1)} +u_{t+5}^{(k)})+4076u_{5m+t+2}^{(k)}+3781(u_{3m+t-1}^{(k+1)}+u_{3m+t+5}^{(k)})+3315(u_{5m+t+1}^{(k)}+ u_{5m+t+3}^{(k)})+1774(u_{4m+t-1}^{(k+1)}+u_{5m+t}^{(k)}+u_{4m+t+5}^{(k)}+u_{5m+t+4}^{(k)})] ,$$

by using a similar formulae again, the mesh points $u_{2m+t+1}^{(k+1)}$, $u_{2m+t+3}^{(k+1)}$ and $u_{3m+t+2}^{(k+1)}$ are obtained. Finally,

$$u_{2m+t+2}^{(k+1)} = \frac{1}{104}[8(u_{t-m+2}^{(k+1)}+u_{2m+t-1}^{(k+1)}+u_{2m+t+5}^{(k)}+u_{5m+t+2}^{(k)})+6(u_{t-m+3}^{(k+1)}+ u_{t-m+1}^{(k+1)}+u_{m+t-1}^{(k+1)}+u_{3m+t-1}^{(k+1)}+u_{5m+t+1}^{(k)}+u_{5m+t+3}^{(k)}+u_{3m+t+5}^{(k)}+u_{m+t+5}^{(k)})+ +3(u_{t-m}^{(k+1)}+u_{t-1}^{(k+1)}+u_{4m+t-1}^{(k+1)}+u_{5m+t}^{(k)}+u_{5m+t+4}^{(k)}+u_{4m+t+5}^{(k)}+u_{t+5}^{(k)}+u_{t-m+4}^{(k+1)})] , \tag{4.2.43}$$

A similar set of equations can be written down for the 25-point Jacobi, J.O.R. and S.O.R. iterative methods.

4.3 EXPERIMENTAL RESULTS FOR THE GROUP S.O.R. METHODS

In order to choose the most efficient group of points which were described in Section (4.2), and to compare them with the standard point, line and two line S.O.R. methods, we present here some numerical experiments. All the results were obtained on the "ICL 1904S computer" at Loughborough University.

Problem

The problem is that of solving Laplace's equation,

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} = 0 \ , \quad (x,y) \in R, \tag{4.3.1}$$

and

$$\left.\begin{aligned} U(0,y) &= 100 \ , && 0 \leq y \leq 1 \\ U(x,0) = U(1,y) = U(x,1) &= 0 \ , && 0 \leq x,y \leq 1 \end{aligned}\right\} \tag{4.3.2}$$

where R is the unit square.

A red-black ordering of the points of the mesh and also of the groups has been used. Therefore, the resulting coefficient matrices possess Property $A^{(\pi)}$ and are π-consistently ordered so that the theory of block S.O.R. is valid. Thus, the optimum relaxation factor, ω_b, can be calculated using the formula,

$$\omega_b = \frac{2}{1+\sqrt{1-[\rho(J_p)]^2}} \ . \tag{4.3.3}$$

where $\rho(J_p)$ is the spectral radius of the $P\times P$ group Jacobi scheme and can be estimated by the following formula, [Parter,(1981)],

$$\rho(J_p) \approx 1 - \frac{P}{2}\pi^2 h^2 \ . \tag{4.3.4}$$

The convergence test used in the numerical experiments was the average test, i.e. the iteration was continued until the following condition was satisfied,

$$\frac{\left|\left|u_{i,j}^{(k+1)}-u_{i,j}^{(k)}\right|\right|}{1+\left|\left|u_{i,j}^{(k)}\right|\right|} < \varepsilon \qquad \text{for all } i,j \;, \tag{4.3.5}$$

where $\varepsilon=10^{-7}$. The number of iterations can be obtained by combining equations (3.7.9) and (3.9.5) to give,

$$k \simeq \frac{\ln\varepsilon}{\ln(\omega_b-1)} \quad . \tag{4.3.6}$$

The experimental optimum values of ω are determined to within ± 0.001 by solving the problem for a range of values of ω and choosing those which give the minimum number of iterations. Different mesh sizes were used to produce the (14×14),(26×26),(38×38),(50×50) and (62×62) networks whilst for the 25 point group the mesh sizes used were the closely associated (12×12),(27×27),(37×37),(52×52) and (62×62) networks.

The spectral radius of the point Jacobi iteration matrix is given by (3.9.13) and for the line and 2-line Jacobi scheme the formula is given in (3.11.43).

The results obtained from these experiments are recorded in Tables (4.3.1), (4.3.2),...,(4.3.10). The execution times of the different schemes are recorded in Table (4.3.11). Also for each method, the logarithm of the minimum number of iterations was plotted against $\log(h^{-1})$, where h the mesh size, the graphs are shown in Figure (4.3.1). As expected, the plots for the methods were straight lines with a slope of unity thus verifying the S.O.R. theory.

$\varepsilon = 10^{-7}$

h^{-1}	$\rho(J_p)$	Optimum Relaxation Factor (ω_b)		Number of Iterations (k)	
		Theory	Experiment	Theory	Experiment
13	0.97094	1.614	1.61-1.616	33	39
25	0.992115	1.777	1.776-1.777	64	73
37	0.9964	1.844	1.842	95	108
49	0.99795	1.879	1.877	126	143
61	0.99867	1.902	1.9	156	179

TABLE (4.3.1): Point S.O.R.

h^{-1}	$\rho(J_p)$	ω_b		k	
		Theory	Experiment	Theory	Experiment
13	0.9416	1.496	1.509-1.514	24	25
25	0.98421	1.6992	1.704-1.71	47	47
37	0.99279	1.786	1.789-1.791	68	68
49	0.99589	1.834	1.836-1.837	90	88
61	0.99735	1.864	1.866	112	109

TABLE (4.3.2): 1-line S.O.R.

h^{-1}	$\rho(J_p)$	ω_b		k	
		Theory	Experiment	Theory	Experiment
13	0.8832	1.361	1.384-1.386	17	18
25	0.96842	1.601	1.608-1.611	33	34
37	0.98558	1.711	1.714-1.716	48	50
49	0.99178	1.773	1.774-1.777	63	66
61	0.99469	1.813	1.814	78	81

TABLE (4.3.3): 2-line S.O.R.

h^{-1}	$\rho(J_p)$	ω_b		k	
		Theory	Experiment	Theory	Experiment
13	0.958705	1.557	1.571-1.576	30	33
25	0.98883	1.741	1.751	56	58
37	0.99490	1.817	1.824-1.825	84	86
49	0.99709	1.859	1.864-1.865	111	115
61	0.99812	1.885	1.889-1.89	138	141

TABLE (4.3.4): 2-Point Group S.O.R.

h^{-1}	$\rho(J_p)$	ω_b		k	
		Theory	Experiment	Theory	Experiment
13	0.9416	1.496	1.508	24	25
25	0.98421	1.6992	1.705	47	48
37	0.99279	1.786	1.789	68	70
49	0.99589	1.834	1.836-1.837	90	94
61	0.99735	1.864	1.866-1.867	112	116

TABLE (4.3.5): 4-Point Group S.O.R.

h^{-1}	$\rho(J_p)$	ω_b		k	
		Theory	Experiment	Theory	Experiment
13	0.92847	1.458	1.474-1.476	22	24
25	0.98066	1.673	1.681-1.682	42	44
37	0.99117	1.766	1.771-1.772	62	65
49	0.99497	1.818	1.822-1.823	82	85
61	0.99675	1.851	1.855	103	106

TABLE (4.3.6): 6-Point Group S.O.R.

h^{-1}	$\rho(J_p)$	ω_b		k	
		Theory	Experiment	Theory	Experiment
13	0.9124	1.419	1.436-1.438	19	21
25	0.97631	1.644	1.651	38	39
37	0.98919	1.744	1.749	56	59
49	0.99383	1.80	1.804	74	78
61	0.99602	1.836	1.839	92	93

TABLE (4.3.7): 9-Point Group S.O.R.

h^{-1}	$\rho(J_p)$	ω_b		k	
		Theory	Experiment	Theory	Experiment
13	0.89885	1.391	1.41—1.416	18	21
25	0.97265	1.623	1.632-1.634	35	38
37	0.98751	1.728	1.732-1.733	52	56
49	0.99288	1.787	1.791-1.792	69	73
61	0.99541	1.825	1.829	86	89

TABLE (4.3.8): 12-Point Group S.O.R.

h^{-1}	$\rho(J_p)$	ω_b		k	
		Theory	Experiment	Theory	Experiment
13	0.8832	1.361	1.388-1.389	17	19
25	0.96842	1.601	1.61-1.611	33	35
37	0.98558	1.711	1.716	48	50
49	0.99178	1.773	1.776-1.777	64	67
61	0.99469	1.813	1.816	79	83

TABLE (4.3.9): 16-Point Group S.O.R.

h^{-1}	$\rho(J_p)$	ω_b		k	
		Theory	Experiment	Theory	Experiment
11	0.79608	1.246	1.293-1.295	13	15
26	0.9635	1.578	1.589-1.591	31	33
36	0.98096	1.675	1.681	42	45
51	0.99051	1.758	1.762-1.763	59	63
61	0.99337	1.794	1.797-1.798	71	75

TABLE (4.3.10): 25-Point Group S.O.R.

Method / h^{-1}	Point S.O.R.	1-line S.O.R.	2-line S.O.R.	2-Point Group S.O.R.	4-Point Group S.O.R.	6-Point Group S.O.R.	9-Point Group S.O.R.	12-Point Group S.O.R.	16-Point Group S.O.R.	Method / h^{-1}	25-Point Group S.O.R.
13	4	4	2	4	3	3	2	2	2	11	1
25	31	33	18	29	21	21	14	18	14	26	17
37	105	106	60	96	68	68	47	57	45	36	45
49	246	243	140	229	163	158	110	140	128	51	128
61	483	470	267	435	314	308	203	265	248	61	220

TABLE (4.3.11): The execution time in mill units for the optimum value of ω

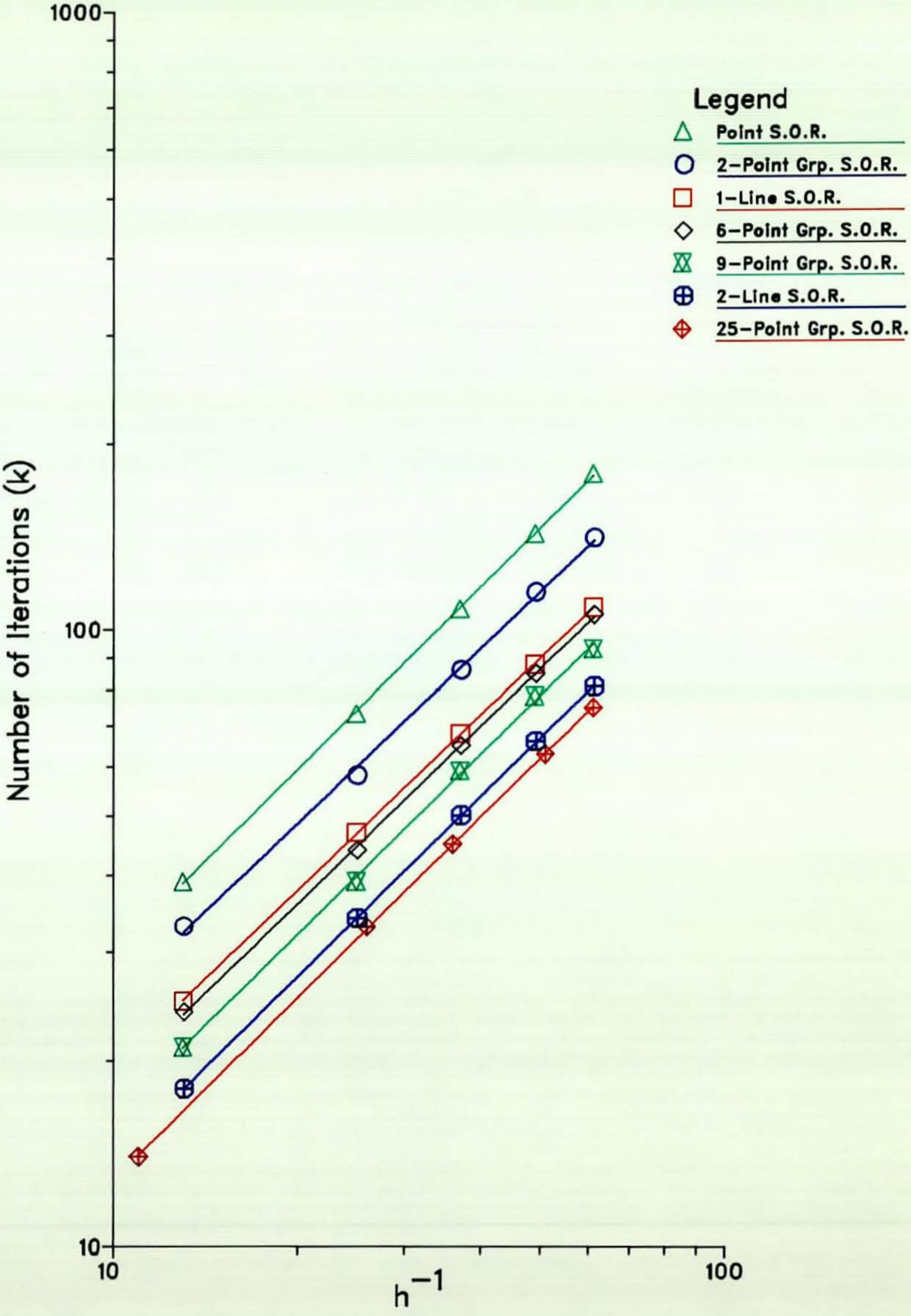

1000
100
10
10
100
Number of Iterations (k)
h⁻¹
Legend
Point S.O.R.
2-Point Grp. S.O.R.
1-Line S.O.R.
6-Point Grp. S.O.R.
9-Point Grp. S.O.R.
2-Line S.O.R.
25-Point Grp. S.O.R.

Figure (4.3.1)

From Tables (4.3.2) and (4.3.5) it can be noted that the experimental optimum values of ω, ω_b, for the 1-line S.O.R. and the 4-point group S.O.R. methods agree to two decimal places, the same can be said about ω_b of the 16-point group and the 2-line S.O.R. methods. This can be proved by considering the model problem in the unit square and for simplicity, take $h^{-1}=5$. First, $\rho(J_p)$, the spectral radius for the block Jacobi scheme, has to be determined, then ω_b can be determined from equation (4.3.3).

(i) The 4-Point Group

We now order the mesh points in groups of four points and then the groups are ordered in a red-black ordering fashion as shown in Figure (4.3.2).

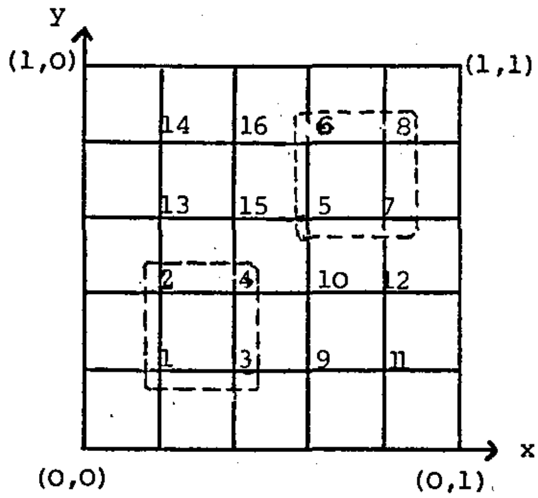


FIGURE (4.3.2)

By applying the 5-point finite difference formula to each point of the region, the resulting coefficient matrix A, has the block structure:

$$A = \left[\begin{array}{cc|cc} R_0 & 0 & R_1 & R_2 \\ 0 & R_0 & R_3 & R_4 \\ \hline R_4 & R_2 & R_0 & 0 \\ R_3 & R_1 & 0 & R_0 \end{array}\right], \qquad (4.3.7a)$$

where,

$$R_0 = \begin{pmatrix} 4 & -1 & -1 & 0 \\ -1 & 4 & 0 & -1 \\ -1 & 0 & 4 & -1 \\ 0 & -1 & -1 & 4 \end{pmatrix}, \quad R_1 = \left(\begin{array}{cc|cc} & \bigcirc & & \bigcirc \\ & & & \\ \hline -1 & 0 & & \bigcirc \\ 0 & -1 & & \end{array}\right), \quad R_2 = \left(\begin{array}{cc|cc} 0 & 0 & & \bigcirc \\ -1 & 0 & & \\ \hline & \bigcirc & 0 & 0 \\ & & -1 & 0 \end{array}\right)$$

$$R_3 = \left(\begin{array}{cc|cc} 0 & -1 & & \bigcirc \\ 0 & 0 & & \\ \hline & \bigcirc & 0 & -1 \\ & & 0 & 0 \end{array}\right), \text{ and } R_4 = \left(\begin{array}{cc|cc} & \bigcirc & -1 & 0 \\ & & 0 & -1 \\ \hline & \bigcirc & & \bigcirc \\ & & & \end{array}\right) . \tag{4.3.7b}$$

Now, from equation (3.11.9), the 4-point group Jacobi matrix, J_{4b}, (say), can be written as:

$$J_{4b} = D^{-1}(C+C^T) , \tag{4.3.8a}$$

where,

$$D = \begin{pmatrix} R_0 & & & \\ & R_0 & & \bigcirc \\ & & R_0 & \\ \bigcirc & & & R_0 \end{pmatrix} \text{ and } C = \left(\begin{array}{c|cc} \bigcirc & R_1 & R_2 \\ & R_3 & R_4 \\ \hline \bigcirc & & \bigcirc \end{array}\right) . \tag{4.3.8b}$$

D^{-1} can be found simply and hence,

$$J_{4b} = \left(\begin{array}{cc|cc} & \bigcirc & S_1 & S_2 \\ & & S_3 & S_4 \\ \hline S_4 & S_2 & & \bigcirc \\ S_3 & S_1 & & \end{array}\right) , \tag{4.3.9a}$$

where,

$$S_1 = \left(\begin{array}{cc|cc} \alpha_2 & \alpha_1 & 0 & 0 \\ \alpha_1 & \alpha_2 & 0 & 0 \\ \hline \alpha_3 & \alpha_2 & 0 & 0 \\ \alpha_2 & \alpha_3 & 0 & 0 \end{array}\right), \quad S_2 = \left(\begin{array}{cc|cc} \alpha_2 & 0 & \alpha_1 & 0 \\ \alpha_3 & 0 & \alpha_2 & 0 \\ \hline \alpha_1 & 0 & \alpha_2 & 0 \\ \alpha_2 & 0 & \alpha_3 & 0 \end{array}\right),$$

$$S_3 = \left[\begin{array}{cc|cc} 0 & \alpha_3 & 0 & \alpha_2 \\ 0 & \alpha_2 & 0 & \alpha_1 \\ \hline 0 & \alpha_2 & 0 & \alpha_3 \\ 0 & \alpha_1 & 0 & \alpha_2 \end{array}\right] , \quad \text{and } S_4 = \left[\begin{array}{cc|cc} 0 & 0 & \alpha_3 & \alpha_2 \\ 0 & 0 & \alpha_2 & \alpha_3 \\ \hline 0 & 0 & \alpha_2 & \alpha_1 \\ 0 & 0 & \alpha_1 & \alpha_2 \end{array}\right] \qquad (4.3.9b)$$

and, $\alpha_1 = -\frac{1}{24}$, $\alpha_2 = -\frac{2}{24}$ and $\alpha_3 = -\frac{7}{24}$.

Hence, the spectral radius can be numerically calculated and is given by,

$$\rho(J_{4b}) \simeq 0.6667$$

$$\simeq 0.67 \text{ approximated to 2 decimal places.}$$

From equation (4.3.3),

$$\omega_b = \frac{2}{1+\sqrt{1-(0.67)^2}}$$

$$= 1.14786$$

$$\simeq 1.15 \text{ approximated to 2 decimal places.}$$

(ii) <u>The 1-line iterative method</u>

For the model problem, and from equation (3.11.38),

$$\rho(J_{1\ line}) = \frac{\cos\pi h}{2-\cos\pi h} \qquad (4.3.10)$$

$$= \frac{\cos\ \pi/5}{2-\cos\ \pi/5}$$

$$\simeq 0.679$$

Again, from equation (4.3.3)

$$\omega_b = \frac{2}{1+\sqrt{1-(0.679)^2}}$$

$$\simeq 1.153$$

$$\simeq 1.15 \text{ approximated to 2 decimal places.}$$

Hence, the optimum values of ω for the two methods agree to 2 decimal places.

Similarly, it can be shown that,

$$\omega_b^{(16\text{-point})} = \omega_b^{(2\text{-line})}$$

approximated to 2-decimal places.

4.4 THE COMPUTATIONAL COMPLEXITY

We now consider the relative efficiency of the various iterative methods which we have discussed for solving the model problem (4.3.1). It is necessary to measure the computational complexity for each iteration and the number of iterations needed to obtain a (sufficiently accurate) solution in order to estimate the amount of computational work to obtain the solution. We shall estimate this computational complexity from the point of view of arithmetic operations performed per iteration.

In each method we assume that there are m^2 internal mesh points.

The Point S.O.R. Method

The 5-point finite difference solution of the model problem (4.3.1) by the point S.O.R. method is given by,

$$u_{i,j}^{(k+1)} = \omega(\tilde{u}_{i,j}^{(k+1)} - u_{i,j}^{(k)}) + u_{i,j}^{(k)}\ , \qquad (4.4.1)$$

where the $\tilde{u}_{i,j}^{(k+1)}$ are the components of the (k+1)th Gauss-Seidel iteration and is defined by,

$$\tilde{u}_{i,j}^{(k+1)} = (u_{i+1,j}^{(k)} + u_{i-1,j}^{(k+1)} + u_{i,j+1}^{(k)} + u_{i,j-1}^{(k+1)}) * 0.25,\ \text{for all } i,j. \qquad (4.4.2)$$

Hence, the number of operations required (excluding the convergent test) is,

$$2m^2 \text{ multiplications} + 5m^2 \text{ additions}, \qquad (4.4.3)$$

per iteration.

The 1-Line S.O.R. Method

In Chapter 3, an efficient algorithm based on the Gauss elimination method was defined, (see equation (3.11.34)). Thus, to implement this algorithm to the model problem for the solution of the 1-line, we have,

$$a_i = -1, \qquad i=1,2,\ldots,m-1,$$

$$c_i = -1, \qquad i=2,3,\ldots,m \qquad (4.4.4)$$

and

$$b_i = 4, \qquad i=1,2,\ldots,m\ .$$

The Gauss-Seidel form of equation (3.11.34c) can be written as,

$$\tilde{u}^{(k+1)}_{i,m} = r^*_m\ ,$$

and

$$\tilde{u}^{(k+1)}_{i,j} = r^*_j - c^*_j \tilde{u}^{(k+1)}_{i,j+1}\ , \quad j=m-1,(-1),1. \qquad (4.4.5)$$

The (k+1)th iterates of the ith row are now modified by the extrapolation equation,

$$u^{(k+1)}_{i,j} = \omega(\tilde{u}^{(k+1)}_{i,j} - u^{(k)}_{i,j}) + u^{(k)}_{i,j}\ , \quad j=1,2,\ldots,m \qquad (4.4.6)$$

Notice that the c_i $(i=1,2,\ldots,m)$, need only be calculated once, since they remain constant for each system of equations and for each iteration for the model problem. Thus the average work per iteration for m^2 internal mesh points is,

$$m(3m-1) \text{ multiplications} + 5m^2 \text{ additions} \qquad (4.4.7)$$

with an additional m multiplications + (m-1) additions before the first iteration to evaluate the c_i $(i=1,2,\ldots,m)$.

The 2-Line S.O.R. Method

An algorithm based on the solution of a quindiagonal matrix was defined in Chapter 3, (see equation (3.11.37) and for the model problem,

$$a_i = -1, \quad i=3,4,\ldots,m,$$

$$b_i = -1, \quad i=2,(2),m,$$

$$d_i = -1, \quad i=1,(2),m-1, \qquad (4.4.8)$$

$$e_i = -1, \quad i=1,2,\ldots,m-2$$

and

$$c_i = 4, \quad i=1,2,\ldots,m\ .$$

The Gauss-Seidel form of equation (3.11.37d) is of the form,

$$\tilde{u}_{i,m}^{(k+1)} = t_m^* ,$$

$$\tilde{u}_{i,m-1}^{(k+1)} = t_{m-1}^* - d_{m-1}^* \tilde{u}_m^{(k+1)} , \tag{4.4.9}$$

and

$$\tilde{u}_{i,j}^{(k+1)} = t_j^* - d_j^* u_{i,j+1} - e_j^* u_{i,j+2}, \quad j=m-2,(-1),1.$$

Now we modify the calculated points by applying the extrapolation equation,

$$u_{i,j}^{(k+1)} = \omega(\tilde{u}_{i,j}^{(k+1)} - u_{i,j}^{(k)}) + u_{i,j}^{(k)} , \quad i,j=1,2,\ldots,m. \tag{4.4.10}$$

Again, it is only necessary to calculate d_i, e_i and c_i, $(i=1,\ldots,m)$ once. Therefore, the total work for m^2 internal mesh points per iteration is,

$$\frac{m}{2}(10m-5)\text{multiplications} + m(6m-1)\text{additions}, \tag{4.4.11}$$

with an additional $(8m-5)$ multiplications + $(6m-5)$ additions for the first iteration to evaluate the values of d_i, e_i and c_i, $(i=1,\ldots,m)$.

<u>The 2-Point Group S.O.R. Method</u>

To calculate the number of operations per iteration using this grouping technique for the solution of the model problem (4.3.1), it can be seen, from equation (4.2.12), that the Gauss-Seidel iterative method involves solving $\frac{m^2}{2}$ systems of equations of the form,

$$\begin{pmatrix} 4 & -1 \\ -1 & 4 \end{pmatrix} \begin{pmatrix} \tilde{u}_{i,j}^{(k+1)} \\ \tilde{u}_{i+1,j}^{(k+1)} \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} \quad \begin{matrix} i=1,(2),m-1, \\ j=1,m, \end{matrix} \tag{4.4.12a}$$

where,

$$b_1 = u_{i,j-1}^{(k+1)} + u_{i-1,j}^{(k+1)} + u_{i,j+1}^{(k)} ,$$

$$\tag{4.4.12b}$$

and

$$b_2 = u_{i+1,j+1}^{(k)} + u_{i+2,j}^{(k)} + u_{i+1,j-1}^{(k+1)} .$$

The (k+1)th iterate of the S.O.R. iterative scheme is defined by,

$$u_{i,j}^{(k+1)} = \omega(\tilde{u}_{i,j}^{(k+1)} - u_{i,j}^{(k)}) + u_{i,j}^{(k)} \quad , \; i=1,(2),m-1, \tag{4.4.13}$$

and

$$u_{i+1,j}^{(k+1)} = \omega(\tilde{u}_{i+1,j}^{(k+1)} - u_{i+1,j}^{(k)}) + u_{i+1,j}^{(k)} \quad j=1,m,$$

where again $\tilde{u}^{(k+1)}$ represents the components of the (k+1)th Gauss-Seidel iteration defined by the formulae,

$$\left.\begin{aligned} \tilde{u}^{(k+1)}_{i,j} &= \frac{1}{15}(4b_1 + b_2)\ , \\ \text{and} \qquad \tilde{u}^{(k+1)}_{i+1,j} &= \frac{1}{15}(4b_2 + b_1)\ , \end{aligned}\right\} \quad i=1,(2),m-1,\ j=1,m, \tag{4.4.14}$$

so, this process requires,

$$3m^2 \text{ multiplications} + 5m^2 \text{ additions}\ , \tag{4.4.15}$$

for m^2 internal mesh points per iteration, assuming that the constant $\frac{1}{15}$ is stored beforehand.

Further savings in work can be made by solving $\tilde{u}^{(k+1)}_{i+1,j}$, using the value of $\tilde{u}^{(k+1)}_{i,j}$. Hence, instead of equation (4.4.14), we have,

$$\tilde{u}^{(k+1)}_{i+1,j} = 0.25(\tilde{u}^{(k+1)}_{i,j} + b_2)\ . \tag{4.4.16}$$

Therefore, the number of operations required for m^2 internal mesh points including the over-relaxation process, is,

$$\frac{5}{2}m^2 \text{ multiplications} + 5m^2 \text{ additions.} \tag{4.4.17}$$

<u>The 4-Point Group S.O.R. Method</u>

For this grouping, the 4-point Gauss-Seidel iterative method involves solving $\frac{m^2}{4}$ systems of equations of the form,

$$\begin{bmatrix} 4 & -1 & -1 & 0 \\ -1 & 4 & 0 & -1 \\ -1 & 0 & 4 & -1 \\ 0 & -1 & -1 & 4 \end{bmatrix} \begin{bmatrix} \tilde{u}^{(k+1)}_{i,j} \\ \tilde{u}^{(k+1)}_{i+1,j} \\ \tilde{u}^{(k+1)}_{i,j+1} \\ \tilde{u}^{(k+1)}_{i+1,j+1} \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix}, \quad i,j=1,(2),m-1, \tag{4.4.18a}$$

where,

$$\left.\begin{aligned} b_1 &= u^{(k+1)}_{i,j-1} + u^{(k+1)}_{i-1,j}\ , \\ b_2 &= u^{(k+1)}_{i+1,j-1} + u^{(k)}_{i+2,j}\ , \\ b_3 &= u^{(k+1)}_{i-1,j+1} + u^{(k)}_{i,j+2} \\ \text{and} \qquad b_4 &= u^{(k)}_{i+1,j+2} + u^{(k)}_{i+2,j+1}\ . \end{aligned}\right\} \tag{4.4.18b}$$

We now set,

$$s_1 = b_1 + b_1 + b_4 + b_4 \ ,$$

and

$$s_2 = b_2 + b_2 + b_3 + b_3 \ , \tag{4.4.19}$$

hence, similar sets of equations to equation (4.2.18) can be obtained for the solution of the system (4.4.18a) and is given by,

$$\tilde{u}^{(k+1)}_{i,j} = \frac{1}{24}(7b_1 + s_2 + b_4) \ , \tag{4.4.20a}$$

$$\tilde{u}^{(k+1)}_{i+1,j} = \frac{1}{24}(7b_2 + s_1 + b_3) \ , \tag{4.4.20b}$$

$$\tilde{u}^{(k+1)}_{i,j+1} = \frac{1}{24}(7b_3 + s_1 + b_2) \ , \tag{4.4.20c}$$

$$\tilde{u}^{(k+1)}_{i+1,j+1} = \frac{1}{24}(7b_4 + s_2 + b_1) \ . \tag{4.4.20d}$$

Thus, by applying the S.O.R. technique to the above group of 4 points we have,

$$\left.\begin{aligned} u^{(k+1)}_{i,j} &= \omega(\tilde{u}^{(k+1)}_{i,j} - u^{(k)}_{i,j}) + u^{(k)}_{i,j} \ , \\ u^{(k+1)}_{i+1,j} &= \omega(\tilde{u}^{(k+1)}_{i+1,j} - u^{(k)}_{i+1,j}) + u^{(k)}_{i+1,j} \ , \\ u^{(k+1)}_{i,j+1} &= \omega(\tilde{u}^{(k+1)}_{i,j+1} - u^{(k)}_{i,j+1}) + u^{(k)}_{i,j+1} \ , \\ \text{and} \quad u^{(k+1)}_{i+1,j+1} &= \omega(\tilde{u}^{(k+1)}_{i+1,j+1} - u^{(k)}_{i+1,j+1}) + u^{(k)}_{i+1,j+1} \end{aligned}\right\} \ i,j=1(2)m \tag{4.4.21}$$

by assuming that the constant $\frac{1}{24}$ is stored beforehand, this process requires per iteration,

$$3m^2 \text{ multiplications} + \frac{13}{2}m^2 \text{ additions.} \tag{4.4.22}$$

Further savings in work can be made by solving $\tilde{u}^{(k+1)}_{i,j}$ and $\tilde{u}^{(k+1)}_{i+1,j+1}$, and using these values to determine $\tilde{u}^{(k+1)}_{i+1,j}$ and $\tilde{u}^{(k+1)}_{i,j+1}$ by applying the standard 5-point finite difference formula [Evans & Biggins, (1982)]. We calculate the values of b_1,b_2,b_3,b_4 and s_2, then $\tilde{u}^{(k+1)}_{i,j}$ and $\tilde{u}^{(k+1)}_{i+1,j+1}$ as before, and set,

$$s_3 = \tilde{u}^{(k+1)}_{i,j} + \tilde{u}^{(k+1)}_{i+1,j+1} \ . \tag{4.4.23}$$

Hence,

$$\tilde{u}^{(k+1)}_{i+1,j} = 0.25(s_3+b_2) \ ,$$

and

$$\tilde{u}^{(k+1)}_{i,j+1} = 0.25(s_3+b_3) \ . \tag{4.4.24}$$

So, by applying the above technique, the average work per iteration for m^2 internal mesh points is,

$$\frac{5}{2}m^2 \text{ multiplications} + \frac{11}{2}m^2 \text{ additions.} \qquad (4.4.25)$$

The 6-Point Group S.O.R. Method

To calculate the number of operations per iteration for this method, a technique similar to the 2-point and the 4-point groups can be applied, the 6-point Gauss-Seidel iterative method involves solving $\frac{m^2}{6}$ systems of equations of the form,

$$\left[\begin{array}{ccc|ccc} 4 & -1 & 0 & -1 & & \\ -1 & 4 & -1 & & -1 & \\ 0 & -1 & 4 & & & -1 \\ \hline -1 & & & 4 & -1 & 0 \\ & -1 & & -1 & 4 & -1 \\ & & -1 & 0 & -1 & 4 \end{array}\right] \left[\begin{array}{c} \tilde{u}^{(k+1)}_{i,j} \\ \tilde{u}^{(k+1)}_{i+1,j} \\ \tilde{u}^{(k+1)}_{i+2,j} \\ \hline \tilde{u}^{(k+1)}_{i,j+1} \\ \tilde{u}^{(k+1)}_{i+1,j+1} \\ \tilde{u}^{(k+1)}_{i+2,j+1} \end{array}\right] = \left[\begin{array}{c} b_1 \\ b_2 \\ b_3 \\ \hline b_4 \\ b_5 \\ b_6 \end{array}\right], \quad \text{for } \begin{array}{l} j=1,(2),m-1 \\ i=1,(3),m-2 \end{array}$$

(4.4.26a)

where, $b_1 = u^{(k+1)}_{i,j-1} + u^{(k+1)}_{i-1,j}$, $b_2 = u^{(k+1)}_{i+1,j-1}$,

$$b_3 = u^{(k+1)}_{i+2,j-1} + u^{(k)}_{i+3,j}, \quad b_4 = u^{(k+1)}_{i-1,j+1} + u^{(k)}_{i,j+2}, \qquad (4.4.26b)$$

$$b_5 = u^{(k)}_{i+1,j+2} \quad \text{and} \quad b_6 = u^{(k)}_{i+2,j+2} + u^{(k)}_{i+3,j+1}$$

Thus, if we set,

$$s_1 = b_1 + b_3 \quad \text{and} \quad s_2 = b_4 + b_6, \qquad (4.4.27)$$

then, a set of Gauss-Seidel iteration equations similar to the equations given in (4.2.25) can be obtained and is given by,

$$\tilde{u}^{(k+1)}_{i,j} = \frac{1}{2415}(712b_1+225b_2+208b_4+120b_5+68b_3+47b_6),$$

$$\tilde{u}^{(k+1)}_{i+1,j} = \frac{1}{161}(52b_2+17b_5+15s_1+8s_2),$$

$$\tilde{u}^{(k+1)}_{i+2,j} = \frac{1}{2415}(712b_3+225b_2+208b_6+120b_5+68b_1+47b_4),$$

$$\tilde{u}^{(k+1)}_{i,j+1} = \frac{1}{2415}(712b_4+225b_5+208b_1+120b_2+68b_6+47b_3) \;,$$

$$\tilde{u}^{(k+1)}_{i+1,j+1} = \frac{1}{161}(52b_5+17b_2+15s_2+8s_1) \;,$$

and
$$\tilde{u}^{(k+1)}_{i+2,j+1} = \frac{1}{2415}(712b_6+225b_5+208b_3+120b_2+68b_4+47b_1) . \qquad (4.4.28)$$

Now, the S.O.R. technique for the (k+1)th iteration of the group of 6 points can be applied and we have the following formulae,

$$u^{(k+1)}_{i+\ell_1,j+\ell_2} = \omega(\tilde{u}^{(k+1)}_{i+\ell_1,j+\ell_2} - u^{(k)}_{i+\ell_1,j+\ell_2}) + u^{(k)}_{i+\ell_1,j+\ell_2} . \qquad (4.4.29)$$

for $j=1,(2),m-1$, $i=1,(3),m-2$ and $\ell_1=0,1,2$, $\ell_2=0,1$.

Hence, by assuming that the constants $\frac{1}{2415}$ and $\frac{1}{161}$ are stored beforehand, this scheme requires per iteration,

$$\frac{22}{3}m^2 \text{ multiplications} + \frac{22}{3}m^2 \text{ additions.}$$

Once again this number of operations can be reduced, so that we calculate $\tilde{u}^{(k+1)}_{i,j}$,$\tilde{u}^{(k+1)}_{i+2,j}$ and $\tilde{u}^{(k+1)}_{i+1,j+1}$ first from (4.4.28) then set

$$s_3 = \tilde{u}^{(k+1)}_{i,j} + \tilde{u}^{(k+1)}_{i+1,j+1} \;,$$

$$s_4 = \tilde{u}^{(k+1)}_{i+2,j} + \tilde{u}^{(k+1)}_{i+1,j+1} \;. \qquad (4.4.30)$$

Then, we calculate the remaining three points by applying the 5-point finite difference formula. Hence we have,

$$\left.\begin{aligned}
&\tilde{u}^{(k+1)}_{i+1,j} = 0.25(\tilde{u}^{(k+1)}_{i,j} + s_4 + b_2) \;, \\
&\tilde{u}^{(k+1)}_{i,j+1} = 0.25(b_4+s_3) \;, \\
\text{and} \quad &\tilde{u}^{(k+1)}_{i+2,j+1} = 0.25(b_6+s_4) \;.
\end{aligned}\right\} \qquad (4.4.31)$$

The amount of work per iteration for m^2 internal mesh points including the over-relaxation process is,

$$\frac{14}{3}m^2 \text{ multiplications} + \frac{37}{6}m^2 \text{ additions.} \qquad (4.4.32)$$

The 9-Point Group S.O.R. Method

For this group of points, a set of Gauss-Seidel iteration equations similar to the equations given in (4.2.33) can be calculated. To determine the amount of work, first we calculate $\tilde{u}^{(k+1)}_{i+1,j}$, $\tilde{u}^{(k+1)}_{i,j+1}$, $\tilde{u}^{(k+1)}_{i+2,j+1}$ and $\tilde{u}^{(k+1)}_{i+1,j+2}$ then apply the 5-point finite difference formula to the remaining five points. Hence, we have,

$$\left.\begin{aligned}
\tilde{u}^{(k+1)}_{i+1,j} &= \frac{1}{112}(37b_2+11s_1+7s_2+5b_8+s_8) ,\\
\tilde{u}^{(k+1)}_{i,j+1} &= \frac{1}{112}(37b_4+11s_4+7\ s_5+5b_6+s_{10}) ,\\
\tilde{u}^{(k+1)}_{i+2,j+1} &= \frac{1}{112}(37b_6+11s_6+7s_5+5b_4+s_9) ,\\
\text{and, }\quad \tilde{u}^{(k+1)}_{i+1,j+2} &= \frac{1}{112}(37b_8+11s_3+7s_2+5b_2+s_7) .
\end{aligned}\right\} \quad (4.4.33a)$$

where,

$$\left.\begin{aligned}
b_1 &= u^{(k+1)}_{i,j-1}+u^{(k+1)}_{i-1,j}, & b_2 &= u^{(k+1)}_{i+1,j-1} ,\\
b_3 &= u^{(k+1)}_{i+2,j-1}+u^{(k)}_{i+3,j} , & b_4 &= u^{(k+1)}_{i-1,j+1} \\
b_6 &= u^{(k)}_{i+3,j+1} , & b_7 &= u^{(k+1)}_{i-1,j+2}+u^{(k)}_{i,j+3} ,\\
b_8 &= u^{(k)}_{i+1,j+3} \quad \text{and} & b_9 &= u^{(k)}_{i+2,j+3}+u^{(k)}_{i+3,j+2} ,
\end{aligned}\right\} \quad (4.4.33b)$$

and

$$s_1 = b_1 + b_3 , \quad s_2 = b_4 + b_6 , \quad s_3 = b_7 + b_9 ,$$

$$s_4 = b_1 + b_7 , \quad s_5 = b_2 + b_8 , \quad s_6 = b_3 + b_9 ,$$

$$s_7 = s_1+s_1+s_1 , \quad s_8 = s_3+s_3+s_3 , \quad s_9 = s_4+s_4+s_4$$

and $$s_{10} = s_6+s_6+s_6 , \qquad (4.4.34)$$

then, we set,

$$s_{11} = \tilde{u}^{(k+1)}_{i+1,j} + \tilde{u}^{(k+1)}_{i,j+1} , \quad s_{12} = \tilde{u}^{(k+1)}_{i+1,j} + \tilde{u}^{(k+1)}_{i+2,j+1} ,$$

$$s_{13} = \tilde{u}^{(k+1)}_{i,j+1} + \tilde{u}^{(k+1)}_{i+1,j+2} \quad \text{and} \quad s_{14} = \tilde{u}^{(k+1)}_{i+2,j+1} + \tilde{u}^{(k+1)}_{i+1,j+2} . \qquad (4.4.35)$$

Hence, $$\tilde{u}^{(k+1)}_{i,j} = 0.25(b_1+s_{11}) , \quad \tilde{u}^{(k+1)}_{i+2,j} = 0.25(b_3+s_{12}) ,$$

$$\tilde{u}^{(k+1)}_{i+1,j+1} = 0.25(s_{11}+s_{14}) , \quad \tilde{u}^{(k+1)}_{i,j+2} = 0.25(b_7+ s_{13}) ,$$

and $$\tilde{u}^{(k+1)}_{i+2,j+2} = 0.25(b_9+s_{14}) . \qquad (4.4.36)$$

Now, the (k+1)th iterates of the S.O.R. iterative scheme is given by,

$$u^{(k+1)}_{i+\ell_1,j+\ell_2} = \omega(\tilde{u}^{(k+1)}_{i+\ell_1,j+\ell_2} - u^{(k)}_{i+\ell_1,j+\ell_2}) + u^{(k)}_{i+\ell_1,j+\ell_2} \,, \tag{4.4.37}$$

for $i,j=1,(3),(m-2)$ and $\ell_1,\ell_2=0,1,2$.

So, by assuming that the constant $\frac{1}{112}$ is stored beforehand, this scheme requires,

$$\frac{34}{9}m^2 \text{ multiplications} + \frac{61}{9}m^2 \text{ additions.} \tag{4.4.38}$$

<u>The 12-Point Group S.O.R. Method</u>

Again, for this group, a set of Gauss-Seidel iterative equations similar to those given in (4.2.35) can be obtained. Similar to the previous cases, we first calculate $\tilde{u}^{(k+1)}_{i,j}$, $\tilde{u}^{(k+1)}_{i+2,j}$, $\tilde{u}^{(k+1)}_{i+1,j+1}$, $\tilde{u}^{(k+1)}_{i+3,j+1}$, $\tilde{u}^{(k+1)}_{i,j+2}$ and $\tilde{u}^{(k+1)}_{i+2,j+2}$, using the 15-point formula (4.2.35), then calculate the remaining 6 points using the 5-point finite difference formula, so that we have,

$$\tilde{u}^{(k+1)}_{i,j} = (413948b_1+138661b_2+137104b_5+50408b_3+44180b_9+39616b_{10}+23996b_{11}+ 18392b_8+17200b_4+10597b_{12})/1380027 \,,$$

$$\tilde{u}^{(k+1)}_{i+2,j} = (464356b_3+155861b_2+138661b_4+90288b_8+68176b_{11}+50408b_1+50213b_{10}+ 45771b_5+39616b_{12}+23996b_9)/1380027 \,,$$

$$\tilde{u}^{(k+1)}_{i+1,j+1} = (182875s_1+178277b_5+108680s_2+90288s_3+74404b_8+45771s_4)/1380027,$$

$$\tilde{u}^{(k+1)}_{i+3,j+1} = (458128b_8+137104s_4+90288s_2+45771s_1+27797b_5+18392s_3)/1380027,$$

$$\tilde{u}^{(k+1)}_{i,j+2} = (413948b_9+138661b_{10}+137104b_5+50408b_{11}+44180b_1+39616b_2+ 23996b_3+18392b_8+17200b_{12}+10597b_4)/1380027,$$

and

$$\tilde{u}^{(k+1)}_{i+2,j+2} = (464356b_{11}+155861b_{10}+138661b_{12}+90288b_8+68176b_3+ 50408b_9+50213b_2+45771b_5+39616b_4+23996b_1)/1380027 \,, \tag{4.4.39a}$$

where
$$b_1 = u^{(k+1)}_{i,j-1}+u^{(k+1)}_{i-1,j}, \quad b_2 = u^{(k+1)}_{i+1,j-1},$$
$$b_3 = u^{(k+1)}_{i+2,j-1}, \quad b_4 = u^{(k+1)}_{i+3,j-1}+u^{(k)}_{i+4,j},$$
$$b_5 = u^{(k+1)}_{i-1,j+1}, \quad b_8 = u^{(k)}_{i+4,j+1}, \tag{4.4.39b}$$
$$b_9 = u^{(k+1)}_{i-1,j+2}+u^{(k)}_{i,j+3}, b_{10} = u^{(k)}_{i+1,j+3},$$
$$b_{11} = u^{(k)}_{i+2,j+3}, \quad b_{12} = u^{(k)}_{i+3,j+3}+u^{(k)}_{i+4,j+2}$$

and $s_1 = b_2+b_{10}$, $s_2 = b_3+b_{11}$, $s_3 = b_1+b_9$

and $s_4 = b_4+b_{16}$.

Then we set,
$$s_5 = \tilde{u}^{(k+1)}_{i,j}+\tilde{u}^{(k+1)}_{i+1,j+1}, \quad s_6 = \tilde{u}^{(k+1)}_{i+2,j}+\tilde{u}^{(k+1)}_{i+3,j+1},$$
$$s_7 = \tilde{u}^{(k+1)}_{i+1,j+1}+\tilde{u}^{(k+1)}_{i+2,j+2}, \text{ and } s_8 = \tilde{u}^{(k+1)}_{i+3,j+1}+\tilde{u}^{(k+1)}_{i+2,j+2} \tag{4.4.40}$$

Hence,
$$\tilde{u}^{(k+1)}_{i+1,j} = 0.25(s_5+\tilde{u}^{(k+1)}_{i+2,j}+b_2),$$
$$\tilde{u}^{(k+1)}_{i+3,j} = 0.25(s_6+b_4),$$
$$\tilde{u}^{(k+1)}_{i,j+1} = 0.25(s_5+\tilde{u}^{(k+1)}_{i,j+2}+b_5), \tag{4.4.41}$$
$$\tilde{u}^{(k+1)}_{i+2,j+1} = 0.25(s_6+s_7),$$
$$\tilde{u}^{(k+1)}_{i+1,j+2} = 0.25(s_7+\tilde{u}^{(k+1)}_{i,j+2}+b_{10}),$$
and
$$\tilde{u}^{(k+1)}_{i+3,j+2} = 0.25(s_8+b_{12}).$$

Now, by applying the S.O.R. iterative method to the (k+1)th iterate we have,

$$u^{(k+1)}_{i+\ell_1,j+\ell_2} = \omega(\tilde{u}^{(k+1)}_{i+\ell_1,j+\ell_2}-u^{(k)}_{i+\ell_1,j+\ell_2})+u^{(k)}_{i+\ell_1,j+\ell_2}, \tag{4.4.42}$$

for $j=1,(3),m-2$, $i=1,(4),m-3$ and $\ell_1=0,1,2,3$, $\ell_2=0,1,2$.

Hence, the number of operations per iteration for the m^2 internal mesh points, assuming that the constant $\frac{1}{1380027}$ is stored beforehand, is
$$\frac{35}{6}m^2 \text{ multiplications } + \frac{91}{12}m^2 \text{ additions.} \tag{4.4.43}$$

The 16-Point Group S.O.R. Method

For this grouping of points the Gauss-Seidel iterative method involves solving $\frac{m^2}{16}$ systems of equations, and each system consists of 17 equations similar to the equations given in (4.2.39). Again the amount of work can be reduced by calculating 8 of the points using the 16-point explicit formula (4.2.39). Then calculate the other 8 points by the usual 5-point finite difference formula. Hence we first calculate,

$$\tilde{u}^{(k+1)}_{i,j} = (1987b_1+674s_1+251s_2+101s_3+88s_4+74s_5+37b_{16})/6600,$$

$$\tilde{u}^{(k+1)}_{i+2,j} = (2238b_3+762b_2+674b_4+458b_8+251b_1+242s_6+162b_{15}+158b_9+138b_{14}+ 101b_{16}+74b_{13})/6600,$$

$$\tilde{u}^{(k+1)}_{i+1,j+1} = (916s_1+559s_2+458b_1+409s_3+316s_5+242s_4+158b_{16})/6600 ,$$

$$\tilde{u}^{(k+1)}_{i+3,j+1} = (2238b_8+762b_{12}+674b_4+458b_3+251b_{16}+242s_7+162b_5+158b_{14}+ +138b_9+101b_1+74b_{13})/6600 ,$$

$$\tilde{u}^{(k+1)}_{i,j+2} = (2238b_9+762b_5+674b_{13}+458b_{14}+251b_1+242s_7+162b_{12}+158b_3+ 138b_8+101b_{16}+74b_4)/6600 ,$$

$$\tilde{u}^{(k+1)}_{i+2,j+2} = (916s_5+559s_3+458b_{16}+409s_2+316s_1+242s_4+158b_1)/6600 ,$$

$$\tilde{u}^{(k+1)}_{i+1,j+3} = (2238b_{14}+762b_{15}+674b_{13}+458b_9+251b_{16}+242s_6+162b_2+ 158b_8+138b_3+101b_1+74b_4)/6600 ,$$

and $$\tilde{u}^{(k+1)}_{i+3,j+3} = (1987b_{16}+674s_5+251s_3+101s_2+88s_4+74s_1+37b_1)/6600 , \qquad (4.4.44a)$$

where,

$$b_1 = u^{(k+1)}_{i,j-1}+u^{(k+1)}_{i-1,j} , \qquad b_2 = u^{(k+1)}_{i+1,j-1} ,$$

$$b_3 = u^{(k+1)}_{i+2,j-1} , \qquad b_4 = u^{(k+1)}_{i+3,j-1}+u^{(k)}_{i+4,j} ,$$

$$b_5 = u^{(k+1)}_{i-1,j+1} , \qquad b_{12} = u^{(k)}_{i+4,j+2} , \qquad (4.4.44b)$$

$$b_{13} = u^{(k+1)}_{i-1,j+3}+u^{(k)}_{i,j+4} , \qquad b_{14} = u^{(k)}_{i+1,j+4} ,$$

$$b_{15} = u^{(k)}_{i+2,j+4} , \qquad b_{16} = u^{(k)}_{i+3,j+4}+u^{(k)}_{i+4,j+3} ,$$

$$b_8 = u^{(k)}_{i+4,j+1} , \qquad b_9 = u^{(k+1)}_{i-1,j+2} ,$$

and $s_1 = b_2 + b_5,\quad s_2 = b_3 + b_9,\quad s_3 = b_8 + b_{14}$,

$s_4 = b_4 + b_{13},\quad s_5 = b_{12} + b_{15}\ , s_6 = b_5 + b_{12}$

and $s_7 = b_2 + b_{15}$.

Then, we set,

$$
\begin{aligned}
& s_8 = \tilde{u}^{(k+1)}_{i,j} + \tilde{u}^{(k+1)}_{i+1,j+1},\quad s_9 = \tilde{u}^{(k+1)}_{i+2,j} + \tilde{u}^{(k+1)}_{i+3,j+1}\ , \\
& s_{10} = \tilde{u}^{(k+1)}_{i+1,j+1} + \tilde{u}^{(k+1)}_{i+2,j+2},\quad s_{11} = \tilde{u}^{(k+1)}_{i,j+2} + \tilde{u}^{(k+1)}_{i+1,j+3}\ , \\
\text{and}\quad & s_{12} = \tilde{u}^{(k+1)}_{i+2,j+2} + \tilde{u}^{(k+1)}_{i+3,j+3}\ .
\end{aligned}
\tag{4.4.45}
$$

Hence, by applying the 5-point finite difference formula to the remaining 8 points we have,

$$
\begin{aligned}
& \tilde{u}^{(k+1)}_{i+1,j} = 0.25(s_8 + \tilde{u}^{(k+1)}_{i+2,j} + b_2)\ , \\
& \tilde{u}^{(k+1)}_{i+3,j} = 0.25(s_9 + b_4)\ , \\
& \tilde{u}^{(k+1)}_{i,j+1} = 0.25(s_8 + \tilde{u}^{(k+1)}_{i,j+2} + b_5\ ), \\
& \tilde{u}^{(k+1)}_{i+2,j+1} = 0.25(s_9 + s_{10})\ , \\
& \tilde{u}^{(k+1)}_{i+1,j+2} = 0.25(s_{10} + s_{11})\ , \\
& \tilde{u}^{(k+1)}_{i+3,j+2} = 0.25(s_{12} + \tilde{u}^{(k+1)}_{i+3,j+1} + b_{12})\ , \\
& \tilde{u}^{(k+1)}_{i,j+3} = 0.25(s_{11} + b_{13})\ , \\
\text{and}\quad & \tilde{u}^{(k+1)}_{i+2,j+3} = 0.25(s_{12} + \tilde{u}^{(k+1)}_{i+1,j+3} + b_{15})\ .
\end{aligned}
\tag{4.4.46}
$$

The (k+1)th iterates of the S.O.R. iterative scheme are given by,

$$
u^{(k+1)}_{i+\ell_1,j+\ell_2} = \omega(\tilde{u}^{(k+1)}_{i+\ell_1,j+\ell_2} - u^{(k)}_{i+\ell_1,j+\ell_2}) + u^{(k)}_{i+\ell_1,j+\ell_2}\ , \tag{4.4.47}
$$

for $i,j=1,(3),m-3$ and $\ell_1,\ell_2=0,1,2,3$.

Hence, for this scheme the average work per iteration for the m^2 internal mesh points, assuming that the constant $\frac{1}{6600}$ is stored beforehand, is

$$
6m^2 \text{ multiplications} + \frac{31}{4} m^2 \text{ additions.} \tag{4.4.48}
$$

The 25-Point Group S.O.R. Method

Finally, in this series of groups, we consider the group of 25 points, in which the Gauss-Seidel iterative method involves solving $\frac{m^2}{25}$ systems of equations, each system consisint of 25 equations similar to those given in (4.2.43). In a scheme similar to the previous groups, we first calculate 13 points out of the 25 points applying the 21-point explicit formula (4.2.43), then calculate the remaining 12 points using the 5-point finite difference formula. So we first calculate,

$$\tilde{u}^{(k+1)}_{i,j} = (31067b_1+10654s_1+4139s_2+1736s_3+904s_4+811s_5+660s_6+566s_7+283b_{25})/102960 ,$$

$$\tilde{u}^{(k+1)}_{i+2,j} = (35866b_3+12390s_8+4166s_9+4139s_{10}+2970s_{11}+1774s_{12}+1754b_{23}+1470s_{13}+811s_{14})/102960 ,$$

$$\tilde{u}^{(k+1)}_{i+4,j} = (31067b_5+10654s_{15}+4139s_{16}+1736s_{17}+904s_{18}+811s_{19}+660s_{20}+566s_{21}+283b_{21})/102960 ,$$

$$\tilde{u}^{(k+1)}_{i+1,j+1} = (76s_1+49s_2+38b_1+26s_3+18s_4+17s_5+12s_7+11s_6+6b_{25})/528 ,$$

$$\tilde{u}^{(k+1)}_{i+3,j+1} = (76s_{15}+49s_{16}+38b_5+26s_{17}+18s_{18}+17s_{19}+12s_{21}+11s_{20}+6b_{21})/528,$$

$$\tilde{u}^{(k+1)}_{i,j+2} = (35866b_{11}+12390s_{22}+4166s_{23}+4139s_{24}+2970s_{25}+1774s_{26}+1754b_{15}+1470s_{27}+811s_{28})/102960 ,$$

$$\tilde{u}^{(k+1)}_{i+2,j+2} = [8(s_2+s_5)+6(s_1+s_7+s_{15}+s_{21})+3(s_{10}+s_{14})]/104 ,$$

$$\tilde{u}^{(k+1)}_{i+4,j+2} = 35866b_{15}+12390s_{27}+4166s_{26}+4139s_{28}+2970s_{25}+1774s_{23}+1754b_{11}+1470s_{22}+811s_{24})/102960 ,$$

$$\tilde{u}^{(k+1)}_{i+1,j+3} = (76s_{21}+49s_{19}+38b_{21}+26s_{18}+18s_{17}+17s_{16}+12s_{15}+11s_{20}+6b_5)/528,$$

$$\tilde{u}^{(k+1)}_{i+3,j+3} = (76s_7+49s_5+38b_{25}+26s_4+18s_3+17s_2+12s_1+11s_6+6b_1)/528 ,$$

$$\tilde{u}^{(k+1)}_{i,j+4} = (31067b_{21}+10654s_{21}+4139s_{19}+1736s_{18}+904s_{17}+811s_{16}+660s_{20}+566s_{15}+283b_5)/102960 ,$$

$$\tilde{u}^{(k+1)}_{i+2,j+4} = (35866b_{23}+12390s_{13}+4166s_{12}+4139s_{14}+2970s_{11}+1774s_9 +1754b_3+1470s_8+811s_{10})/102960 ,$$

and
$$\tilde{u}^{(k+1)}_{i+4,j+4} = (31067b_{25}+10654s_7+4139s_5+1736s_4+904s_3+811s_2+660s_6+ 566s_1+283b_1)/102960 , \qquad (4.4.49a)$$

where,

$$\begin{aligned}
&b_1 = u^{(k+1)}_{i,j-1}+u^{(k+1)}_{i-1,j} , && b_2 = u^{(k+1)}_{i+1,j-1} ,\\
&b_3 = u^{(k+1)}_{i+2,j-1} , && b_4 = u^{(k+1)}_{i+3,j-1} ,\\
&b_5 = u^{(k+1)}_{i+4,j-1}+u^{(k)}_{i+5,j}, && b_6 = u^{(k+1)}_{i-1,j+1} ,\\
&b_{10} = u^{(k)}_{i+5,j+1} , && b_{11} = u^{(k+1)}_{i-1,j+2} ,\\
&b_{15} = u^{(k)}_{i+5,j+2} , && b_{16} = u^{(k+1)}_{i-1,j+3} ,\\
&b_{20} = u^{(k)}_{i+5,j+3} , && b_{21} = u^{(k+1)}_{i-1,j+4}+u^{(k)}_{i,j+5} ,\\
&b_{22} = u^{(k)}_{i+1,j+5} , && b_{23} = u^{(k)}_{i+2,j+5} ,\\
&b_{24} = u^{(k)}_{i+3,j+5} , && b_{25} = u^{(k)}_{i+4,j+5}+u^{(k)}_{i+5,j+4} , \qquad (4.4.49b)
\end{aligned}$$

and,

$$\begin{aligned}
&s_1 = b_2 + b_6 , && s_2 = b_3 + b_{11}, && s_3 = b_4 + b_{16} ,\\
&s_4 = b_{10} + b_{22} , && s_5 = b_{15} + b_{23}, && s_6 = b_5 + b_{21} ,\\
&s_7 = b_{20} + b_{24}, && s_8 = b_2 + b_4 , && s_9 = b_6 + b_{10},\\
&s_{10} = b_1 + b_5 , && s_{11} = b_{11} + b_{15}, && s_{12} = b_{16} + b_{20} ,\\
&s_{13} = b_{22} + b_{24}, && s_{14} = b_{21} + b_{25}, && s_{15} = b_4 + b_{10} ,\\
&s_{16} = b_3 + b_{15} , && s_{17} = b_2 + b_{20}, && s_{18} = b_6 + b_{24} ,\\
&s_{19} = b_{11}+b_{23}, && s_{20} = b_1 + b_{25}, && s_{21} = b_{16} + b_{22} ,\\
&s_{22} = b_6 + b_{16}, && s_{23} = b_2 + b_{22}, && s_{24} = b_1 + b_{21} ,\\
&s_{25} = b_3 + b_{23} , && s_{26} = b_4 + b_{24}, && s_{27} = b_{10} + b_{20} ,
\end{aligned}$$

and $s_{28} = b_5 + b_{25}$.

Now, we set,

$$s_{29} = \tilde{u}^{(k+1)}_{i,j} + \tilde{u}^{(k+1)}_{i+1,j+1}\,, \quad s_{30} = \tilde{u}^{(k+1)}_{i+2,j} + \tilde{u}^{(k+1)}_{i+3,j+1}\,,$$

$$s_{31} = \tilde{u}^{(k+1)}_{i+1,j+1} + \tilde{u}^{(k+1)}_{i+2,j+2}\,, \quad s_{32} = \tilde{u}^{(k+1)}_{i+3,j+1} + \tilde{u}^{(k+1)}_{i+4,j+2}\,, \qquad (4.4.50)$$

$$s_{33} = \tilde{u}^{(k+1)}_{i,j+2} + \tilde{u}^{(k+1)}_{i+1,j+3}\,, \quad s_{34} = \tilde{u}^{(k+1)}_{i+2,j+2} + \tilde{u}^{(k+1)}_{i+3,j+3}\,,$$

$$s_{35} = \tilde{u}^{(k+1)}_{i+1,j+3} + \tilde{u}^{(k+1)}_{i+2,j+4} \quad \text{and} \quad s_{36} = \tilde{u}^{(k+1)}_{i+3,j+3} + \tilde{u}^{(k+1)}_{i+4,j+4}\,.$$

Then, the 5-Point finite difference formula is applied at each remaining, interior point, to give,

$$\begin{aligned}
\tilde{u}^{(k+1)}_{i+1,j} &= 0.25(s_{29} + \tilde{u}^{(k+1)}_{i+3,j} + b_2)\,,\\
\tilde{u}^{(k+1)}_{i+3,j} &= 0.25(s_{30} + \tilde{u}^{(k+1)}_{i+4,j} + b_4)\,,\\
\tilde{u}^{(k+1)}_{i,j+1} &= 0.25(s_{29} + \tilde{u}^{(k+1)}_{i,j+2} + b_6)\,,\\
\tilde{u}^{(k+1)}_{i+2,j+1} &= 0.25(s_{30} + s_{31})\,,\\
\tilde{u}^{(k+1)}_{i+4,j+1} &= 0.25(s_{32} + \tilde{u}^{(k+1)}_{i+4,j} + b_{10})\,,\\
\tilde{u}^{(k+1)}_{i+1,j+2} &= 0.25(s_{31} + s_{33})\,,\\
\tilde{u}^{(k+1)}_{i+3,j+2} &= 0.25(s_{32} + s_{34})\,, \qquad (4.4.51)\\
\tilde{u}^{(k+1)}_{i,j+3} &= 0.25(s_{33} + \tilde{u}^{(k+1)}_{i,j+4} + b_{21})\,,\\
\tilde{u}^{(k+1)}_{i+2,j+3} &= 0.25(s_{34} + s_{35})\,,\\
\tilde{u}^{(k+1)}_{i+4,j+3} &= 0.25(s_{36} + \tilde{u}^{(k+1)}_{i+4,j+2} + b_{20})\,,\\
\tilde{u}^{(k+1)}_{i+1,j+4} &= 0.25(s_{35} + \tilde{u}^{(k+1)}_{i,j+4} + b_{22})\,,\\
\text{and} \quad \tilde{u}^{(k+1)}_{i+3,j+4} &= 0.25(s_{36} + \tilde{u}^{(k+1)}_{i+2,j+4} + b_{24})\,.
\end{aligned}$$

Now, we modify the calculated points by applying the extrapolation process to give the S.O.R. iterative method, i.e.,

$$u^{(k+1)}_{i+\ell_1,j+\ell_2} = \omega(\tilde{u}^{(k+1)}_{i+\ell_1,j+\ell_2} - u^{(k)}_{i+\ell_1,j+\ell_2}) + u^{(k)}_{i+\ell_1,j+\ell_2} \qquad (4.4.52)$$

for $i,j=1,(4),m-4$ and $\ell_1,\ell_2=0,1,2,3,4$.

Hence, the average work per iteration for the m^2 internal mesh points for this method is,

$$\frac{148}{25} m^2 \text{ multiplications } + \frac{213}{25} m^2 \text{ additions,} \qquad (4.4.53)$$

assuming that the constants $\frac{1}{102960}$, $\frac{1}{528}$ and $\frac{1}{104}$ are stored beforehand.

The results for the experimental number of iterations shown in Tables (4.3.1)-(4.3.10) can be combined with the number of operations in each iteration required by each of the methods given in (4.4.3), (4.4.7), (4.4.11), (4.4.17), (4.4.25), (4.4.32), (4.4.38), (4.4.43), (4.4.48) and (4.4.53) to give the total number of arithmetic operations required for a solution and these are recorded in Table (4.4.1a), (4.4.1b) and (4.4.1c).

Method / h^{-1}	Point S.O.R.		1-Line S.O.R.		2-Line S.O.R.	
	Multiplication	Addition	M	A	M	A
13	$78m^2$	$195m^2$	$75m^2-25m$	$125m^2$	$90m^2-45m$	$108m^2-18m$
25	$146m^2$	$365m^2$	$141m^2-47m$	$235m^2$	$170m^2-85m$	$204m^2-34m$
37	$216m^2$	$540m^2$	$204m^2-68m$	$340m^2$	$250m^2-125m$	$300m^2-50m$
49	$286m^2$	$715m^2$	$264m^2-88m$	$440m^2$	$380m^2-165m$	$396m^2-66m$
61	$358m^2$	$895m^2$	$327m^2-109m$	$545m^2$	$405m^2-202m$	$486m^2-81m$

TABLE (4.4.1a)

1362.5 1498.5

$1790m^2$

Method / h^{-1}	2-Point Group S.O.R.		4-Point Group S.O.R.		6-Point Group S.O.R.		9-Point Group S.O.R.	
	M	A	M	A	M	A	M	A
13	$82.5m^2$	$165m^2$	$62.5m^2$	$137.5m^2$	$112m^2$	$148m^2$	$79.3m^2$	$142.3m^2$
25	$145m^2$	$290m^2$	$120m^2$	$264m^2$	$205.3m^2$	$271.3m^2$	$147.3m^2$	$264.3m^2$
37	$215m^2$	$430m^2$	$175m^2$	$385m^2$	$303.3m^2$	$400.8m^2$	$222.9m^2$	$399.9m^2$
49	$287.5m^2$	$575m^2$	$235m^2$	$517m^2$	$396.7m^2$	$524.2m^2$	$294.7m^2$	$528.7m^2$
61	$352.5m^2$	$705m^2$	$290m^2$	$638m^2$	$494.7m^2$	$653.7m^2$	$351.3m^2$	$630.3m^2$

TABLE (4.4.1b)

Method / h^{-1}	12-Point Group S.O.R.		16-Point Group S.O.R.		25-Point Group S.O.R.		
	M	A	M	A	h^{-1}	M	A
13	$122.5m^2$	$159.25m^2$	$114m^2$	$147.25m^2$	11	$88.8m^2$	$127.8m^2$
25	$221.7m^2$	$288.2m^2$	$210m^2$	$271.25m^2$	26	$195.4m^2$	$281.2m^2$
37	$326.7m^2$	$424.7m^2$	$300m^2$	$387.5m^2$	36	$266.4m^2$	$383.4m^2$
49	$425.8m^2$	$553.6m^2$	$402m^2$	$519.25m^2$	51	$373m^2$	$536.7m^2$
61	$519.2m^2$	$674.9m^2$	$498m^2$	$643.25m^2$	61	$444m^2$	$639m^2$

TABLE (4.4.1c)

4.5 THE 3-SPACE DIMENSIONAL CASE

Within the past few years a growing desire to solve three-dimensional problems, together with the development of novel computer architectures - array processors, vector machines and multiprocessors - has rekindled interest in block iterative methods for elliptic systems.

In this section we present an explicit 8-point group iterative method to solve the elliptic p.d.e. in 3-space dimensions. First, consider the Laplace's equation in 3-space dimensions,

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} + \frac{\partial^2 U}{\partial z^2} = 0 \ , \qquad (4.5.1)$$

defined in the unit cube, $0 \leq x,y,z \leq 1$, with m^3 internal mesh points. We order the mesh points in groups of 8 as shown in Figure (4.5.1).

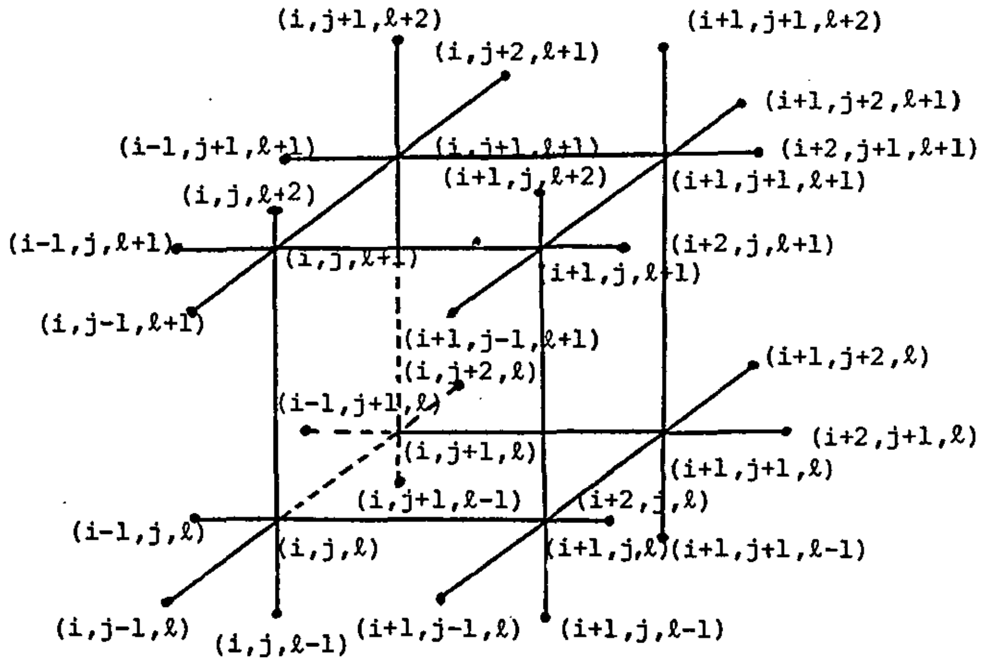


FIGURE (4.5.1)

and the groups themselves are ordered in red-black ordering, see Figure (4.5.2).

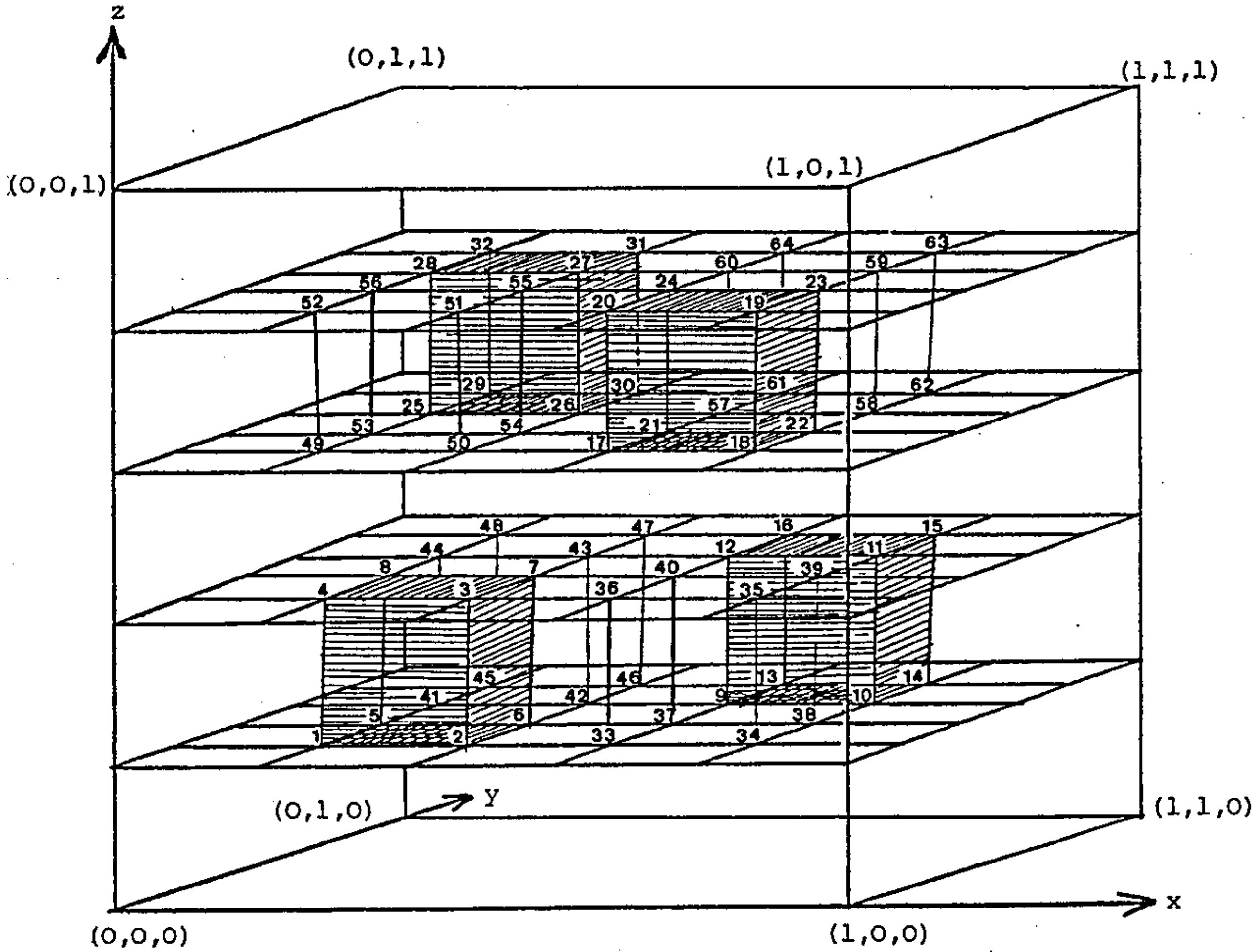


FIGURE (4.5.2)

By using the 7-point approximation scheme, the finite difference equation at the point, (i,j,ℓ), has the form,

$$u_{i,j,\ell}+\alpha_1 u_{i-1,j,\ell}+\alpha_2 u_{i,j-1,\ell}+\alpha_3 u_{i+1,j,\ell}+\alpha_4 u_{i,j+1,\ell}+\alpha_5 u_{i,j,\ell-1}+\alpha_6 u_{i,j,\ell-1}$$

$$= b_{(i,j,\ell)} \quad , \qquad (4.5.2)$$

This can be represented diagrammatically as a *stencil* to be applied to each of the internal mesh points in turn, as shown in Figure (4.5.3).

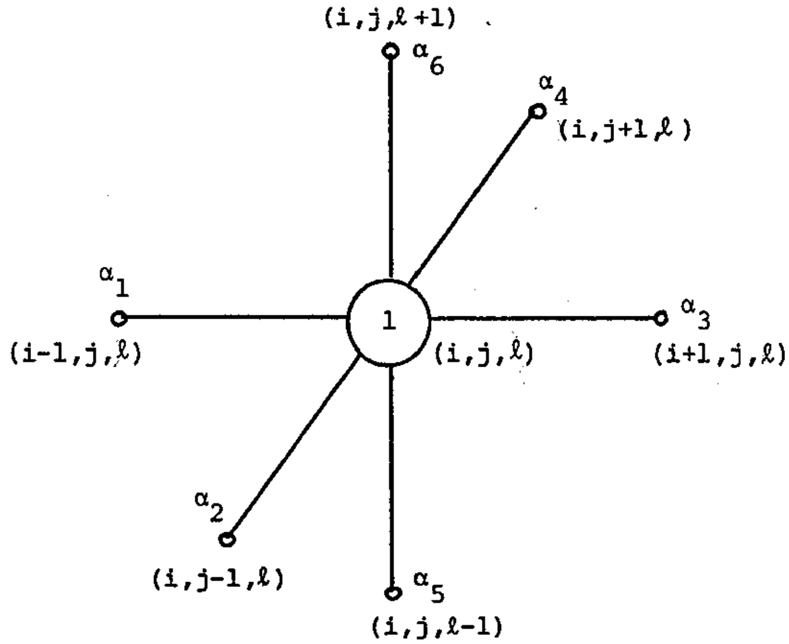


FIGURE (4.5.3)

The resulting coefficient matrix A then, illustrated here for a 4×4×4 mesh, has the block structure:

$$A = \left(\begin{array}{cccc|cccc}
R_0 & & \bigcirc & & R_3 & R_4 & R_6 & 0 \\
 & R_0 & & & R_2 & R_1 & 0 & R_6 \\
\bigcirc & & R_0 & & R_5 & 0 & R_1 & R_4 \\
 & & & R_0 & 0 & R_5 & R_2 & R_3 \\
\hline
R_1 & R_4 & R_6 & 0 & R_0 & & \bigcirc & \\
R_2 & R_3 & 0 & R_6 & & R_0 & & \\
R_5 & 0 & R_3 & R_4 & \bigcirc & & R_0 & \\
0 & R_5 & R_2 & R_1 & & & & R_0
\end{array}\right) \qquad (4.5.3a)$$

where,

$$R_0 = \left(\begin{array}{cccc|cccc}
1 & \alpha_3 & 0 & \alpha_6 & \alpha_4 & & & \\
\alpha_1 & 1 & \alpha_6 & 0 & & \alpha_4 & \bigcirc & \\
0 & \alpha_5 & 1 & \alpha_1 & \bigcirc & & \alpha_4 & \\
\alpha_5 & 0 & \alpha_3 & 1 & & & & \alpha_4 \\
\hline
\alpha_2 & & & & 1 & \alpha_3 & 0 & \alpha_6 \\
 & \alpha_2 & \bigcirc & & \alpha_1 & 1 & \alpha_6 & 0 \\
\bigcirc & & \alpha_2 & & 0 & \alpha_5 & 1 & \alpha_1 \\
 & & & \alpha_2 & \alpha_5 & 0 & \alpha_3 & 1
\end{array}\right), \quad R_1 = \left(\begin{array}{cccc|cccc}
0 & \alpha_1 & 0 & 0 & & & & \\
0 & 0 & 0 & 0 & & \bigcirc & & \\
0 & 0 & 0 & 0 & & & & \\
0 & 0 & \alpha_1 & 0 & & & & \\
\hline
 & & & & 0 & \alpha_1 & 0 & 0 \\
 & \bigcirc & & & 0 & 0 & 0 & 0 \\
 & & & & 0 & 0 & 0 & 0 \\
 & & & & 0 & 0 & \alpha_1 & 0
\end{array}\right)$$

$$R_2 = \left(\begin{array}{c|cccc} & \alpha_2 & & & O \\ O & & \alpha_2 & & \\ & & & \alpha_2 & \\ & O & & & \alpha_2 \\ \hline & & & & \\ O & & & O & \\ & & & & \end{array}\right), \quad R_3 = \left(\begin{array}{cccc|cccc} 0 & 0 & 0 & 0 & & & & \\ \alpha_3 & 0 & 0 & 0 & & O & & \\ 0 & 0 & 0 & \alpha_3 & & & & \\ 0 & 0 & 0 & 0 & & & & \\ \hline & & & & 0 & 0 & 0 & 0 \\ & & & & \alpha_3 & 0 & 0 & 0 \\ & O & & & 0 & 0 & 0 & \alpha_3 \\ & & & & 0 & 0 & 0 & 0 \end{array}\right)$$

$$R_4 = \left(\begin{array}{cccc|c} & & & & \\ & O & & & O \\ & & & & \\ \hline \alpha_4 & & O & & \\ & \alpha_4 & & & O \\ O & & \alpha_4 & & \\ & & & \alpha_4 & \end{array}\right), \quad R_5 = \left(\begin{array}{cccc|cccc} 0 & 0 & 0 & \alpha_5 & & & & \\ 0 & 0 & \alpha_5 & 0 & & O & & \\ 0 & 0 & 0 & 0 & & & & \\ 0 & 0 & 0 & 0 & & & & \\ \hline & & & & 0 & 0 & 0 & \alpha_5 \\ & & & & 0 & 0 & \alpha_5 & 0 \\ & O & & & 0 & 0 & 0 & 0 \\ & & & & 0 & 0 & 0 & 0 \end{array}\right),$$

and

$$R_6 = \left(\begin{array}{cccc|cccc} 0 & 0 & 0 & 0 & & & & \\ 0 & 0 & 0 & 0 & & O & & \\ 0 & \alpha_6 & 0 & 0 & & & & \\ \alpha_6 & 0 & 0 & 0 & & & & \\ \hline & & & & 0 & 0 & 0 & 0 \\ & O & & & 0 & 0 & 0 & 0 \\ & & & & 0 & \alpha_6 & 0 & 0 \\ & & & & \alpha_6 & 0 & 0 & 0 \end{array}\right) . \qquad (4.5.3b)$$

This matrix has Property $A^{(\pi)}$ and is π-consistently ordered. Hence, the theory of block S.O.R. is valid.

For the derivation of the explicit group S.O.R. method for this problem, we calculate the transformed matrix A^E and the modified vector $\underline{b}^E$, where, once again,

$$A^E = [\mathrm{diag}\{R_O\}]^{-1}A \ , \tag{4.5.4}$$

and
$$\underline{b}^E = [\mathrm{diag}\{R_O\}]^{-1}\underline{b} \ . \tag{4.5.5}$$

The matrix $[\mathrm{diag}\{R_O\}]^{-1}$ is simply the $\mathrm{diag}\{R_O^{-1}\}$.

Now, to determine the inverse of R_O, assume that,

$$R_O = \begin{bmatrix} T & D_1 \\ D_2 & T \end{bmatrix} \ , \tag{4.5.6a}$$

where,

$$T = \begin{bmatrix} 1 & \alpha_3 & 0 & \alpha_6 \\ \alpha_1 & 1 & \alpha_6 & 0 \\ 0 & \alpha_5 & 1 & \alpha_1 \\ \alpha_5 & 0 & \alpha_3 & 1 \end{bmatrix}, \ D_1 = \begin{bmatrix} \alpha_4 & & & \\ & \alpha_4 & & O \\ & & \alpha_4 & \\ O & & & \alpha_4 \end{bmatrix} \text{ and } D_2 = \begin{bmatrix} \alpha_2 & & & \\ & \alpha_2 & & O \\ & & \alpha_2 & \\ O & & & \alpha_2 \end{bmatrix} \tag{4.5.6b}$$

If we set,

$$R_O^{-1} = \begin{bmatrix} B_1 & C_1 \\ C_2 & B_2 \end{bmatrix} \ , \tag{4.5.7}$$

then,

$$C_1 = [D_2 - TD_1^{-1}T]^{-1} \ , \tag{4.5.8a}$$

$$C_2 = [D_1 - TD_2^{-1}T]^{-1} \ , \tag{4.5.8b}$$

$$B_1 = -C_1TD_1^{-1}, \tag{4.5.8c}$$

and
$$B_2 = -C_2TD_2^{-1} \ . \tag{4.5.8d}$$

First, we determine the matrix $TD_1^{-1}T$, which is given by,

$$TD_1^{-1}T = \frac{1}{\alpha_4}\begin{bmatrix} 1+\alpha_1\alpha_3+\alpha_5\alpha_6 & 2\alpha_3 & 2\alpha_3\alpha_6 & 2\alpha_6 \\ 2\alpha_1 & 1+\alpha_1\alpha_3+\alpha_5\alpha_6 & 2\alpha_6 & 2\alpha_1\alpha_6 \\ 2\alpha_1\alpha_5 & 2\alpha_5 & 1+\alpha_1\alpha_3+\alpha_5\alpha_6 & 2\alpha_1 \\ 2\alpha_5 & 2\alpha_3\alpha_5 & 2\alpha_3 & 1+\alpha_1\alpha_3+\alpha_5\alpha_6 \end{bmatrix} . \tag{4.5.9}$$

Hence,

$$C_1 = \begin{bmatrix} \frac{\beta}{\alpha_4} & -\frac{2\alpha_3}{\alpha_4} & -\frac{2\alpha_3}{\alpha_4} & -\frac{2\alpha_6}{\alpha_4} \\ -\frac{2\alpha_1}{\alpha_4} & \frac{\beta}{\alpha_4} & -\frac{2\alpha_6}{\alpha_4} & -\frac{2\alpha_1\alpha_6}{\alpha_4} \\ -\frac{2\alpha_1\alpha_5}{\alpha_4} & -\frac{2\alpha_5}{\alpha_4} & \frac{\beta}{\alpha_4} & -\frac{2\alpha_1}{\alpha_4} \\ -\frac{2\alpha_5}{\alpha_4} & -\frac{2\alpha_3\alpha_5}{\alpha_4} & -\frac{2\alpha_3}{\alpha_4} & \frac{\beta}{\alpha_4} \end{bmatrix}^{-1} , \qquad (4.5.10)$$

where, $\beta = \alpha_2\alpha_4 - \alpha_1\alpha_3 - \alpha_5\alpha_6 - 1$.

Similarly, $TD_2^{-1}T$ can be obtained by replacing α_4 by α_2 in (4.5.9) and then,

$$C_2 = \begin{bmatrix} \frac{\beta}{\alpha_2} & -\frac{2\alpha_3}{\alpha_2} & -\frac{2\alpha_3\alpha_6}{\alpha_2} & -\frac{2\alpha_6}{\alpha_2} \\ -\frac{2\alpha_1}{\alpha_2} & \frac{\beta}{\alpha_2} & -\frac{2\alpha_6}{\alpha_2} & -\frac{2\alpha_1\alpha_6}{\alpha_2} \\ -\frac{2\alpha_1\alpha_5}{\alpha_2} & -\frac{2\alpha_5}{\alpha_2} & \frac{\beta}{\alpha_2} & -\frac{2\alpha_1}{\alpha_2} \\ -\frac{2\alpha_5}{\alpha_2} & -\frac{2\alpha_3\alpha_5}{\alpha_2} & -\frac{2\alpha_3}{\alpha_2} & \frac{\beta}{\alpha_2} \end{bmatrix}^{-1} , \qquad (4.5.11)$$

C_1 can then be determined and is given by,

$$C_1 = \frac{\alpha_4}{K} \begin{bmatrix} G & 2\alpha_3 H & 2\alpha_3\alpha_6 L & 2\alpha_6 V \\ 2\alpha_1 H & G & 2\alpha_6 V & 2\alpha_1\alpha_6 L \\ 2\alpha_1\alpha_5 L & 2\alpha_5 V & G & 2\alpha_1 H \\ 2\alpha_5 V & 2\alpha_3\alpha_5 L & 2\alpha_3 H & G \end{bmatrix} , \qquad (4.5.12)$$

C_2 is similar to C_1 but instead of $\frac{\alpha_4}{K}$, we will have $\frac{\alpha_2}{K}$ multiplied by the (4×4) matrix. Now, we substitute the two matrices C_1 and C_2 in (4.5.8c) and (4.5.8d) respectively to obtain,

$$B_1 = B_2 = -\frac{1}{K} \begin{pmatrix} F & \alpha_3 E & 2\alpha_3\alpha_6 M & \alpha_6 Y \\ \alpha_1 E & F & \alpha_6 Y & 2\alpha_1\alpha_6 M \\ 2\alpha_1\alpha_5 M & \alpha_5 Y & F & \alpha_1 E \\ \alpha_5 Y & 2\alpha_3\alpha_5 M & \alpha_3 E & F \end{pmatrix} , \qquad (4.5.13)$$

where,

$$K = \alpha_2^4\alpha_4^4 + \alpha_1^4\alpha_3^4 + \alpha_5^4\alpha_6^4 + 4(\alpha_1\alpha_3\alpha_2\alpha_4+\alpha_1\alpha_3\alpha_5\alpha_6+\alpha_2\alpha_4\alpha_5\alpha_6+\alpha_1^2\alpha_3^2\alpha_2\alpha_4\alpha_5\alpha_6$$
$$+\alpha_1\alpha_3\alpha_2\alpha_4\alpha_5^2\alpha_6^2+\alpha_1\alpha_3\alpha_2^2\alpha_4^2\alpha_5\alpha_6+\alpha_1\alpha_3\alpha_5^2\alpha_6^2+\alpha_1^2\alpha_3^2\alpha_5\alpha_6+\alpha_1^2\alpha_3^2\alpha_2\alpha_4+\alpha_1\alpha_3\alpha_2^2\alpha_4^2$$
$$+\alpha_2^2\alpha_4^2\alpha_5\alpha_6+\alpha_2\alpha_4\alpha_5^2\alpha_6^2-\alpha_1\alpha_3\alpha_5^3\alpha_6^3-\alpha_1^3\alpha_3^3\alpha_5\alpha_6-\alpha_1\alpha_3\alpha_2^3\alpha_4^3-\alpha_1^3\alpha_3^3\alpha_2\alpha_4-\alpha_2\alpha_4\alpha_5^3\alpha_6^3$$
$$-\alpha_2^3\alpha_4^3\alpha_5\alpha_6-\alpha_2^3\alpha_4^3-\alpha_1^3\alpha_3^3-\alpha_5^3\alpha_6^3-\alpha_2\alpha_4-\alpha_1\alpha_3-\alpha_5\alpha_6)+6(\alpha_2^2\alpha_4^2+\alpha_1^2\alpha_3^2+\alpha_5^2\alpha_6^2$$
$$+\alpha_1^2\alpha_3^2\alpha_2^2\alpha_4^2+\alpha_1^2\alpha_3^2\alpha_5^2\alpha_6^2+\alpha_2^2\alpha_4^2\alpha_5^2\alpha_6^2)-40\alpha_1\alpha_2\alpha_3\alpha_4\alpha_5\alpha_6+1\ ,$$

$$G = \alpha_2^3\alpha_4^3 - \alpha_1^3\alpha_3^3 - \alpha_5^3\alpha_6^3 + \alpha_1^2\alpha_3^2 + \alpha_5^2\alpha_6^2 - 3\alpha_2^2\alpha_4^2+\alpha_1\alpha_3+\alpha_5\alpha_6+3\alpha_2\alpha_4+2\alpha_1\alpha_2\alpha_3\alpha_4$$
$$+2\alpha_2\alpha_4\alpha_5\alpha_6-10\alpha_1\alpha_3\alpha_5\alpha_6+2\alpha_1\alpha_2\alpha_3\alpha_4\alpha_5\alpha_6+\alpha_1\alpha_3\alpha_5^2\alpha_6^2 + 3\alpha_1^2\alpha_3^2\alpha_2\alpha_4+3\alpha_2\alpha_4\alpha_5^2\alpha_6^2$$
$$-3\alpha_2^2\alpha_4^2\alpha_5\alpha_6- 3\alpha_1\alpha_3\alpha_2^2\alpha_4^2 + \alpha_1^2\alpha_3^2\alpha_5\alpha_6-1\ ,$$

$$H = \alpha_2^2\alpha_4^2 + \alpha_1^2\alpha_3^2 - 3\alpha_5^2\alpha_6^2+2(\alpha_1\alpha_3\alpha_5\alpha_6+\alpha_2\alpha_4\alpha_5\alpha_6-\alpha_1\alpha_3\alpha_2\alpha_4-\alpha_1\alpha_3-\alpha_2\alpha_4+\alpha_5\alpha_6)+1,$$

$$L = \alpha_2^2\alpha_4^2+\alpha_1^2\alpha_3^2+\alpha_5^2\alpha_6^2+2(\alpha_1\alpha_3+\alpha_2\alpha_4+\alpha_5\alpha_6-\alpha_1\alpha_3\alpha_2\alpha_4-\alpha_1\alpha_3\alpha_5\alpha_6-\alpha_2\alpha_4\alpha_5\alpha_6)-3,$$

$$V = \alpha_2^2\alpha_4^2+\alpha_5^2\alpha_6^2-3\alpha_1^2\alpha_3^2+2(\alpha_1\alpha_3\alpha_5\alpha_6+\alpha_1\alpha_3\alpha_2\alpha_4-\alpha_2\alpha_4\alpha_5\alpha_6+\alpha_1\alpha_3-\alpha_2\alpha_4-\alpha_5\alpha_6)+1\ ,$$

$$F = \alpha_1^3\alpha_3^3+\alpha_2^3\alpha_4^3+\alpha_5^3\alpha_6^3-3(\alpha_1^2\alpha_3^2+\alpha_2^2\alpha_4^2+\alpha_5^2\alpha_6^2-\alpha_1\alpha_3-\alpha_2\alpha_4-\alpha_5\alpha_6)-2(\alpha_1\alpha_2\alpha_3\alpha_4+\alpha_1\alpha_3\alpha_5\alpha_6$$
$$+\alpha_2\alpha_4\alpha_5\alpha_6)+10\alpha_1\alpha_2\alpha_3\alpha_4\alpha_5\alpha_6-\alpha_1\alpha_3\alpha_2^2\alpha_4^2-\alpha_1^2\alpha_3^2\alpha_2\alpha_4-\alpha_1\alpha_3\alpha_5^2\alpha_6^2-\alpha_1^2\alpha_3^2\alpha_5\alpha_6$$
$$-\alpha_2\alpha_4\alpha_5^2\alpha_6^2 -\alpha_2^2\alpha_4^2\alpha_5\alpha_6-1\ ,$$

$$E = \alpha_2^3\alpha_4^3-\alpha_1^3\alpha_3^3+\alpha_5^3\alpha_6^3+3\alpha_1^2\alpha_3^2-\alpha_2^2\alpha_4^2-\alpha_5^2\alpha_6^2- 3\alpha_1\alpha_3-\alpha_2\alpha_4-\alpha_5\alpha_6-2\alpha_1\alpha_2\alpha_3\alpha_4-2\alpha_1\alpha_3\alpha_5\alpha_6$$
$$+10\alpha_2\alpha_4\alpha_5\alpha_6-2\alpha_1\alpha_2\alpha_3\alpha_4\alpha_5\alpha_6-3(\alpha_1\alpha_3\alpha_2^2\alpha_4^2+\alpha_1\alpha_3\alpha_5^2\alpha_6^2-\alpha_1^2\alpha_3^2\alpha_2\alpha_4-\alpha_1^2\alpha_3^2\alpha_5\alpha_6)-$$
$$\alpha_2\alpha_4\alpha_5^2\alpha_6^2-\alpha_2^2\alpha_4^2\alpha_5\alpha_6+1\ ,$$

$$M = 3\alpha_2^2\alpha_4^2-\alpha_1^2\alpha_3^2-\alpha_5^2\alpha_6^2+2(\alpha_1\alpha_3-\alpha_2\alpha_4+\alpha_5\alpha_6+\alpha_1\alpha_3\alpha_5\alpha_6-\alpha_1\alpha_3\alpha_2\alpha_4-\alpha_2\alpha_4\alpha_5\alpha_6)-1,$$

and,

$$Y = \alpha_1^3\alpha_3^3+\alpha_2^3\alpha_4^3-\alpha_5^3\alpha_6^3-\alpha_1^2\alpha_3^2-\alpha_2^2\alpha_4^2+3\alpha_5^2\alpha_6^2-\alpha_1\alpha_3-\alpha_2\alpha_4-3\alpha_5\alpha_6+10\alpha_1\alpha_2\alpha_3\alpha_4-2(\alpha_1\alpha_3\alpha_5\alpha_6$$
$$+\alpha_2\alpha_4\alpha_5\alpha_6+\alpha_1\alpha_2\alpha_3\alpha_4\alpha_5\alpha_6)+3(\alpha_1\alpha_3\alpha_5^2\alpha_6^2+\alpha_2\alpha_4\alpha_5^2\alpha_6^2-\alpha_1^2\alpha_3^2\alpha_5\alpha_6-\alpha_2^2\alpha_4^2\alpha_5\alpha_6)-$$
$$\alpha_1\alpha_3\alpha_2^2\alpha_4^2 -\alpha_1^2\alpha_3^2\alpha_2\alpha_4)+1. \qquad (4.5.14)$$

Hence, the inverse of R_0 can be written as,

$$R_0^{-1} = \frac{1}{K}\left[\begin{array}{cccc|cccc}
-F & -\alpha_3 E & -2\alpha_3\alpha_6 M & -\alpha_6 Y & \alpha_4 G & 2\alpha_3\alpha_4 H & 2\alpha_3\alpha_4\alpha_6 L & 2\alpha_4\alpha_6 V \\
-\alpha_1 E & -F & -\alpha_6 Y & -2\alpha_1\alpha_6 M & 2\alpha_1\alpha_4 H & \alpha_4 G & 2\alpha_4\alpha_6 V & 2\alpha_1\alpha_4\alpha_6 L \\
-2\alpha_1\alpha_5 M & -\alpha_5 Y & -F & -\alpha_1 E & 2\alpha_1\alpha_4\alpha_5 L & 2\alpha_4\alpha_5 V & \alpha_4 G & 2\alpha_1\alpha_4 H \\
-\alpha_5 Y & -2\alpha_3\alpha_5 M & -\alpha_3 E & -F & 2\alpha_4\alpha_5 V & 2\alpha_3\alpha_4\alpha_5 L & 2\alpha_3\alpha_4 H & \alpha_4 G \\
\hline
\alpha_2 G & 2\alpha_2\alpha_3 H & 2\alpha_2\alpha_3\alpha_6 L & 2\alpha_2\alpha_6 V & -F & -\alpha_3 E & -2\alpha_3\alpha_6 M & -\alpha_6 Y \\
2\alpha_1\alpha_2 H & \alpha_2 G & 2\alpha_2\alpha_6 V & 2\alpha_1\alpha_2\alpha_6 L & -\alpha_1 E & -F & -\alpha_6 Y & -2\alpha_1\alpha_6 M \\
2\alpha_1\alpha_2\alpha_5 L & 2\alpha_2\alpha_5 V & \alpha_2 G & 2\alpha_1\alpha_2 H & -2\alpha_1\alpha_5 M & -\alpha_5 Y & -F & -\alpha_1 E \\
2\alpha_2\alpha_5 V & 2\alpha_2\alpha_3\alpha_5 L & 2\alpha_2\alpha_3 H & \alpha_2 G & -\alpha_5 Y & -2\alpha_3\alpha_5 M & -\alpha_3 E & -F
\end{array}\right]$$

and (4.5.15)

$$R_0^{-1}R_1 = \frac{1}{K}\begin{bmatrix}
0 & -\alpha_1 F & -\alpha_1\alpha_6 Y & 0 & 0 & \alpha_1\alpha_4 G & 2\alpha_1\alpha_4\alpha_6 V & 0 \\
0 & -\alpha_1^2 E & -2\alpha_1^2\alpha_6 M & 0 & 0 & 2\alpha_1^2\alpha_4 H & 2\alpha_1^2\alpha_4\alpha_6 L & 0 \\
0 & -2\alpha_1^2\alpha_5 M & -\alpha_1^2 E & 0 & 0 & 2\alpha_1^2\alpha_4\alpha_5 L & 2\alpha_1^2\alpha_4 H & 0 \\
0 & -\alpha_1\alpha_5 Y & -\alpha_1 F & 0 & 0 & 2\alpha_1\alpha_4\alpha_5 V & \alpha_1\alpha_4 G & 0 \\
0 & \alpha_1\alpha_2 G & 2\alpha_1\alpha_2\alpha_6 V & 0 & 0 & -\alpha_1 F & -\alpha_1\alpha_6 Y & 0 \\
0 & 2\alpha_1^2\alpha_2 H & 2\alpha_1^2\alpha_2\alpha_6 L & 0 & 0 & -\alpha_1^2 E & -2\alpha_1^2\alpha_6 M & 0 \\
0 & 2\alpha_1^2\alpha_2\alpha_5 L & 2\alpha_1^2\alpha_2 H & 0 & 0 & -2\alpha_1^2\alpha_5 M & -\alpha_1^2 E & 0 \\
0 & 2\alpha_1\alpha_2\alpha_5 V & \alpha_1\alpha_2 G & 0 & 0 & -\alpha_1\alpha_5 Y & -\alpha_1 F & 0
\end{bmatrix} \quad (4.5.16)$$

For the model problem (4.5.1) and for,

$$\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = \alpha_5 = \alpha_6 = -\frac{1}{6} \ , \quad (4.5.17)$$

the values of K,G,...,Y can be obtained now, and they are given by,

$$K = \frac{1157625}{6^8} \ , \quad G = -\frac{40425}{6^6} \ , \quad H = \frac{1225}{6^4} \ ,$$

$$L = -\frac{3675}{6^4} \ , \quad V = H, \quad F = -\frac{35525}{6^6}, \quad (4.5.18)$$

$$E = -G, \quad M = -H \quad \text{and} \quad Y = -G.$$

Hence,

$$R_0^{-1}R_1 = -\frac{1}{315}\begin{pmatrix} 0 & 58 & 11 & 0 & 0 & 11 & 4 & 0 \\ 0 & 11 & 4 & 0 & 0 & 4 & 2 & 0 \\ 0 & 4 & 11 & 0 & 0 & 2 & 4 & 0 \\ 0 & 11 & 58 & 0 & 0 & 4 & 11 & 0 \\ 0 & 11 & 4 & 0 & 0 & 58 & 11 & 0 \\ 0 & 4 & 2 & 0 & 0 & 11 & 4 & 0 \\ 0 & 2 & 4 & 0 & 0 & 4 & 11 & 0 \\ 0 & 4 & 11 & 0 & 0 & 11 & 58 & 0 \end{pmatrix} \qquad (4.5.19)$$

Similarly, we can obtain $R_0^{-1}R_2, R_0^{-1}R_3, \ldots, R_0^{-1}R_6$, from which the computational molecule at the point P can be established, see Figure (4.5.4),

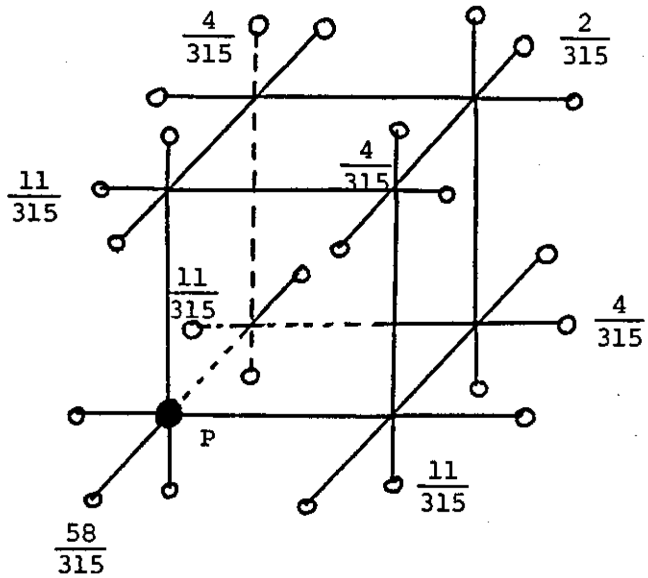


FIGURE (4.5.4)

From this molecule, an explicit group Jacobi iterative method can be derived and is given by,

$$\begin{aligned} u^{(k+1)}_{i,j,\ell} = \frac{1}{315}\Big[& 58\big(u^{(k)}_{i-1,j,\ell}+u^{(k)}_{i,j-1,\ell}+u^{(k)}_{i,j,\ell-1}\big)+11\big(u^{(k)}_{i+1,j-1,\ell}+u^{(k)}_{i+1,j,\ell-1} \\ & +u^{(k)}_{i+2,j,\ell}+u^{(k)}_{i-1,j+1,\ell}+u^{(k)}_{i,j+1,\ell-1}+u^{(k)}_{i,j+2,\ell}+u^{(k)}_{i-1,j,\ell+1}+u^{(k)}_{i,j-1,\ell+1}+ \\ & u^{(k)}_{i,j,\ell+2}\big)+4\big(u^{(k)}_{i+1,j+1,\ell-1}+u^{(k)}_{i+2,j+1,\ell}+u^{(k)}_{i+1,j+2,\ell}+u^{(k)}_{i+1,j-1,\ell+1} \\ & +u^{(k)}_{i+2,j,\ell+1}+u^{(k)}_{i+1,j,\ell+2}+u^{(k)}_{i-1,j+1,\ell+1}+u^{(k)}_{i,j+2,\ell+1}+ \\ & u^{(k)}_{i,j+1,\ell+2}\big)+2\big(u^{(k)}_{i+2,j+1,\ell+1}+u^{(k)}_{i+1,j+2,\ell+1}+u^{(k)}_{i+1,j+1,\ell+2}\big)\Big] \,, \end{aligned} \qquad (4.5.20)$$

Similarly, the other seven points are found from completely analogous expressions obtained by rotating the mesh points of Figure (4.5.4).

Now, to compare this group with the standard point S.O.R. iterative method for solving the model problem (4.5.1), we shall determine the computational complexity for each iteration from the point of view of arithmetic operations performed per iteration.

The Point S.O.R. Method

For the solution of the problem (4.5.1), using the 7-point finite difference formula, by the point S.O.R. method we have:

$$u_{i,j,\ell}^{(k+1)} = \omega(\tilde{u}_{i,j,\ell}^{(k+1)} - u_{i,j,\ell}^{(k)})+u_{i,j,\ell}^{(k)} \quad , \tag{4.5.21}$$

where,

$$\tilde{u}_{i,j,\ell}^{(k+1)} = (u_{i-1,j,\ell}^{(k+1)}+u_{i,j-1,\ell}^{(k+1)}+u_{i+1,j,\ell}^{(k)}+u_{i,j+1,\ell}^{(k)}+u_{i,j,\ell-1}^{(k+1)}+ u_{i,j,\ell+1}^{(k)})/6 \quad , \tag{4.5.22}$$

represents the Gauss-Seidel solution of the problem. Therefore, if we have m^3 internal mesh points, then the total number of operations per iteration will be:

$$2m^3 \text{ multiplications } + 7m^3 \text{ additions,} \tag{4.5.23}$$

assuming that the constant $\frac{1}{6}$ is stored beforehand.

The 8-Point Group S.O.R. Method

For this group, the Gauss-Seidel iterative method involves $\frac{m^3}{8}$ systems of equations of the form;

$$\left(\begin{array}{cccc|cccc} 6 & -1 & 0 & -1 & -1 & & & \\ -1 & 6 & -1 & 0 & & -1 & & \bigcirc \\ 0 & -1 & 6 & -1 & & \bigcirc & -1 & \\ -1 & 0 & -1 & 6 & & & & -1 \\ \hline -1 & & & & 6 & -1 & 0 & -1 \\ & -1 & \bigcirc & & -1 & 6 & -1 & 0 \\ & & -1 & & 0 & -1 & 6 & -1 \\ \bigcirc & & & -1 & -1 & 0 & -1 & 6 \end{array}\right) \left(\begin{array}{c} \tilde{u}^{(k+1)}_{i,j,\ell} \\ \tilde{u}^{(k+1)}_{i+1,j,\ell} \\ \tilde{u}^{(k+1)}_{i+1,j,\ell+1} \\ \tilde{u}^{(k+1)}_{i,j,\ell+1} \\ \hline \tilde{u}^{(k+1)}_{i,j+1,\ell} \\ \tilde{u}^{(k+1)}_{i+1,j+1,\ell} \\ \tilde{u}^{(k+1)}_{i+1,j+1,\ell+1} \\ \tilde{u}^{(k+1)}_{i,j+1,\ell+1} \end{array}\right) = \left(\begin{array}{c} b_1 \\ b_2 \\ b_3 \\ b_4 \\ \hline b_5 \\ b_6 \\ b_7 \\ b_8 \end{array}\right) \qquad (4.5.24)$$

$$i,j,\ell=1,(2),(m-1),$$

where,

$$b_1 = u^{(k+1)}_{i-1,j,\ell}+u^{(k+1)}_{i,j-1,\ell}+u^{(k+1)}_{i,j,\ell-1},\ b_2 = u^{(k+1)}_{i+1,j-1,\ell}+u^{(k)}_{i+2,j,\ell}+u^{(k+1)}_{i+1,j,\ell-1},$$

$$b_3 = u^{(k+1)}_{i+1,j-1,\ell+1}+u^{(k)}_{i+2,j,\ell+1}+u^{(k)}_{i+1,j,\ell+2},\ b_4 = u^{(k+1)}_{i-1,j,\ell+1}+u^{(k+1)}_{i,j-1,\ell+1}+ u^{(k)}_{i,j,\ell+2},$$

$$b_5 = u^{(k+1)}_{i-1,j+1,\ell}+u^{(k)}_{i,j+2,\ell}+u^{(k+1)}_{i,j+1,\ell-1},\ b_6 = u^{(k)}_{i+2,j+1,\ell}+u^{(k)}_{i+1,j+2,\ell}+ u^{(k+1)}_{i+1,j+1,\ell-1},$$

$$b_7 = u^{(k)}_{i+2,j+1,\ell+1}+u^{(k)}_{i+1,j+2,\ell+1}+u^{(k)}_{i+1,j+1,\ell+2},\ \text{and}$$

$$b_8 = u^{(k+1)}_{i-1,j+1,\ell+1}+u^{(k)}_{i,j+2,\ell+1}+u^{(k)}_{i,j+1,\ell+2}. \qquad (4.5.25)$$

Now, if we set,

$$\begin{aligned} &s_1 = b_2 + b_4, \quad s_2 = b_1 + b_3, \quad s_3 = b_6 + b_8, \\ &s_4 = b_5 + b_7, \quad s_5 = b_5 + s_1, \quad s_6 = b_6 + s_2, \\ &s_7 = b_7 + s_1, \quad s_8 = b_8 + s_2, \quad s_9 = b_1 + s_3, \\ &s_{10} = b_2 + s_4, \quad s_{11} = b_3 + s_3 \text{ and } s_{12} = b_4 + s_4. \end{aligned} \qquad (4.5.26)$$

Then, we have the Gauss-Seidel solution given by,

$$\tilde{u}^{(k+1)}_{i,j,\ell} = \frac{1}{315}(58b_1+11s_5+4s_{11}+2b_7),$$

$$\tilde{u}^{(k+1)}_{i+1,j,\ell} = \frac{1}{315}(58b_2 + 11s_6+4s_{12}+2b_8),$$

$$\tilde{u}^{(k+1)}_{i+1,j,\ell+1} = \frac{1}{315}(58b_3+11s_7+4s_9+2b_5) ,$$

$$\tilde{u}^{(k+1)}_{i,j,\ell+1} = \frac{1}{315}(58b_4+11s_8+4s_{10}+2b_6) ,$$

$$\tilde{u}^{(k+1)}_{i,j+1,\ell} = \frac{1}{315}(58b_5+11s_9+4s_7+2b_3) ,$$

$$\tilde{u}^{(k+1)}_{i+1,j+1,\ell} = \frac{1}{315}(58b_6+11s_{10}+4s_8+2b_4) , \qquad (4.5.27)$$

$$\tilde{u}^{(k+1)}_{i+1,j+1,\ell+1} = \frac{1}{315}(58b_7+11s_{11}+4s_5+2b_1) ,$$

and

$$\tilde{u}^{(k+1)}_{i,j+1,\ell+1} = \frac{1}{315}(58b_8+11s_{12}+4s_6+2b_2) .$$

By the application of the over-relaxation technique, we have the following formula,

$$u^{(k+1)}_{i+n_1,j+n_2,\ell+n_3} = \omega(\tilde{u}^{(k+1)}_{i+n_1,j+n_2,\ell+n_3} - u^{(k)}_{i+n_1,j+n_2,\ell+n_3})+u^{(k)}_{i+n_1,j+n_2,\ell+n_3}$$

for $i,j,\ell=1,(2),m-1$ and $n_1,n_2,n_3=0,1$. (4.5.28)

Hence, for m^3 internal mesh points, the amount of work required per iteration assuming that the constant $\frac{1}{315}$ is stored beforehand, is given by,

$$5m^3 \text{ multiplications} + \frac{17}{2} m^3 \text{ additions} , \qquad (4.5.29)$$

Further savings in work can be made by solving $\tilde{u}^{(k+1)}_{i,j,\ell}$, $\tilde{u}^{(k+1)}_{i+1,j,\ell+1}$, $\tilde{u}^{(k+1)}_{i+1,j+1,\ell}$ and $\tilde{u}^{(k+1)}_{i,j+1,\ell+1}$ first using equation (4.5.27), then calculating the other four points by applying the 7-point finite difference formula.

So, by setting,

$$s_{13} = \tilde{u}^{(k+1)}_{i,j,\ell} + \tilde{u}^{(k+1)}_{i+1,j,\ell+1} ,$$

and

$$s_{14} = \tilde{u}^{(k+1)}_{i+1,j+1,\ell} + \tilde{u}^{(k+1)}_{i,j+1,\ell+1} , \qquad (4.5.30)$$

then, we have,

$$\tilde{u}^{(k+1)}_{i+1,j,\ell} = (s_{13}+\tilde{u}^{(k+1)}_{i+1,j+1,\ell}+b_2)/6 ,$$

$$\tilde{u}^{(k+1)}_{i,j,\ell+1} = (s_{13}+\tilde{u}^{(k+1)}_{i,j+1,\ell+1}+b_4)/6 , \qquad (4.5.31)$$

$$\tilde{u}^{(k+1)}_{i,j+1,\ell} = (s_{14}+\tilde{u}^{(k+1)}_{i,j,\ell} + b_5)/6 ,$$

and

$$\tilde{u}^{(k+1)}_{i+1,j+1,\ell+1} = (s_{14}+\tilde{u}^{(k+1)}_{i+1,j,\ell+1}+b_7)/6 .$$

so the amount of work per iteration for this method including the over relaxation process, assuming that the constant $\frac{1}{6}$ is stored beforehand, is,

$$\frac{7}{2}m^3 \text{ multiplications} + \frac{33}{4}m^3 \text{ additions} , \tag{4.5.32}$$

Experimental Results

We present here some numerical experiments for the point and the 8-point group S.O.R. iterative method in 3-space dimensions, for solving Laplace's equation (4.5.1) in the unit cube subject to the boundary conditions,

$$\left.\begin{aligned} &U(0,y,z) = U(1,y,z) = 0 , && 0\leq y,z\leq 1 \\ &U(x,0,z) = U(x,1,z) = \sin\pi x.\sin\pi z , && 0\leq x,z\leq 1 \\ &U(x,y,0) = U(x,y,1) = 0 , && 0\leq x,y\leq 1 \end{aligned}\right\} \tag{4.5.33}$$

A red-black ordering of the points of the mesh and also of the groups has been used. Since the theory of block S.O.R. is valid, the optimum relaxation factor, ω_b, can be calculated using the formula,

$$\omega_b = \frac{2}{1+\sqrt{1-\{\rho(J)\}^2}} , \tag{4.5.34}$$

where $\rho(J)$ is the spectral radius of the Jacobi scheme and according to equation (4.3.4), the spectral radius for the 8-point group Jacobi scheme can be calculated using the formula,

$$\rho(J_{8\text{-point}}) \simeq 1 - \pi^2h^2 , \tag{4.5.35}$$

while for the point Jacobi method it is given by,

$$\begin{aligned} \rho(J_{\text{point}}) &= \frac{1}{3}(\cos ih\pi+\cos jh\pi+\cos \ell h\pi) \\ &= \cos h\pi . \end{aligned} \tag{4.5.36}$$

The convergence test used in the numerical experiment was the average test (see condition (4.3.5)), and $\varepsilon=5\times 10^{-5}$. The experimental values of ω are determined to within ± 0.001. Different mesh sizes were used to produce the $(6\times6\times6)$, $(12\times12\times12)$, $(16\times16\times16)$, $(22\times22\times22)$, $(26\times26\times26)$, $(32\times32\times32)$ and $(36\times36\times36)$ networks.

h^{-1}	$\rho(B)$	ω_b		No. of iterations (k)
		Theory	Experiment	
5	0.809017	1.26	1.254-1.27	11
11	0.959493	1.56	1.548-1.572	24
15	0.978148	1.656	1.646-1.658	32
21	0.988831	1.74	1.726-1.741	45
25	0.9921147	1.777	1.766-1.774	53
31	0.9948693	1.816	1.802-1.812	66
35	0.9959743	1.835	1.822-1.832	74

TABLE (4.5.1): Point S.O.R.

h^{-1}	$\rho(B)$	ω_b		No. of iterations (k)
		Theory	Experiment	
5	0.6052158	1.114	1.158-1.179	7
11	0.918433	1.433	1.458	14
15	0.9561351	1.547	1.56-1.566	19
21	0.9776199	1.652	1.662-1.668	26
25	0.9842086	1.70	1.705-1.711	31
31	0.9897299	1.75	1.759-1.761	37
35	0.9919432	1.775	1.78-1.787	42

TABLE (4.5.2): 8-Point Group S.O.R.

Hence, the result for the experimental number of iterations shown in Tables (4.5.1) and (4.5.2) can be combined with the number of operations in each iteration required by the two methods to give the total number of arithmetic operations required for a solution and these are recorded in Table (4.5.3).

h^{-1} \ Method	Point S.O.R.		8-Point Group S.O.R.	
	M	A	M	A
5	$22m^3$	$77m^3$	$24.5m^3$	$57.75m^3$
11	$48m^3$	$168m^3$	$49m^3$	$115.5m^3$
15	$64m^3$	$224m^3$	$66.5m^3$	$156.75m^3$
21	$90m^3$	$315m^3$	$91m^3$	$214.5m^3$
25	$106m^3$	$371m^3$	$108.5m^3$	$255.75m^3$
31	$132m^3$	$462m^3$	$129.5m^3$	$305.25m^3$
35	$148m^3$	$518m^3$	$147m^3$	$346.5m^3$

TABLE (4.5.3)

Again, the logarithm of the minimum number of iterations was plotted against $\log(h^{-1})$, where h the mesh size, the graphs which are shown in Figure (4.5.5) reveal that the plots for the two methods were straight lines with slope of unity thus supporting the S.O.R. theory.

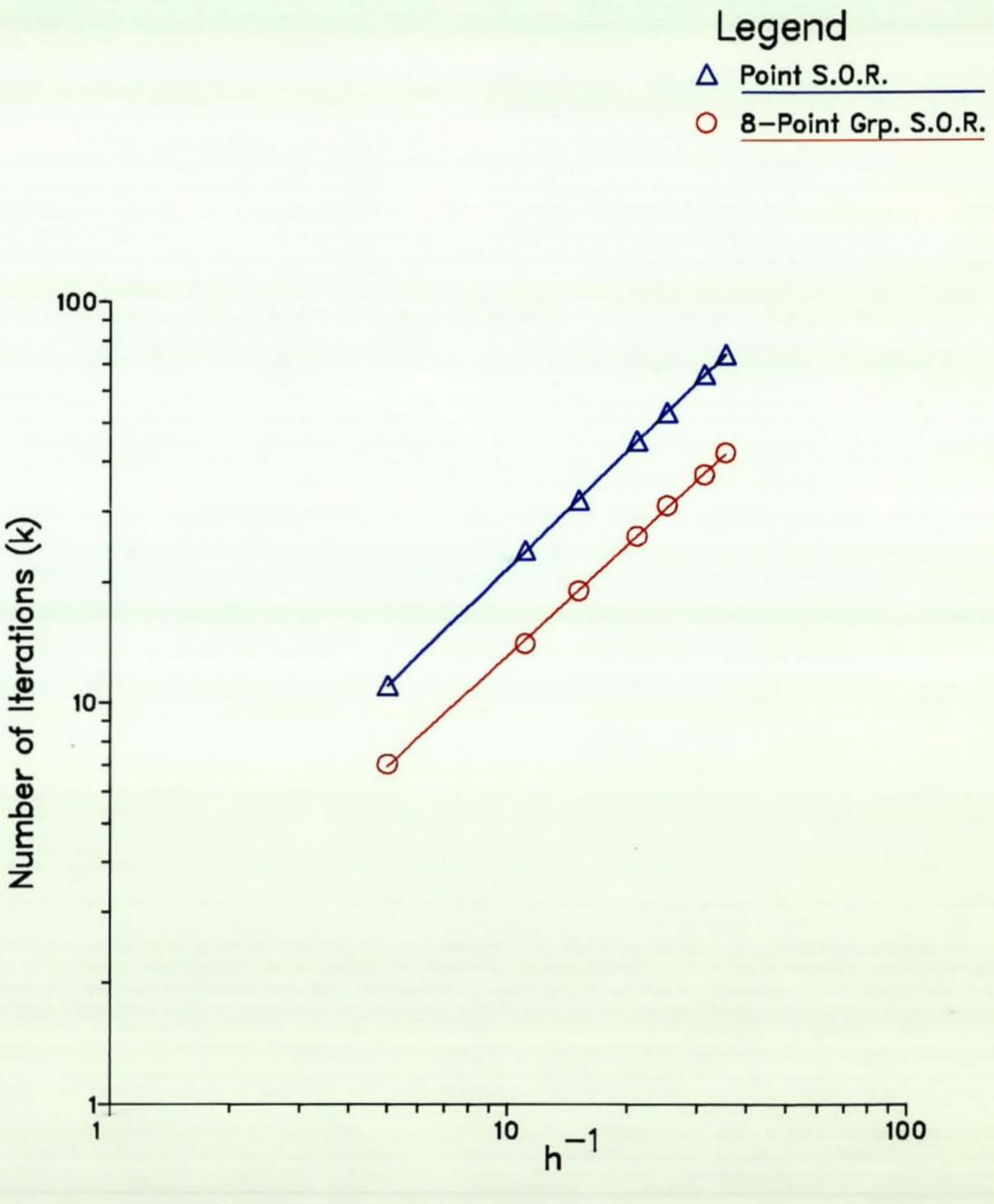

Legend
Point S.O.R.
8–Point Grp. S.O.R.
100
10
1
Number of Iterations (k)
1
10
100
h^{-1}

Figure (4.5.5)

4.6 CONCLUDLING REMARKS

The numerical experiments carried out on the solution of the model problem of elliptic partial differential equations using the standard point and block S.O.R. methods (the 1-line and 2-line S.O.R.) and also by implementing the novel approach of using a group of points of fixed size, i.e., the 2-point group, 4-point group, etc., lead to the following remarks on the theoretical and computational efficiency of the above methods.

From the fact that the 2-line S.O.R. method has a greater rate of convergence than the point and the 1-line S.O.R. methods together with the result given in Tables (4.3.11) and (4.3.12a) indicates that the 2-line S.O.R. algorithm is the most efficient within this class of algorithms.

On the subject of the novel group iterative methods we are faced with a wider set of alternatives depending upon the number of the points chosen to form a group. This at the same time makes it more difficult to make the final decision of selecting the grouping which will result in the least amount of computational work, minimum complexity, cost and highest programming flexibility.

However, our analysis of the group iterative methods indicate that the 4,9 and 25-point grouping of the mesh points are more efficient than any other alternative grouping in this class of algorithms. One of the main reasons behind the above fact is the inherent symmetry which the 4,9 and 25-point group produce which enables more efficient manipulation of the algorithm by reducing the multiplicative operations required to solve the problem.

Further, it can be noted from Table (4.3.11), where the execution times for each method are recorded, then the 9-point group method appears to be the most efficient.

Finally comparing the above two alternative classes of algorithms, (i.e. block and group S.O.R. iterative methods) for solving p.d.e.'s of elliptic type, a comparative discussion can be given as follows, the group iterative method is simpler to program than the classic block methods and it requires less storage, whereas the usual methods use large arrays in solving the line block of equations.

For the solution of the elliptic p.d.e. in 3-space dimensions, from the results given in Tables (4.5.1) and (4.5.2) together with the result shown in Table (4.5.3), we can indicate that the 8-point group S.O.R. iterative method is more efficient than the point S.O.R. iterative method in 3-space dimensions.

CHAPTER 5

THE IMPLICIT BLOCK-EXPLICIT BLOCK ITERATIVE METHODS

5.1 INTRODUCTION

This chapter presents a new implicit block, explicit block (IBEB) composite iterative technique for the solution of Laplace's equation in the unit square. The two methods are presented for the 2-point, 1-line and the 4-point, 2-line groupings.

The technique used is that of solving the 2×1 point block (or the 2×2 point block) explicitly and then group the new explicit point equations along a single line (or 2 lines) and solve implicitly using line algorithms. The two methods are introduced in Sections (5.2) and (5.3), the numerical experiments are carried out and the results given in Section (5.5).

5.2 THE 2-POINT, 1-LINE ITERATIVE METHOD

Consider the linear system (3.11.1) defined in the unit square $0 \leq x,y \leq 1$ with m^2 internal mesh points in the region shown in Fig. (4.2.1). By using the techniques of section (4.2.1) and by applying equation (4.2.1) to the points (i,j) and $(i+1,j)$ of Fig. (5.2.1),

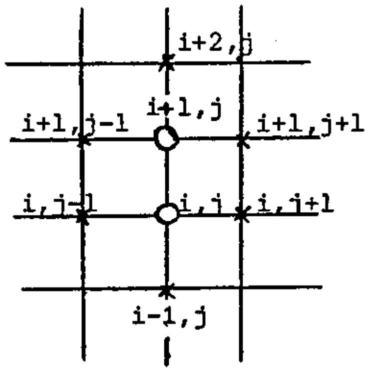


FIGURE 5.2.1

we obtain the $\sim(k+1)^{th}$ iterates of the Gauss-Seidel iterative scheme which is defined by

$$\left.\begin{aligned} \tilde{u}^{(k+1)}_{i,j} &= \frac{1}{\alpha_5}\Big(-\alpha_4\tilde{u}^{(k+1)}_{i,j-1}-\alpha_1\tilde{u}^{(k+1)}_{i-1,j}-\alpha_2 u^{(k)}_{i,j+1}+\alpha_2\alpha_3 u^{(k)}_{i+1,j+1}+\alpha_3^2 u^{(k)}_{i+2,j}+ \\ &\qquad +\alpha_3\alpha_4\tilde{u}^{(k+1)}_{i+1,j-1}\Big)\,, \\ \text{and}\quad \tilde{u}^{(k+1)}_{i+1,j} &= \frac{1}{\alpha_5}\Big(-\alpha_2 u^{(k)}_{i+1,j+1}-\alpha_3 u^{(k)}_{i+2,j}-\alpha_4\tilde{u}^{(k+1)}_{i+1,j-1}+\alpha_1\alpha_4\tilde{u}^{(k+1)}_{i,j-1}+\alpha_1^2\tilde{u}^{(k+1)}_{i-1,j}+ \\ &\qquad +\alpha_1\alpha_2 u^{(k)}_{i,j+1}\Big)\,, \end{aligned}\right\} \quad (5.2.1)$$

where $\alpha_5=1-\alpha_1\alpha_3$.

Now, by applying the extrapolation process, the $\wedge(k+1)^{th}$ iterates of the group S.O.R. iterative scheme can be obtained and is given by,

$$\left.\begin{aligned} \hat{u}^{(k+1)}_{i,j} &= \omega_1(\tilde{u}^{(k+1)}_{i,j}-u^{(k)}_{i,j}) + u^{(k)}_{i,j}\,, \\ \hat{u}^{(k+1)}_{i+1,j} &= \omega_1(\tilde{u}^{(k+1)}_{i+1,j}-u^{(k)}_{i+1,j})+u^{(k)}_{i+1,j}\,, \end{aligned}\right\} \quad (5.2.2)$$

where ω_1 is the first relaxation factor. By substituting the equivalence of $\tilde{u}^{(k+1)}_{i,j}$ and $\tilde{u}^{(k+1)}_{i+1,j}$ from equation (5.2.1) in equation (5.2.2), we have

$$\frac{1}{\alpha_5}[\alpha_4\tilde{u}^{(k+1)}_{i,j-1}-\alpha_3\alpha_4\tilde{u}^{(k+1)}_{i+1,j-1}+(\alpha_1\tilde{u}^{(k+1)}_{i-1,j}+\frac{\alpha_5}{\omega_1}\hat{u}^{(k+1)}_{i,j}-\alpha_3^2u^{(k)}_{i+2,j})+\alpha_2u^{(k)}_{i,j+1}-$$
$$-\alpha_2\alpha_3u^{(k)}_{i+1,j+1}] = (\frac{1}{\omega_1}-1)u^{(k)}_{i,j} ,$$

and

$$\frac{1}{\alpha_5}[-\alpha_1\alpha_4\tilde{u}^{(k+1)}_{i,j-1}+\alpha_4\tilde{u}^{(k+1)}_{i+1,j-1}+(-\alpha_1^2\tilde{u}^{(k+1)}_{i-1,j}+\frac{\alpha_5}{\omega_1}\hat{u}^{(k+1)}_{i+1,j}+\alpha_3u^{(k)}_{i+2,j})-\alpha_1\alpha_2u^{(k)}_{i,j+1}+$$
$$+\alpha_2u^{(k)}_{i+1,j+1}] = (\frac{1}{\omega_1}-1)u^{(k)}_{i+1,j} . \qquad (5.2.3)$$

Now, from a columnwise ordering, we group the points 1,2;3,4;...;m-1,m along the 1st column of the grid, m+1,m+2;...;2m-1,2m along the 2nd column etc. as shown in Fig. (5.2.2), where m is even.

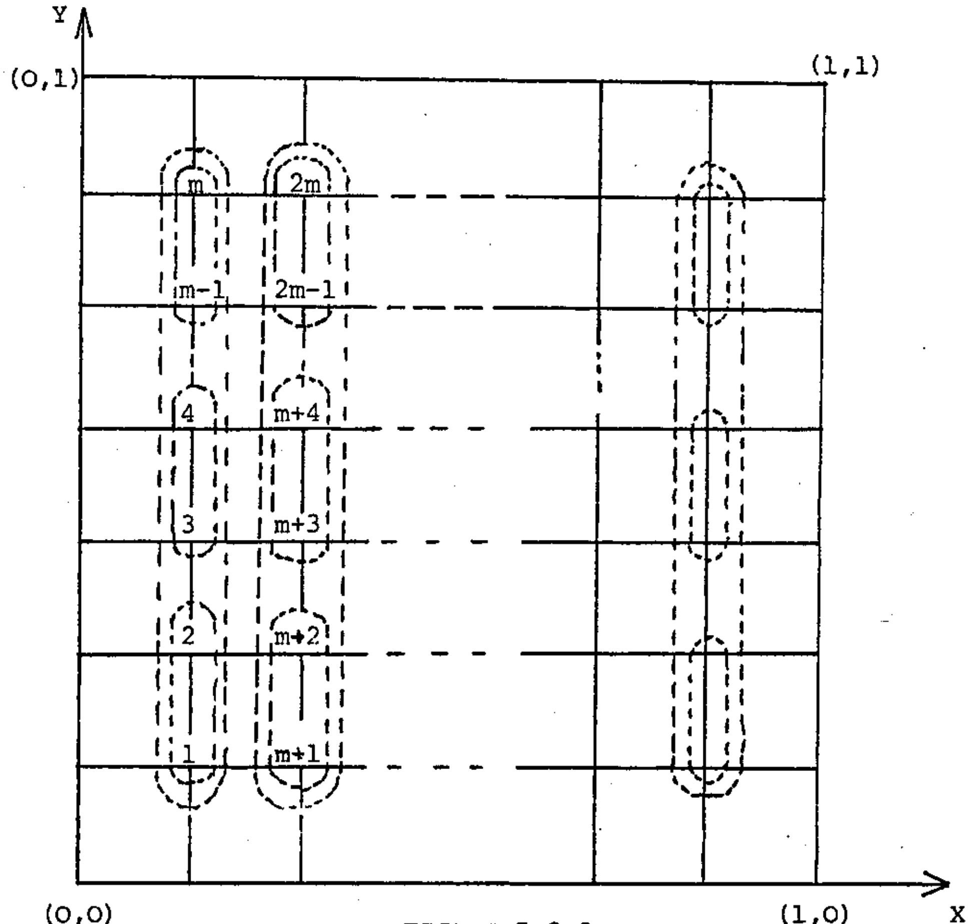


FIGURE 5.2.2

Hence, the resulting coefficient matrix is block tridiagonal where each diagonal submatrix is a quindiagonal matrix, and the system can be written as,

$$\begin{bmatrix} R_0 & R_2 & & & \\ R_1 & R_0 & R_2 & & O \\ & \ddots & \ddots & \ddots & \\ O & & R_1 & R_0 & R_2 \\ & & & R_1 & R_0 \end{bmatrix} \begin{bmatrix} \underline{U}_1 \\ \underline{U}_2 \\ \vdots \\ \vdots \\ \underline{U}_m \end{bmatrix} = \begin{bmatrix} \underline{B}_1 \\ \underline{B}_2 \\ \vdots \\ \vdots \\ \underline{B}_m \end{bmatrix} \qquad (5.2.4)$$

where

$$R_0 = \left[\begin{array}{cc|cc|cc|cc} \alpha_5 & 0 & -\alpha_3^2\omega_1 & 0 & & & & \\ 0 & \alpha_5 & \alpha_3\omega_1 & 0 & & & & \\ \hline 0 & \alpha_1\omega_1 & \alpha_5 & 0 & -\alpha_3^2\omega_1 & 0 & & \\ 0 & -\alpha_1^2\omega_1 & 0 & \alpha_5 & \alpha_3\omega_1 & 0 & & \\ \hline & & \ddots & & \ddots & & \ddots & \\ \hline & & & & & & -\alpha_3^2\omega_1 & 0 \\ & & & & & & \alpha_3\omega_1 & 0 \\ \hline & & & & 0 & \alpha_1\omega_1 & \alpha_5 & 0 \\ & & & & 0 & -\alpha_1^2\omega_1 & 0 & \alpha_5 \end{array}\right] \qquad (5.2.5a)$$

$$R_1 = \left[\begin{array}{cc|cc|c|cc} \alpha_4\omega_1 & -\alpha_3\alpha_4\omega_1 & & & & & \\ -\alpha_1\alpha_4\omega_1 & \alpha_4\omega_1 & & & & & \\ \hline & & \alpha_4\omega_1 & -\alpha_3\alpha_4\omega_1 & & & \\ & & -\alpha_1\alpha_4\omega_1 & \alpha_4\omega_1 & & & \\ \hline & & & & \ddots & & \\ \hline & & & & & \alpha_4\omega_1 & -\alpha_3\alpha_4\omega_1 \\ & & & & & -\alpha_1\alpha_4\omega_1 & \alpha_4\omega_1 \end{array}\right] \qquad (5.2.5b)$$

and

$$R_2 = \left[\begin{array}{cc|cc|c|cc}
\alpha_2\omega_1 & -\alpha_2\alpha_3\omega_1 & & & & & \\
-\alpha_1\alpha_2\omega_1 & \alpha_2\omega_1 & & & & \multicolumn{2}{c}{0} \\
\hline
 & & \alpha_2\omega_1 & -\alpha_2\alpha_3\omega_1 & & & \\
 & & -\alpha_1\alpha_2\omega_1 & \alpha_2\omega_1 & & & \\
\hline
 & & & & \ddots & & \\
\hline
 & \multicolumn{2}{c}{0} & & & \alpha_2\omega_1 & -\alpha_2\alpha_3\omega_1 \\
 & & & & & -\alpha_1\alpha_2\omega_1 & \alpha_2\omega_1
\end{array}\right] \tag{5.2.5c}$$

and $\underline{U}_r$ $(1\leq r\leq m)$ is the column partitioned vector of the values on the r^{th} line of the grid points.

For the model problem (3.2.1) and a square grid, $\alpha_1=\alpha_2=\alpha_3=\alpha_4=-\frac{1}{4}$, equations (5.2.3) can now be written as,

$$\left.\begin{aligned}
&-\omega_1[4\tilde{u}^{(k+1)}_{i,j-1}+\tilde{u}^{(k+1)}_{i+1,j-1}+(4\tilde{u}^{(k+1)}_{i-1,j}-\frac{15}{\omega_1}\hat{u}^{(k+1)}_{i,j}+u^{(k)}_{i+2,j})+4u^{(k)}_{i,j+1}+u^{(k)}_{i+1,j+1}] \\
&\qquad = 15(1-\omega_1)u^{(k)}_{i,j}\ , \\
&-\omega_1[\tilde{u}^{(k+1)}_{i,j-1}+4\tilde{u}^{(k+1)}_{i+1,j-1}+(\tilde{u}^{(k+1)}_{i-1,j}-\frac{15}{\omega_1}\hat{u}^{(k+1)}_{i+1,j}+4u^{(k)}_{i+2,j})+u^{(k)}_{i,j+1}+4u^{(k)}_{i+1,j+1}] \\
&\qquad = 15(1-\omega_1)u^{(k)}_{i+1,j}\ .
\end{aligned}\right\} \tag{5.2.6}$$

The matrix A will have the same block structure as given in (5.2.4) with

$$R_0 = \left[\begin{array}{cc|cc|cc|cc|cc}
15 & 0 & -\omega_1 & 0 & & & & & & \\
0 & 15 & -4\omega_1 & 0 & & & & & \multicolumn{2}{c}{0} \\
\hline
0 & -4\omega_1 & 15 & 0 & -\omega_1 & 0 & & & & \\
0 & -\omega_1 & 0 & 15 & -4\omega_1 & 0 & & & & \\
\hline
 & & \ddots & & \ddots & & \ddots & & & \\
\hline
 & & & & \ddots & & \ddots & & -\omega_1 & 0 \\
 & \multicolumn{2}{c}{0} & & & & & \ddots & -4\omega_1 & 0 \\
\hline
 & & & & & & 0 & -4\omega_1 & 15 & 0 \\
 & & & & & & 0 & -\omega_1 & 0 & 15
\end{array}\right] \tag{5.2.7a}$$

$$R_1 = R_2 = \left[\begin{array}{cc|cc|c|cc}
-4\omega_1 & -\omega_1 & & & & & \\
-\omega_1 & -4\omega_1 & & & & & \\
\hline
 & & -4\omega_1 & -\omega_1 & & 0 & \\
 & & -\omega_1 & -4\omega_1 & & & \\
\hline
 & & & & \ddots & & \\
\hline
 & 0 & & & & -4\omega_1 & -\omega_1 \\
 & & & & & -\omega_1 & -4\omega_1
\end{array}\right] \qquad (5.2.7b)$$

Now, the system (5.2.4) can be solved in a similar way as the line iterative method. First, subsystems of equations of the form,

$$R_1\tilde{\underline{U}}_{r-1}^{(k+1)} + R_0\hat{\underline{U}}_r^{(k+1)} + R_2\underline{U}_{r+1}^{(k)} = \underline{B}_r \ , \qquad (5.2.8)$$

have to be solved to evaluate $R_0\hat{\underline{U}}_r^{(k+1)}$, then by applying the algorithm given in Section (3.11) for solving a quindiagonal system, $\hat{\underline{U}}_r^{(k+1)}$ now can be determined by solving

$$R_0\hat{\underline{U}}_r^{(k+1)} = T_r \ , \ \text{(say)} \ , \ 1 \leq r \leq m, \qquad (5.2.9)$$

and by applying the extrapolation process once again, we have,

$$\underline{U}_r^{(k+1)} = \omega_2(\hat{\underline{U}}_r^{(k+1)} - \underline{U}_r^{(k)}) + \underline{U}_r^{(k)} \ , \qquad (5.2.10)$$

to obtain the final solution.

5.3 THE 4-POINT, 2-LINE ITERATIVE METHOD

Similar to the 2-point, 1-line iterative method, we now present the 4-point, 2-line iterative method. Consider the linear system (3.11.1) defined in the unit square $0 \leq x,y \leq 1$ with m^2 internal mesh points. We order the points in groups of 4 as shown in Fig. (5.3.1).

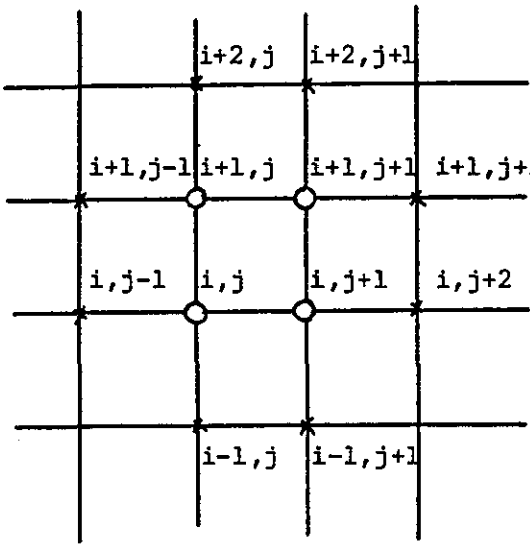


FIGURE 5.3.1

By applying equation (4.2.1) to the points (i,j), (i,j+1), (i+1,j) and (i+1,j+1) in turn, we obtain the $\sim(k+1)^{th}$ iterates of the Gauss-Seidel iterative scheme and is given by,

$$\tilde{u}_{i,j}^{(k+1)} = \frac{1}{d}\,[\alpha_5(-\alpha_1\tilde{u}_{i-1,j}^{(k+1)}-\alpha_4\tilde{u}_{i,j-1}^{(k+1)})+\alpha_2\alpha_6(-\alpha_1\tilde{u}_{i-1,j+1}^{(k+1)}-\alpha_2 u_{i,j+2}^{(k)})+$$
$$+\alpha_3\alpha_7(-\alpha_3 u_{i+2,j}^{(k)}-\alpha_4\tilde{u}_{i+1,j-1}^{(k+1)})+2\alpha_2\alpha_3(-\alpha_2 u_{i+1,j+2}^{(k)}-\alpha_3 u_{i+2,j+1}^{(k)})],$$

$$\tilde{u}_{i,j+1}^{(k+1)} = \frac{1}{d}[\alpha_4\alpha_6(-\alpha_1\tilde{u}_{i-1,j}^{(k+1)}-\alpha_4\tilde{u}_{i,j-1}^{(k+1)})+\alpha_5(-\alpha_1\tilde{u}_{i-1,j+1}^{(k+1)}-\alpha_2 u_{i,j+2}^{(k)})+2\alpha_3\alpha_4$$
$$(-\alpha_3 u_{i+2,j}^{(k)}-\alpha_4\tilde{u}_{i+1,j-1}^{(k+1)})+\alpha_3\alpha_7(-\alpha_2 u_{i+1,j+2}^{(k)}-\alpha_3 u_{i+2,j+1}^{(k)})]\;,$$

$$\tilde{u}_{i+1,j}^{(k+1)} = \frac{1}{d}[\alpha_1\alpha_7(-\alpha_1\tilde{u}_{i-1,j}^{(k+1)}-\alpha_4\tilde{u}_{i,j-1}^{(k+1)})+2\alpha_1\alpha_2(-\alpha_1\tilde{u}_{i-1,j+1}^{(k+1)}-\alpha_2 u_{i,j+2}^{(k)})+\alpha_5$$
$$(-\alpha_3 u_{i+2,j}^{(k)}-\alpha_4\tilde{u}_{i+1,j-1}^{(k+1)})+\alpha_2\alpha_6(-\alpha_2 u_{i+1,j+2}^{(k)}-\alpha_3 u_{i+2,j+1}^{(k)})]\;,$$

$$\tilde{u}^{(k+1)}_{i+1,j+1} = \frac{1}{d}\,[2\alpha_1\alpha_4(-\alpha_1\tilde{u}^{(k+1)}_{i-1,j}-\alpha_4\tilde{u}^{(k+1)}_{i,j-1})+\alpha_1\alpha_7(-\alpha_1\tilde{u}^{(k+1)}_{i-1,j+1}-\alpha_2 u^{(k)}_{i,j+2})$$
$$+\alpha_4\alpha_6(-\alpha_3 u^{(k)}_{i+2,j}-\alpha_4\tilde{u}^{(k+1)}_{i+1,j-1})+\alpha_5(-\alpha_2 u^{(k)}_{i+1,j+2}-\alpha_3 u^{(k)}_{i+2,j+1})] \tag{5.3.1}$$

where $d = (\alpha_6+1)^2+2\alpha_5-1,\quad \alpha_5 = 1-\alpha_2\alpha_4-\alpha_1\alpha_3,\quad \alpha_6 = \alpha_2\alpha_4-\alpha_1\alpha_3-1$,

and $\alpha_7 = \alpha_1\alpha_3-\alpha_2\alpha_4-1$. (5.3.2)

Hence, by applying the group S.O.R. iterative method to the $\sim(k+1)^{th}$ iterate and re-arranging the terms of equation (5.3.1) we have,

$$\frac{1}{d}[\alpha_4\alpha_5\tilde{u}^{(k+1)}_{i,j-1}+\alpha_3\alpha_4\alpha_7\tilde{u}^{(k+1)}_{i+1,j-1}+(\alpha_1\alpha_5\tilde{u}^{(k+1)}_{i-1,j}+\alpha_1\alpha_2\alpha_6\tilde{u}^{(k+1)}_{i-1,j+1} + \frac{d}{\omega_1}\hat{u}^{(k+1)}_{i,j} +$$
$$\alpha_3^2\alpha_7 u^{(k)}_{i+2,j}+2\alpha_2\alpha_3^2 u^{(k)}_{i+2,j+1})+\alpha_2^2\alpha_6 u^{(k)}_{i,j+2}+2\alpha_2^2\alpha_3 u^{(k)}_{i+1,j+2}] = (\frac{1}{\omega_1}-1)u^{(k)}_{i,j}\ ,$$

$$\frac{1}{d}[\alpha_4^2\alpha_6\tilde{u}^{(k+1)}_{i,j-1}+2\alpha_3\alpha_4^2\tilde{u}^{(k+1)}_{i+1,j-1}+(\alpha_1\alpha_4\alpha_6\tilde{u}^{(k+1)}_{i-1,j}+\alpha_1\alpha_5\tilde{u}^{(k+1)}_{i-1,j+1}+ \frac{d}{\omega_1}\hat{u}^{(k+1)}_{i,j+1} +$$
$$2\alpha_3^2\alpha_4 u^{(k)}_{i+2,j}+\alpha_3^2\alpha_7 u^{(k)}_{i+2,j+1})+\alpha_2\alpha_5 u^{(k)}_{i,j+2}+\alpha_2\alpha_3\alpha_7 u^{(k)}_{i+1,j+2}] = (\frac{1}{\omega_1}-1)u^{(k)}_{i,j+1},$$

$$\frac{1}{d}[\alpha_1\alpha_4\alpha_7\tilde{u}^{(k+1)}_{i,j-1}+\alpha_4\alpha_5\tilde{u}^{(k+1)}_{i+1,j-1}+(\alpha_1^2\alpha_7\tilde{u}^{(k+1)}_{i-1,j}+2\alpha_1^2\alpha_2\tilde{u}^{(k+1)}_{i-1,j+1} + \frac{d}{\omega_1}\hat{u}^{(k+1)}_{i+1,j}+$$
$$\alpha_3\alpha_5 u^{(k)}_{i+2,j}+\alpha_2\alpha_3\alpha_6 u^{(k)}_{i+2,j+1})+2\alpha_1\alpha_2^2 u^{(k)}_{i,j+2}+\alpha_2^2\alpha_6 u^{(k)}_{i+1,j+2}] = (\frac{1}{\omega_1}-1)u^{(k)}_{i+1,j},$$

and

$$\frac{1}{d}[2\alpha_1\alpha_4^2\tilde{u}^{(k+1)}_{i,j-1}+\alpha_4^2\alpha_6\tilde{u}^{(k+1)}_{i+1,j-1}+(2\alpha_1^2\alpha_4\tilde{u}^{(k+1)}_{i-1,j}+\alpha_1^2\alpha_7\tilde{u}^{(k+1)}_{i-1,j+1}+ \frac{d}{\omega_1}\hat{u}^{(k+1)}_{i+1,j+1}+$$
$$\alpha_3\alpha_4\alpha_6 u^{(k)}_{i+2,j}+\alpha_3\alpha_5 u^{(k)}_{i+2,j+1})+\alpha_1\alpha_2\alpha_7 u^{(k)}_{i,j+2}+\alpha_2\alpha_5 u^{(k)}_{i+1,j+2}] =$$
$$= (\frac{1}{\omega_1}-1)u^{(k)}_{i+1,j+1}\ , \tag{5.3.3}$$

where ω_1 is the first relaxation factor.

Then, we proceed further by grouping the points 1,2,3,4;5,6,7,8;...; 2m-3,2m-2,2m-1,2m along the 1st and 2nd columns of the grid, 2m+1,2m+2, 2m+3, 2m+4;...;4m-3,4m-2,4m-1,4m along the 3rd and 4th columns, etc., similar to the 2-line iterative method, see Fig.(5.3.2).

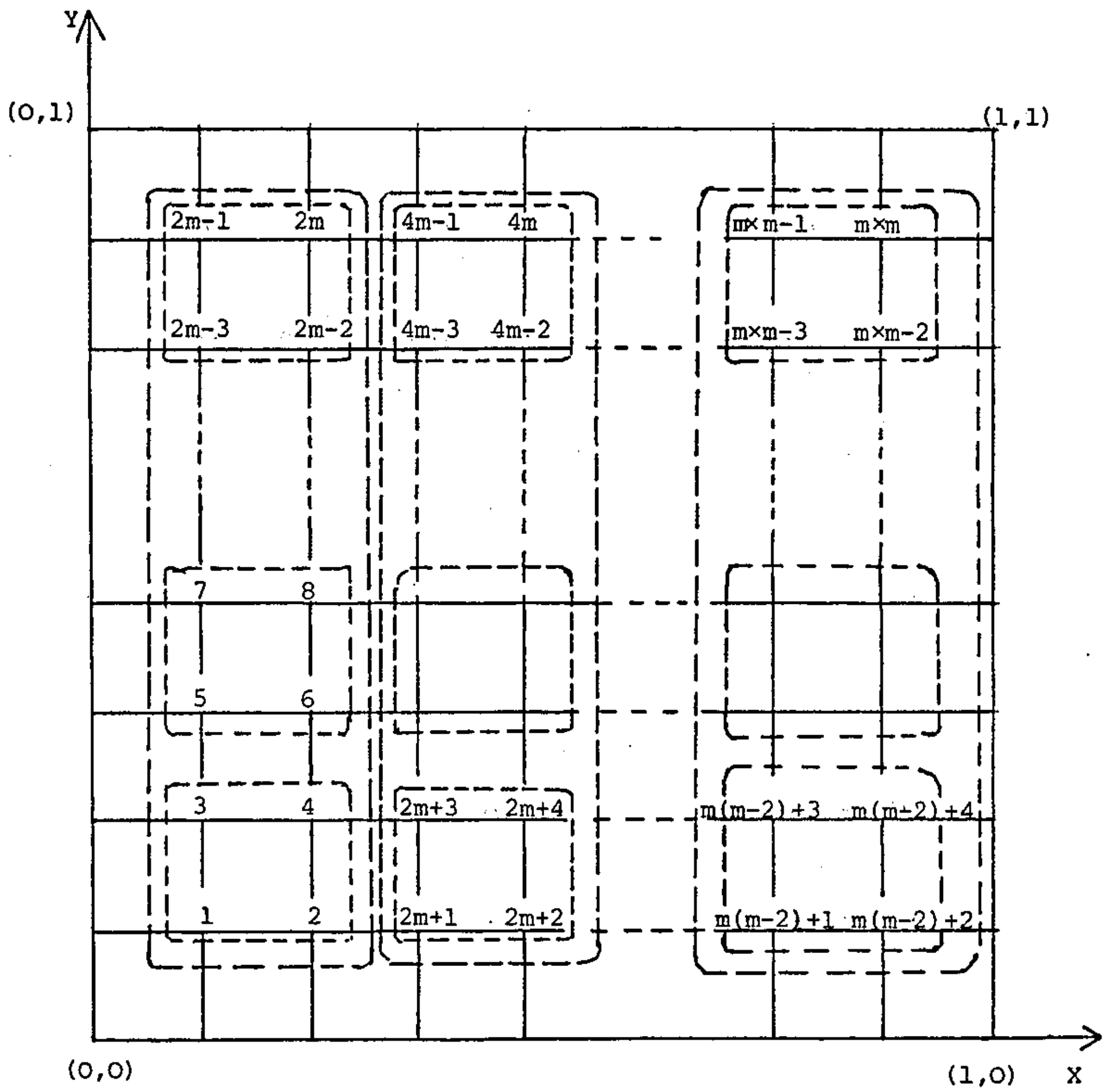


FIGURE 5.3.2

The resulting coefficient matrix A is of order m^2 and has the block structure

$$A = \begin{bmatrix} R_0 & R_2 & & & \\ R_1 & R_0 & R_2 & & \bigcirc \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & R_2 \\ & \bigcirc & & R_1 & R_0 \end{bmatrix} \qquad (5.3.4)$$

where the submatrices R_0, R_1, R_2 are square matrices of order 2m and possess the following form,

$$
R_0=\omega_1\left(\begin{array}{c|c|c|c}
\begin{matrix} \frac{d}{\omega_1} & & & \\ & \frac{d}{\omega_1} & 0 & \\ & 0 & \frac{d}{\omega_1} & \\ & & & \frac{d}{\omega_1} \end{matrix} &
\begin{matrix} \alpha_3^2\alpha_7 & 2\alpha_2\alpha_3^2 & 0 & 0 \\ 2\alpha_3^2\alpha_4 & \alpha_3^2\alpha_7 & 0 & 0 \\ \alpha_3\alpha_5 & \alpha_2\alpha_3\alpha_6 & 0 & 0 \\ \alpha_3\alpha_4\alpha_6 & \alpha_3\alpha_5 & 0 & 0 \end{matrix} &
 & 0 \\
\hline
\begin{matrix} 0 & 0 & \alpha_1\alpha_5 & \alpha_1\alpha_2\alpha_6 \\ 0 & 0 & \alpha_1\alpha_4\alpha_6 & \alpha_1\alpha_5 \\ 0 & 0 & \alpha_1^2\alpha_7 & 2\alpha_1^2\alpha_2 \\ 0 & 0 & 2\alpha_1^2\alpha_4 & \alpha_1^2\alpha_7 \end{matrix} &
\begin{matrix} \frac{d}{\omega_1} & & & \\ & \frac{d}{\omega_1} & 0 & \\ & 0 & \frac{d}{\omega_1} & \\ & & & \frac{d}{\omega_1} \end{matrix} &
\begin{matrix} \alpha_3^2\alpha_7 & 2\alpha_2\alpha_3^2 & 0 & 0 \\ 2\alpha_3^2\alpha_4 & \alpha_3^2\alpha_7 & 0 & 0 \\ \alpha_3\alpha_5 & \alpha_2\alpha_3\alpha_6 & 0 & 0 \\ \alpha_3\alpha_4\alpha_6 & \alpha_3\alpha_5 & 0 & 0 \end{matrix} &
 \\
\hline
 & \ddots & \ddots & \ddots \\
\hline
0 & &
\begin{matrix} 0 & 0 & \alpha_1\alpha_5 & \alpha_1\alpha_2\alpha_6 \\ 0 & 0 & \alpha_1\alpha_4\alpha_6 & \alpha_1\alpha_5 \\ 0 & 0 & \alpha_1^2\alpha_7 & 2\alpha_1^2\alpha_2 \\ 0 & 0 & 2\alpha_1^2\alpha_7 & \alpha_1^2\alpha_7 \end{matrix} &
\begin{matrix} \frac{d}{\omega_1} & & & 0 \\ 0 & \frac{d}{\omega_1} & & \\ & & \frac{d}{\omega_1} & \\ & & & \frac{d}{\omega_1} \end{matrix}
\end{array}\right), \quad (5.3.5a)
$$

$$R_1 = \omega_1 \left[\begin{array}{c|c|c} \begin{matrix} 0 & \alpha_4\alpha_5 & 0 & \alpha_3\alpha_4\alpha_7 \\ 0 & \alpha_4^2\alpha_6 & 0 & 2\alpha_3\alpha_4^2 \\ 0 & \alpha_1\alpha_4\alpha_7 & 0 & \alpha_4\alpha_5 \\ 0 & 2\alpha_1\alpha_4^2 & 0 & \alpha_4^2\alpha_6 \end{matrix} & O & O \\ \hline O & \ddots & O \\ \hline O & O & \begin{matrix} 0 & \alpha_4\alpha_5 & 0 & \alpha_3\alpha_4\alpha_7 \\ 0 & \alpha_4^2\alpha_6 & 0 & 2\alpha_3\alpha_4^2 \\ 0 & \alpha_1\alpha_4\alpha_7 & 0 & \alpha_4\alpha_5 \\ 0 & 2\alpha_1\alpha_4^2 & 0 & \alpha_4^2\alpha_6 \end{matrix} \end{array} \right] , \tag{5.3.5b}$$

and

$$R_2 = \omega_1 \left[\begin{array}{c|c|c} \begin{matrix} \alpha_2^2\alpha_6 & 0 & 2\alpha_2^2\alpha_3 & 0 \\ \alpha_2\alpha_5 & 0 & \alpha_2\alpha_3\alpha_7 & 0 \\ 2\alpha_1\alpha_2^2 & 0 & \alpha_2^2\alpha_6 & 0 \\ \alpha_1\alpha_2\alpha_7 & 0 & \alpha_2\alpha_5 & 0 \end{matrix} & O & O \\ \hline O & \ddots & O \\ \hline O & O & \begin{matrix} \alpha_2^2\alpha_6 & 0 & 2\alpha_2^2\alpha_3 & 0 \\ \alpha_2\alpha_5 & 0 & \alpha_2\alpha_3\alpha_7 & 0 \\ 2\alpha_1\alpha_2^2 & 0 & \alpha_2^2\alpha_6 & 0 \\ \alpha_1\alpha_2\alpha_7 & 0 & \alpha_2\alpha_5 & 0 \end{matrix} \end{array} \right] . \tag{5.3.5c}$$

Hence, the system,

$$A\underline{U} = \underline{B} , \tag{5.3.6}$$

where
$$\underline{U} = (\underline{U}_1, \underline{U}_2, \ldots, \underline{U}_{\frac{m}{2}})^T ,$$
$$\underline{B} = (\underline{B}_1, \underline{B}_2, \ldots, \underline{B}_{\frac{m}{2}})^T , \tag{5.3.7}$$

can be solved in a similar manner to that of solving the system (5.2.4) in which the subsystems of equations of the form given in (5.2.8), in this case $1 \leq r \leq \frac{m}{2}$, have to be solved to determine $R_0 \hat{\underline{U}}_r^{(k+1)}$, then for the evaluation of $\hat{\underline{U}}_r^{(k+1)}$ we solve,

$$R_0 \hat{\underline{U}}_r^{(k+1)} = T_r \text{ (say)}, \; 1 \leq r \leq \frac{m}{2} , \tag{5.3.8}$$

by applying the Block Gauss Jordan algorithm which will be presented in Section (5.4) for solving systems of equations involving block tridiagonal matrices. Then, by applying the extrapolation process the final solution is obtained as,

$$\underline{U}_r^{(k+1)} = \omega_2 (\hat{\underline{U}}_r^{(k+1)} - \underline{U}_r^{(k)}) + \underline{U}_r^{(k)} \quad . \tag{5.3.9}$$

For the model problem (3.2.1) and a square grid, $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = -\frac{1}{4}$, the equations given in (5.3.3) can now be written as,

$$-\omega_1 [7\tilde{u}_{i,j-1}^{(k+1)} + 2\tilde{u}_{i+1,j-1}^{(k+1)} + (7\tilde{u}_{i-1,j}^{(k+1)} + 2\tilde{u}_{i-1,j+1}^{(k+1)} - \frac{24}{\omega_1} \hat{u}_{i,j}^{(k+1)} + 2u_{i+2,j}^{(k)} + u_{i+2,j+1}^{(k)}) +$$
$$2u_{i,j+2}^{(k)} + u_{i+1,j+2}^{(k)}] = 24(1-\omega_1) u_{i,j}^{(k)} ,$$

$$-\omega_1 [2\tilde{u}_{i,j-1}^{(k+1)} + \tilde{u}_{i+1,j-1}^{(k+1)} + (2\tilde{u}_{i-1,j}^{(k+1)} + 7\tilde{u}_{i-1,j+1}^{(k+1)} - \frac{24}{\omega_1} \hat{u}_{i,j+1}^{(k+1)} + u_{i+2,j}^{(k)} + 2u_{i+2,j+1}^{(k)}) +$$
$$7u_{i,j+2}^{(k)} + 2u_{i+1,j+2}^{(k)}] = 24(1-\omega_1) u_{i,j+1}^{(k)} ,$$

$$-\omega_1 [2\tilde{u}_{i,j-1}^{(k+1)} + 7\tilde{u}_{i+1,j-1}^{(k+1)} + (2\tilde{u}_{i-1,j}^{(k+1)} + \tilde{u}_{i-1,j+1}^{(k+1)} - \frac{24}{\omega_1} \hat{u}_{i+1,j}^{(k+1)} + 7u_{i+2,j}^{(k)} + 2u_{i+2,j+1}^{(k)}) +$$
$$u_{i,j+2}^{(k)} + 2u_{i+1,j+2}^{(k)}] = 24(1-\omega_1) u_{i+1,j}^{(k)} ,$$

and

$$-\omega_1 [\tilde{u}_{i,j-1}^{(k+1)} + 2\tilde{u}_{i+1,j-1}^{(k+1)} + (\tilde{u}_{i-1,j}^{(k+1)} + 2\tilde{u}_{i-1,j+1}^{(k+1)} - \frac{24}{\omega_1} \hat{u}_{i+1,j+1}^{(k+1)} + 2u_{i+2,j}^{(k)} + 7u_{i+2,j+1}^{(k)}) +$$
$$2u_{i,j+2}^{(k)} + 7u_{i+1,j+2}^{(k)}] = 24(1-\omega_1) u_{i+1,j+1}^{(k)} . \tag{5.3.10}$$

For this problem, the coefficient matrix A has the same block structure as that given in (5.3.4), with,

$$
R_0 = \left[\begin{array}{c:c:c:c}
\begin{matrix} 24 & & & \\ & 24 & & O \\ O & & 24 & \\ & & & 24 \end{matrix} & \begin{matrix} -2\omega_1 & -\omega_1 & 0 & 0 \\ -\omega_1 & -2\omega_1 & 0 & 0 \\ -7\omega_1 & -2\omega_1 & 0 & 0 \\ -2\omega_1 & -7\omega_1 & 0 & 0 \end{matrix} & & O \\
\hdashline
\begin{matrix} 0 & 0 & -7\omega_1 & -2\omega_1 \\ 0 & 0 & -2\omega_1 & -7\omega_1 \\ 0 & 0 & -2\omega_1 & -\omega_1 \\ 0 & 0 & -\omega_1 & -2\omega_1 \end{matrix} & \begin{matrix} 24 & & & O \\ & 24 & & \\ O & & 24 & \\ & & & 24 \end{matrix} & \begin{matrix} -2\omega_1 & -\omega_1 & 0 & 0 \\ -\omega_1 & -2\omega_1 & 0 & 0 \\ -7\omega_1 & -2\omega_1 & 0 & 0 \\ -2\omega_1 & -7\omega_1 & 0 & 0 \end{matrix} & \\
\hdashline
 & \ddots & \ddots & \ddots \\
\hdashline
O & & \begin{matrix} 0 & 0 & -7\omega_1 & -2\omega_1 \\ 0 & 0 & -2\omega_1 & -7\omega_1 \\ 0 & 0 & -2\omega_1 & -\omega_1 \\ 0 & 0 & -\omega_1 & -2\omega_1 \end{matrix} & \begin{matrix} 24 & & & O \\ & 24 & & \\ O & & 24 & \\ & & & 24 \end{matrix}
\end{array}\right]
\tag{5.3.11a}
$$

$$
R_1 = \left[\begin{array}{c:c:c}
\begin{matrix} 0 & -7\omega_1 & 0 & -2\omega_1 \\ 0 & -2\omega_1 & 0 & -\omega_1 \\ 0 & -2\omega_1 & 0 & -7\omega_1 \\ 0 & -\omega_1 & 0 & -2\omega_1 \end{matrix} & O & O \\
\hdashline
O & \ddots & O \\
\hdashline
O & O & \begin{matrix} 0 & -7\omega_1 & 0 & -2\omega_1 \\ 0 & -2\omega_1 & 0 & -\omega_1 \\ 0 & -2\omega_1 & 0 & -7\omega_1 \\ 0 & -\omega_1 & 0 & -2\omega_1 \end{matrix}
\end{array}\right],
\tag{5.3.11b}
$$

and

$$R_2 = \left[\begin{array}{cccc|c|cccc}
-2\omega_1 & 0 & -\omega_1 & 0 & & & & & \\
-7\omega_1 & 0 & -2\omega_1 & 0 & O & & O & & \\
-\omega_1 & 0 & -2\omega_1 & 0 & & & & & \\
-2\omega_1 & 0 & -7\omega_1 & 0 & & & & & \\
\hline
 & O & & & \ddots & & O & & \\
\hline
 & & & & & -2\omega_1 & 0 & -\omega_1 & 0 \\
 & O & & & O & -7\omega_1 & 0 & -2\omega_1 & 0 \\
 & & & & & -\omega_1 & 0 & -2\omega_1 & 0 \\
 & & & & & -2\omega_1 & 0 & -7\omega_1 & 0
\end{array}\right] . \qquad (5.3.11c)$$

5.4 BASIC ALGORITHMS FOR SOLVING BLOCK TRIDIAGONAL SYSTEMS OF EQUATIONS

Algorithm 1

Gauss Jordan process for block tridiagonal matrices

Consider the set of equations

$$R_0\underline{U} = \underline{T}\ , \tag{5.4.1}$$

where the matrix R_0 is in block tridiagonal form, i.e.,

$$\begin{bmatrix} B_1 & C_1 & & & & \\ A_2 & B_2 & C_2 & & \bigcirc & \\ & \ddots & \ddots & \ddots & & \\ & & \ddots & \ddots & \ddots & \\ & & & \ddots & \ddots & C_{n-1} \\ & \bigcirc & & & A_n & B_n \end{bmatrix} \begin{bmatrix} \underline{U}_1 \\ \underline{U}_2 \\ \vdots \\ \vdots \\ \underline{U}_{n-1} \\ \underline{U}_n \end{bmatrix} = \begin{bmatrix} \underline{T}_1 \\ \underline{T}_2 \\ \vdots \\ \vdots \\ \underline{T}_{n-1} \\ \underline{T}_n \end{bmatrix} \tag{5.4.2}$$

A_i, B_i, C_i are each submatrices, $1 \le i \le n$.

Multiply the first block of equations by B_1^{-1} (B_1 is nonsingular)

$$\begin{bmatrix} I & G_1 & & & & & \\ A_2 & B_2 & C_2 & & & & \\ & A_3 & B_3 & C_3 & & \bigcirc & \\ & & \ddots & \ddots & \ddots & & \\ & \bigcirc & & \ddots & \ddots & \ddots & \\ & & & & A_{n-1} & B_{n-1} & C_{n-1} \\ & & & & & A_n & B_n \end{bmatrix} \begin{bmatrix} \underline{U}_1 \\ \underline{U}_2 \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \underline{U}_n \end{bmatrix} = \begin{bmatrix} \underline{H}_1 \\ \underline{T}_2 \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \underline{T}_n \end{bmatrix} \tag{5.4.3}$$

where,

$$G_1 = B_1^{-1}C_1 \quad , \quad H_1 = B_1^{-1}\underline{T}_1\ ,$$

adding, $-A_2$ times the first block of equations (5.4.3) to the second block of equations yields,

$$\begin{bmatrix} I & G_1 & & & & \\ (B_2-A_2G_1) & C_2 & & & \bigcirc & \\ A_3 & B_3 & C_3 & & & \\ & \ddots & \ddots & \ddots & & \\ \bigcirc & & A_{n-1} & B_{n-1} & C_{n-1} & \\ & & & A_n & B_n & \end{bmatrix} \begin{bmatrix} \underline{U}_1 \\ \underline{U}_2 \\ \underline{U}_3 \\ \vdots \\ \underline{U}_n \end{bmatrix} = \begin{bmatrix} \underline{H}_1 \\ \underline{T}_2-A_2\underline{H}_1 \\ \underline{T}_3 \\ \vdots \\ \underline{T}_n \end{bmatrix} \qquad (5.4.4)$$

Then multiplying the second block of equations of (5.4.3) by $(B_2-A_2G_1)^{-1}$ (assuming that $(B_2-A_2G_1)$ is non-singular) yields the system,

$$\begin{bmatrix} I & G_1 & & & & \\ & I & G_2 & & \bigcirc & \\ & A_3 & B_3 & C_3 & & \\ & & \ddots & \ddots & \ddots & \\ \bigcirc & & & A_{n-1} & B_{n-1} & C_{n-1} \\ & & & & A_n & B_n \end{bmatrix} \begin{bmatrix} \underline{U}_1 \\ \underline{U}_2 \\ \underline{U}_3 \\ \vdots \\ \underline{U}_n \end{bmatrix} = \begin{bmatrix} \underline{H}_1 \\ \underline{H}_2 \\ \underline{T}_3 \\ \vdots \\ \underline{T}_n \end{bmatrix} \qquad (5.4.5)$$

where,

$$G_2 = (B_2-A_2G_1)^{-1}C_2 \;, \qquad \underline{H}_2 = (B_2-A_2G_1)^{-1}(\underline{T}_2-A_2\underline{H}_1) .$$

Consequently,

$$G_i = (B_i-A_iG_{i-1})^{-1}C_i \;, \quad i=2,3,\ldots,n-1 \qquad (5.4.6)$$

$$\underline{H}_i = (B_i-A_iG_{i-1})^{-1}(T_i-A_i\underline{H}_{i-1}) \;, \quad i=2,3,\ldots,n \qquad (5.4.7)$$

and for $i=2,3,\ldots,n$

$$\underline{H}_{k,i-k} = \underline{H}_{k,i-1-k}+(-1)^{i-k}\left(\prod_{j=k}^{i-1} G_j\right)\underline{H}_{i,0} \;, \quad k=i-1,i-2,\ldots,1$$

$$(\underline{H}_{i,0} = \underline{H}_i) . \qquad (5.4.8)$$

Hence, after (n-1) applications of the above process, the original system is replaced by,

$$\begin{bmatrix} I & & & & & \\ & I & & & O & \\ & & I & & & \\ & & & \ddots & & \\ & O & & & \ddots & \\ & & & & & I \end{bmatrix} \begin{bmatrix} \underline{U}_1 \\ \underline{U}_2 \\ \vdots \\ \underline{U}_{n-1} \\ \underline{U}_n \end{bmatrix} = \begin{bmatrix} \underline{H}_{1,n-1} \\ \underline{H}_{2,n-2} \\ \vdots \\ \underline{H}_{n-1,1} \\ \underline{H}_{n,0} \end{bmatrix} \tag{5.4.9}$$

The solution vector can then be obtained and is given by,

$$\underline{U}_i = \underline{H}_{i,n-i} \quad , \quad i=1,\ldots,n. \tag{5.4.10}$$

Algorithm 2

Block form of Gaussian Elimination Algorithm

Consider the system (5.4.2), where the coefficient matrix is in block tridiagonal form and the vectors $\underline{U}$ and $\underline{T}$ are partitioned relative to the coefficient matrix, then putting,

$$\begin{aligned} G_1 &= B_1^{-1}C_1 \quad , \qquad G_i = (B_i-A_iG_{i-1})^{-1}C_i \ , \quad i=2,3,\ldots,n-1 \\ \underline{H}_1 &= B_1^{-1}\underline{T}_1 \quad , \qquad \underline{H}_i = (B_i-A_iG_{i-1})^{-1}(\underline{T}_i-A_i\underline{H}_{i-1}) \ , \ i=2,3,\ldots,n \end{aligned} \tag{5.4.11}$$

(assuming that B_1 and $(B_i-A_iG_{i-1})$, $i=2,3,\ldots,n$, are non-singular).

The solution can then be obtained by the back substitution process so that

$$\underline{U}_n = \underline{H}_n \tag{5.4.12}$$

and
$$\underline{U}_i = \underline{H}_i-G_i\underline{U}_{i+1} \quad , \ i=n-1,n-2,\ldots,1.$$

Algorithm 3

Partitioning Method (Block Decomposition)

Consider the set of equations

$$R_O\underline{U} = \underline{T} \ , \tag{5.4.13}$$

where R_O is a block tridiagonal matrix of the form given in (5.4.2), in this method, the matrix R_O is partitioned into four submatrices by the scheme,

$$R_0 = \begin{bmatrix} D_1 & B_2 \\ B_1 & D_2 \end{bmatrix} , \tag{5.4.14}$$

where D_1 and D_2 are square submatrices of order p and q, p+q=n, and D_1,D_2 are non-singular, B_1 and B_2 are submatrices of dimensions q×p and p×q, respectively.

The inverse of R_0 may be similarly partitioned as

$$R_0^{-1} = \begin{bmatrix} K_1 & L_2 \\ & \\ L_1 & K_2 \end{bmatrix} . \tag{5.4.15}$$

We wish to find K_1,K_2,L_1 and L_2 in terms of D_1,D_2,B_1 and B_2. From $R_0 R_0^{-1}$=I, we have,

$$\begin{aligned} D_1K_1 + B_2L_1 &= I , \\ D_1L_2 + B_2K_2 &= 0 , \\ B_1K_1 + D_2L_1 &= 0 , \\ B_1L_2 + D_2K_2 &= I . \end{aligned} \tag{5.4.16}$$

The best formulation of this solution is the one involving the minimum number of matrix inversions, and hence the solution procedure is:

(i) Compute D_1^{-1} ,

(ii) Compute $W = D_2-(B_1D_1^{-1})B_2$,

(iii) Then $K_2 = W^{-1}$

$$\begin{aligned} L_1 &= -K_2(B_1D_1^{-1}) , \\ L_2 &= -(D_1^{-1}B_2)K_2 , \\ K_1 &= D_1^{-1} - (D_1^{-1}B_2)L_1 . \end{aligned}$$

Hence, the system (5.4.13) may be solved as

$$\underline{U} = \begin{bmatrix} \underline{U}_1 \\ \underline{U}_2 \end{bmatrix} = R_0^{-1}\underline{T} = \begin{bmatrix} K_1 & L_2 \\ L_1 & K_2 \end{bmatrix} \begin{bmatrix} \underline{T}_1 \\ \underline{T}_2 \end{bmatrix} \tag{5.4.17}$$

(see Faddeev and Faddeeva [1963], pp.161).

The application of the procedure given in Algorithm 3 can be simplified if the two submatrices D_1 and D_2 of the matrix R_0 given in (5.4.14) are diagonal submatrices, since the first step in the procedure required the determination of D_1^{-1} which is straightforward in this case.

This can be achieved by ordering the groups of four points within the blocks of the 2-lines in Red-Black ordering. Hence, the resulting coefficient matrix A is of order m^2 and has the same block structure given in (5.3.4), and for the model problem, the submatrices R_1 and R_2 are the same as those given in (5.3.11b) and (5.3.11c), but the submatrix R_0 is given by,

$$
R_0 = \left[\begin{array}{c:c:c:c|c:c:c:c}
\begin{smallmatrix}24&&&0\\&24&&\\&&24&\\0&&&24\end{smallmatrix} & & & & \begin{smallmatrix}b&a&0&0\\a&b&0&0\\c&b&0&0\\b&c&0&0\end{smallmatrix} & & & \\ \hdashline
 & \begin{smallmatrix}24&&&0\\&24&&\\&&24&\\0&&&24\end{smallmatrix} & & & \begin{smallmatrix}0&0&c&b\\0&0&b&c\\0&0&b&a\\0&0&a&b\end{smallmatrix} & \begin{smallmatrix}b&a&0&0\\a&b&0&0\\c&b&0&0\\b&c&0&0\end{smallmatrix} & & \\ \hdashline
 & & \ddots & & & & \ddots & \\ \hdashline
 & & & \begin{smallmatrix}24&&&0\\&24&&\\&&24&\\0&&&24\end{smallmatrix} & & & \begin{smallmatrix}0&0&c&b\\0&0&b&c\\0&0&b&a\\0&0&a&b\end{smallmatrix} & \begin{smallmatrix}b&a&0&0\\a&b&0&0\\c&b&0&0\\b&c&0&0\end{smallmatrix} \\ \hline
\begin{smallmatrix}0&0&c&b\\0&0&b&c\\0&0&b&a\\0&0&a&b\end{smallmatrix} & \begin{smallmatrix}b&a&0&0\\a&b&0&0\\c&b&0&0\\b&c&0&0\end{smallmatrix} & & & \begin{smallmatrix}24&&&0\\&24&&\\&&24&\\0&&&24\end{smallmatrix} & & & \\ \hdashline
 & \begin{smallmatrix}0&0&c&b\\0&0&b&a\\0&0&b&a\\0&0&a&b\end{smallmatrix} & \begin{smallmatrix}b&a&0&0\\a&b&0&0\\c&b&0&0\\b&c&0&0\end{smallmatrix} & & & \ddots & & \\ \hdashline
 & & \ddots & & & & \ddots & \\ \hdashline
 & & & \begin{smallmatrix}0&0&c&b\\0&0&b&c\\0&0&b&a\\0&0&a&b\end{smallmatrix} & & & & \begin{smallmatrix}24&&&0\\&24&&\\&&24&\\0&&&24\end{smallmatrix}
\end{array}\right]
$$

where, $a=-\omega_1$, $b=-2\omega_1$, and $c=-7\omega_1$.

5.5 EXPERIMENTAL RESULTS

Numerical results are presented here for the two methods described earlier in this chapter for solving Laplace's equation (4.3.1) in the unit square subject to the boundary condition (4.3.2). In order to compare the new methods with the well known point, line and 2-line S.O.R. methods which were described in Chapter 4, we used the same convergence test, i.e. the average test (4.3.5) with $\varepsilon=10^{-7}$.

Concerning the experiments carried out and the results obtained for solving the model problem by the two IBEB methods, we will give more attention to the second method, i.e., the 4-point, 2-line IBEB iterative method, since it is more convergent than the 2-point, 1-line, IBEB method. However considering the 2-point, 1-line IBEB iterative method, we know from the discussion given in Section (5.2) that ω_1 is related to the explicit 2-point group overrelaxation scheme given by equation (5.2.6) and since ω_2 is the parameter related to the 1-line block iterative scheme, then one expects that when $\omega_1=1$, the corresponding optimum value of ω_2 is the same as ω_b for the 1-line S.O.R. iterative method (see Table (4.3.2)), this is ratified by the results shown in Table (5.5.1).

For the 4-point, 2-line IBEB iterative method and from the analysis given in Section (5.3), the parameter ω_1 is related to the explicit 4-point group overrelaxation scheme given by equation (5.3.10) which has a spectral radius given by (4.3.4), since from the theory given in sub-section (4.2.2) we know that the coefficient matrix resulting from this grouping has Property $A^{(\pi)}$ and is π-consistently ordered, then the theory of block S.O.R. is valid and can be used to predict ω_1. However, from the results obtained to solve the model problem (4.3.1) by this method which are given in Tables (5.5.2)-(5.5.5) we can make the following hypothesis for the determination of the optimum value of the parameter ω_1,$\tilde{\omega}_1$ (say).

Hypothesis (5.5.1)

The optimum value of ω_1 ($\tilde{\omega}_1$) is given by the following formulae,

$$\tilde{\omega}_1 = \frac{2}{1+k_1\sqrt{1-[\rho(J_{4pt})]^2}} , \tag{5.5.1}$$

where $\rho(J_{4pt})$ is the spectral radius of the 4-point group Jacobi scheme and k_1 is a scaling parameter that has been inserted and equals approximately $\sqrt{2}$.

The parameter ω_2 is related to the 2-line block iterative scheme and from the experimental results recorded in Tables (5.5.2)-(5.5.5), the optimum value of ω_2, $\tilde{\omega}_2$ (say), can be determined and from the results the following hypothesis can be made.

Hypothesis (5.5.2)

The relation which combines the two optimum parameters $\tilde{\omega}_1$ and $\tilde{\omega}_2$ is given by,

$$\tilde{\omega}_2 = 2-\tilde{\omega}_1 . \tag{5.5.2}$$

Further extensive investigations will be necessary to confirm whether the above two hypotheses are valid over a wider range of problems.

Hence, we can state that this composite method is an S.O.R. method with a convergence rate superior to the well known point, line and 2-line S.O.R. iterative methods as shown in Figure (5.5.1), where log(k) was plotted against $\log(h^{-1})$ with k the minimum number of iterations and h the mesh size for each method. The data for the point, line and 2-line

S.O.R. iterative methods were taken from Tables (4.3.1),(4.3.2) and (4.3.3) respectively.

The graphs of $\log(h^{-1})$ versus $\tilde{\omega}_1$ and $\tilde{\omega}_2$ are shown in Figure (5.5.2). All the results were obtained using the "ICL 1904S computer" at Loughborough University and the "ICL 1904S/CDC 7600" joint computer system at Manchester University using the "MIDNET" link.

h^{-1}	Range of values of ω_1 and ω_2		Number of iterations (k)
	ω_1	ω_2	
13	1.0	1.0	126
	1.0	1.509-1.514	25
	1.02	1.466-1.472	"
	1.04	1.423-1.43	"
	1.06	1.382-1.39	"
	⋮	⋮	
25	1.0	1.0	426
	1.0	1.704-1.71	47
	1.02	1.653-1.661	"
	1.04	1.605-1.613	"
	1.06	1.559-1.566	"
	⋮	⋮	
37	1.0	1.789-1.791	68
	1.02	1.736-1.739	"
	1.04	1.686-1.689	"
	1.06	1.639	67
	⋮	⋮	
49	1.0	1.836-1.837	88
	1.02	1.782	"
	1.04	1.731	"
	1.06	1.681	"
	⋮		

TABLE 5.5.1: The 2-point, 1-line IBEB iterative method

Range of Values of ω_1 and ω_2		k
ω_1	ω_2	
1.0	1.0	68
1.0	1.384-1.386	18
1.04	1.283-1.286	18
1.06	1.236-1.239	18
1.1	1.146-1.149	18
1.14	1.064	17
⋮	⋮	
1.35	0.712-0.713	17
1.37	0.685-0.687	17
1.38	0.672	16
1.39	0.66	16
1.4	0.647-0.648	16
1.41	0.635	15
1.413	0.632	15
1.415	0.629	15
1.416	0.628	15
1.417	0.627	"
1.418	0.626	"
1.419	0.625	"
1.42	0.624	"
1.422	0.622	"
1.425	0.618-0.619	16
1.43	0.61-0.615	17
1.44	0.6-0.602	17
1.45	0.585-0.598	18
1.46	0.575-0.585	18
⋮	⋮	

TABLE 5.5.2: The 4-point, 2-line IBEB method ($h^{-1}=13$)

Range of values of ω_1 and ω_2		k
ω_1	ω_2	
1.0	1.0	226
0.96	1.727-1.732	40
1.0	1.608-1.611	34
1.02	1.547-1.549	33
1.04	1.488	32
1.06	1.431	32
1.1	1.322	31
1.14	1.221	31
1.18	1.126-1.127	32
1.26	0.956	31
1.34	0.806	31
1.42	0.673	31
1.5	0.555	30
1.61	0.414	28
1.615	0.408	27
1.616	0.407	"
1.617	0.406	"
1.618	0.405	"
1.619	0.404	"
1.62	0.403	"
1.621	0.402	"
1.622	0.4	"
1.623	0.399	27
1.624	0.398	28
1.625	0.397	29
1.63	0.391-0.392	30
1.66	0.357-0.36	32
1.7	0.316	34
1.72	0.293-0.298	38
⋮	⋮	

TABLE 5.5.3: The 4-point, 2-line IBEB method (h^{-1}=25)

Range of Values of ω_1 and ω_2		k
ω_1	ω_2	
1.0	1.714-1.716	50
1.02	1.649	47
1.04	1.585	46
1.06	1.523	46
1.1	1.406-1.407	46
1.14	1.298	45
1.18	1.196-1.199	46
1.26	1.013	45
⋮	⋮	
1.56	0.497	45
1.62	0.413	43
1.68	0.339	43
1.7	0.316	42
1.715	0.299	40
1.716	0.297-0.298	40
1.717	0.296	38
1.718	0.295	38
1.719	0.294	39
1.72	0.293	39
1.721	0.292	39
1.722	0.291	40
1.73	0.282	43
1.74	0.271-0.272	45
1.76	0.25	46
⋮	⋮	

TABLE 5.5.4: The 4-point, 2-line IBEB method (h^{-1}=37)

Range of Values of ω_1 and ω_2		k
ω_1	ω_2	
1.0	1.774-1.777	66
1.02	1.708	61
1.04	1.641	61
1.06	1.577	60
1.1	1.456	60
1.14	1.343	59
1.18	1.237-1.238	60
⋮	⋮	
1.75	0.262	56
1.76	0.25	52
1.768	0.241-0.242	53
1.769	0.24	50
1.77	0.239	50
1.771	0.238	51
1.772	0.237	52
1.775	0.234	52
1.777	0.231	55
1.779	0.229	51
1.78	0.228	49
1.781	0.227	51
1.782	0.226	53
1.785	0.223	56
1.79	0.217	56
1.8	0.207	58
⋮	⋮	

TABLE 5.5.5: The 4-point, 2-line, IBEB method ($h^{-1}=49$)

h^{-1}	$\tilde{\omega}_1$	$\tilde{\omega}_2$	k	ρ
13	1.422	0.622	15	0.169536
25	1.621	0.402	27	0.361351
37	1.718	0.295	38	0.587773
49	1.78	0.228	49	0.661283

TABLE 5.5.6: Optimum values of the spectral radius of the 4-point, 2-line IBEB method

h^{-1}	Optimum Parameter ($\tilde{\omega}_1$)		Optimum Parameter ($\tilde{\omega}_2$)		k
	Predicted	Experiment	Predicted	Experiment	
13	1.355	1.41-1.422	0.645	0.629-0.622	15
25	1.6	1.615-1.623	0.4	0.408-0.399	27
37	1.71	1.717&1.718	0.29	0.296&0.295	38
49	1.773	1.78	0.227	0.228	49

TABLE 5.5.7: The range of values of the two optimum parameters $\tilde{\omega}_1$ and $\tilde{\omega}_2$ for the 4-point, 2-line IBEB method using equations (5.5.1) and (5.5.2)

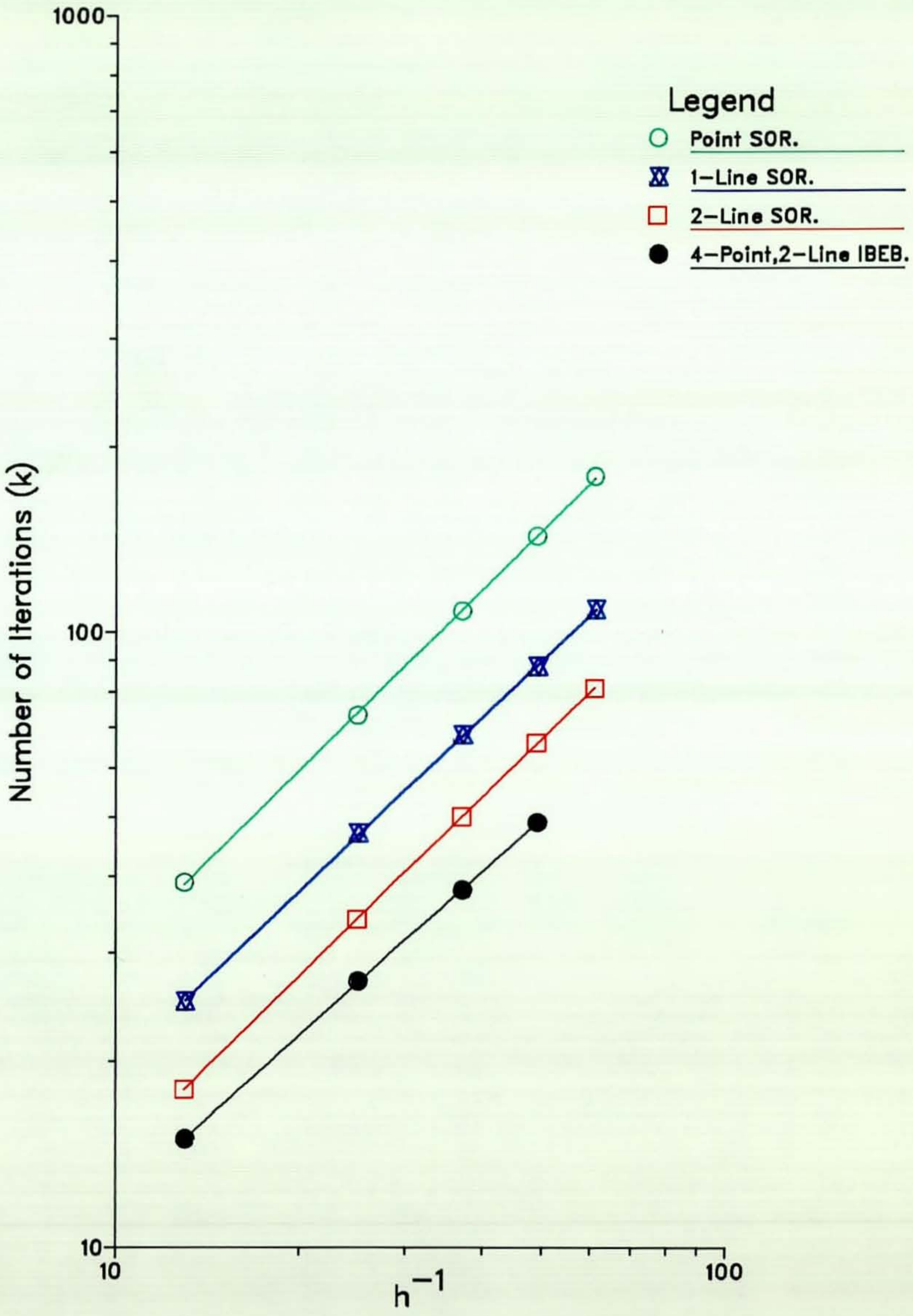
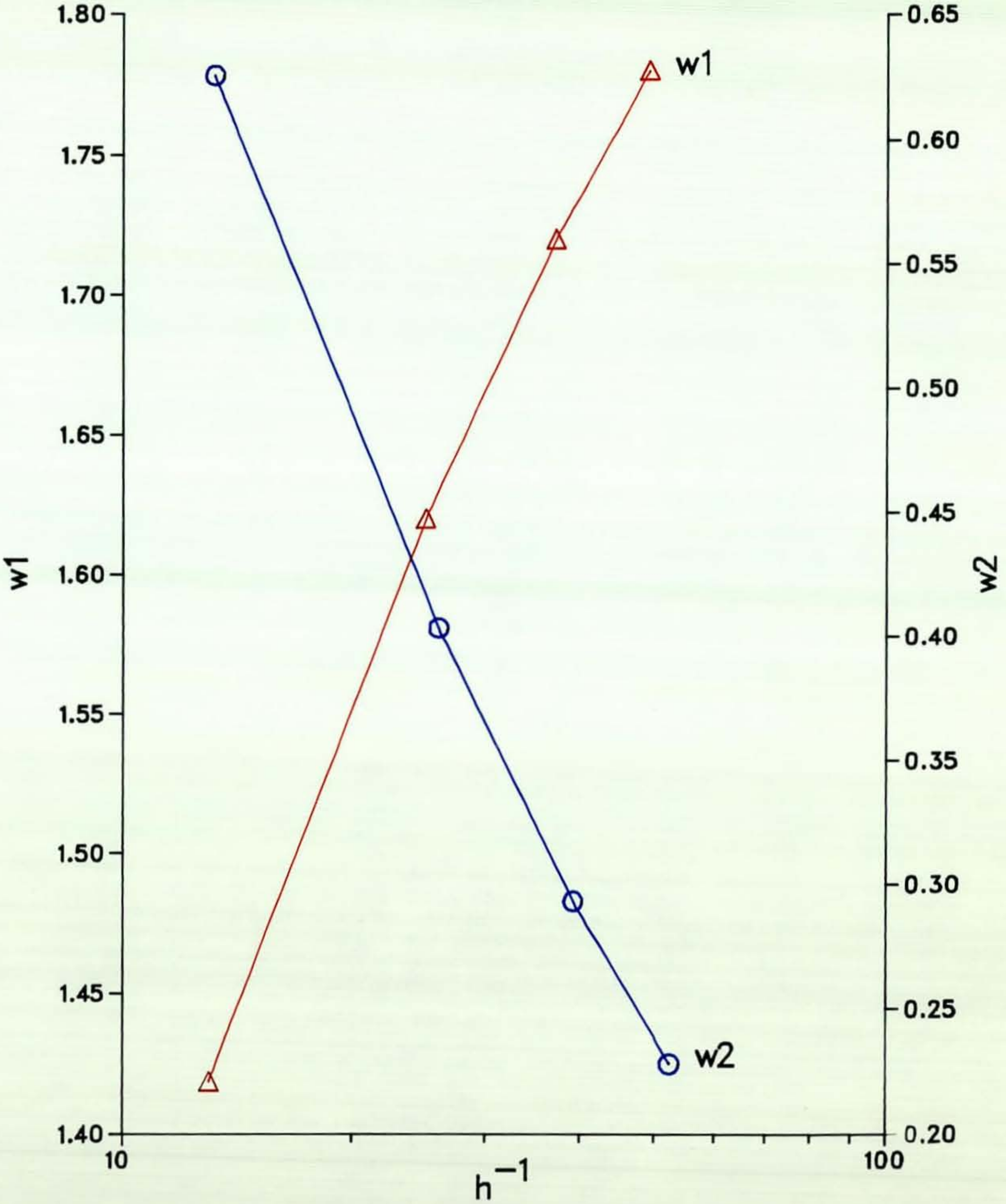

1000
100
10
Number of Iterations (k)
10
100
h^{-1}
Legend
Point SOR.
1-Line SOR.
2-Line SOR.
4-Point,2-Line IBEB.

Figure (5.5.1)



w1
w2
1.80
1.75
1.70
1.65
1.60
1.55
1.50
1.45
1.40
0.65
0.60
0.55
0.50
0.45
0.40
0.35
0.30
0.25
0.20
10
100
h^{-1}

Figure (5.5.2)

CHAPTER 6

THE SOLUTION OF TWO-POINT BOUNDARY VALUE PROBLEMS BY MULTI POINT GROUP ITERATIVE METHODS

6.1 INTRODUCTION

In this chapter we shall investigate the solution of one dimensional boundary value problems using the explicit group iterative methods, in particular the 2,3,4,6,8 and 12 point groups are analysed in order to determine whether an optimum group size exists. Further, some non-linear boundary value problems are examined using a similar group strategy to compare the 2,3 and 4 point explicit groups iterative methods with the direct approach using Picard's method to solve the non-linear system.

Consider then the differential equation,

$$-\frac{d^2U}{dx^2} + q(x)U = f(x) \ , \qquad (6.1.1)$$

over the line segment $a \leq x \leq b$, subject to the boundary conditions at the points a and b, i.e.,

$$U(a) = \alpha, \quad U(b) = \beta. \qquad (6.1.2)$$

Here, α and β are given real constants, and $q(x)$ and $f(x)$ are given real continuous functions on $a \leq x \leq b$, with,

$$q(x) \geq 0.$$

With minor modifications, (6.1.1) may describe:

(i) the steady-state displacement of a taut string subject to a distributed load, (ii) the steady-state distribution of temperature in an internally heated homogeneous wall, with surfaces kept at given temperature, (iii) the steady-state concentration of a pollutant in a porous soil, and (iv) the distribution of electrical potential between two flat electrodes.

Although not technically a partial differential equation, two-point boundary value problems may from most view points be assimilated to elliptic equations, with which they share many mathematical properties and computational methods. In particular, they share the property that their solution in an interior point depends on all boundary conditions, precluding the possibility of directly constructing the solution by marching away from one boundary, [Vichnevetsky, 1981].

For the differential equation (6.1.1), the strategy of the finite difference method, which was discussed in Section (2.8.2) is to replace the above equation by a difference equation. Hence, we cover the interval $a \leq x \leq b$, by a uniform mesh of size,

$$h = (b-a)/(m+1) \quad , \tag{6.1.3}$$

and denote the mesh points of the discrete problem by,

$$x_i = a+ih \;, \quad 0 \leq i \leq m+1 \;, \tag{6.1.4}$$

as illustrated in Figure (6.1.1).

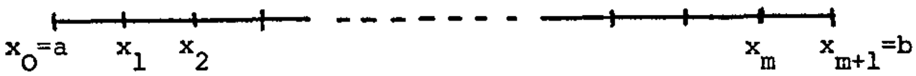


FIGURE (6.1.1)

The finite difference approximations to (6.1.1) can be derived by using the method based on finite Taylor's series expansions of the solution $U(x)$ to (6.1.1). Denoting $U(x_i)$ by U_i, the finite Taylor expansion of $U_{i\pm1}$ is,

$$U_{i\pm1} = U_i \pm h\left(\frac{dU_i}{dx}\right) + \frac{h^2}{2!}\left(\frac{d^2U_i}{dx^2}\right) \pm \frac{h^3}{3!}\left(\frac{d^3U_i}{dx^3}\right) + \frac{h^4}{4!}\frac{d^4U_i}{dx^4} \pm \cdots \tag{6.1.5}$$

from which it follows that,

$$-\frac{d^2U_i}{dx^2} = \frac{2U_i-U_{i-1}-U_{i+1}}{h^2} + \frac{h^2}{12}\frac{d^4U_i}{dx^4} + \cdots \quad . \tag{6.1.6}$$

Hence, the finite difference replacement of equation (6.1.1) can be obtained and is given by,

$$\frac{1}{h^2}(2u_i-u_{i+1}-u_{i-1}) + q_iu_i = f_i\ , \quad 1\leq i\leq m, \tag{6.1.7}$$

with a local truncation error of

$$\frac{h^2}{12}\ \frac{d^4U_i}{dx^4} + \ldots$$

where u_i denotes the function satisfying the difference equation at the mesh point x_i=a+ih. The principal part of the truncation error is meaningful only if $U \in C^4$ in [a,b], i.e. the derivatives of U w.r.t. x are continuous up to order four in this interval. Under these conditions we say that the local truncation error is $O(h^2)$.

In matrix notation, (6.1.7) can be written in the form,

$$A\underline{u} = \underline{b}\ , \tag{6.1.8}$$

where A is a real (m×m) matrix, $\underline{u}$ is the discrete approximation vector to the solution U(x) of (6.1.1)-(6.1.2) and $\underline{b}$ is a column vector given by,

$$A = \frac{1}{h^2}\begin{bmatrix} 1+q_1h^2/2 & -\frac{1}{2} & & & \\ -\frac{1}{2} & 1+q_2h^2/2 & -\frac{1}{2} & & O \\ & \ddots & \ddots & \ddots & \\ O & & -\frac{1}{2} & 1+q_{m-1}h^2/2 & -\frac{1}{2} \\ & & & -\frac{1}{2} & 1+q_mh^2/2 \end{bmatrix}, \tag{6.1.9a}$$

$$\underline{u} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{m-1} \\ u_m \end{bmatrix} \quad \text{and} \quad \underline{b} = \begin{bmatrix} f_1/2\ +\alpha/2h^2 \\ f_2/2 \\ \vdots \\ f_{m-1}/2 \\ f_m/2\ +\beta/2h^2 \end{bmatrix}. \tag{6.1.9b}$$

The tridiagonal matrix A is real and symmetric, and since $q(x)\geq 0$ then it is also diagonally dominant with positive diagonal entries (see Definition (2.2.1). From Definitions (2.2.3) and (2.2.4), the direct graph of A is strongly connected, implying that A is irreducible, and as there is strict inequality in the first and last rows of A, then A is irreducibly diagonally dominant (see Definition (2.2.5)).

In the next section we investigate the class of methods for solving the system (6.1.8) which is based on *splitting* the matrix A into two $(m\times m)$ matrices B_1 and B_2, such that,

$$A = B_1 - B_2 \ , \tag{6.1.10}$$

where we can easily convert the system (6.1.8) into n-point group iterative scheme(n=2,3,4,6,8 or 12), where

$$B_1 = \left[\begin{array}{c:c:c:c} \begin{matrix} 1+q_1h^2/2 & -\frac{1}{2} & O \\ -\frac{1}{2} & 1+q_2h_2/2 & -\frac{1}{2} \ \ -\frac{1}{2} \\ O & -\frac{1}{2} & 1+q_nh^2/2 \end{matrix} & & & O \\ \hdashline & \begin{matrix} 1+q_{n+1}h^2/2 & -\frac{1}{2} & O \\ -\frac{1}{2} & & -\frac{1}{2} \\ O & -\frac{1}{2} & 1+q_{2n}h^2/2 \end{matrix} & & \\ \hdashline & & \ddots & \\ \hdashline O & & & \begin{matrix} 1+q_{m-n+1}h^2/2 & -\frac{1}{2} & O \\ -\frac{1}{2} & & -\frac{1}{2} \\ O & -\frac{1}{2} & 1+q_mh^2/2 \end{matrix} \end{array}\right]$$

where n=2,3,4,6,8 or 12, and, (6.1.11a)

$$B_2 = \left[\begin{array}{c:c:c:c} O & \frac{1}{2} & & O \\ \hdashline \frac{1}{2} & O & & \\ \hdashline & \frac{1}{2} & \ddots & \\ & \frac{1}{2} & & \frac{1}{2} \\ \hdashline O & & \frac{1}{2} & O \end{array}\right] \tag{6.1.11b}$$

and hence, (6.1.8) can be written in the form,

$$B_1\underline{u}^{(k+1)} = B_2\underline{u}^{(k)} + \underline{b} , \tag{6.1.12}$$

but since B_1 can be explicitly inverted then (6.1.12) can be written in explicit form as,

$$\underline{u}^{(k+1)} = B_1^{-1}B_2\underline{u}^{(k)} + B_1^{-1}\underline{b} . \tag{6.1.13}$$

where $B_1^{-1}B_2$ is explicitly obtained.

6.2 ONE DIMENSIONAL n-POINT GROUP ITERATIVE METHODS

Consider equation (6.1.1) with $q(x)=0$ and $f(x)=0$ in the range $(0,1)$, i.e. the two-point boundary value problem,

$$-\frac{d^2U}{dx^2} = f(x) \;, \tag{6.2.1}$$

such that $f(x)=0$ in the range $(0,1)$, with the following boundary conditions,

$$U(0) = 1, \quad U(1) = 1 \;. \tag{6.2.2}$$

By using the method based on finite Taylor's series expansion which was described in Section (6.1), we obtain the finite difference replacement of equation (6.2.1) and is given by,

$$2u_i - u_{i+1} - u_{i-1} = 0 \;, \quad 1 \leq i \leq m \;. \tag{6.2.3}$$

Hence, the related algebraic problem is concerned with solving the following linear system,

$$\begin{bmatrix} 1 & -\frac{1}{2} & & & & \\ -\frac{1}{2} & 1 & -\frac{1}{2} & & \bigcirc & \\ & \ddots & \ddots & \ddots & & \\ & & & -\frac{1}{2} & 1 & -\frac{1}{2} \\ & \bigcirc & & & -\frac{1}{2} & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{m-1} \\ u_m \end{bmatrix} = \begin{bmatrix} \frac{1}{2} \\ 0 \\ \vdots \\ 0 \\ \frac{1}{2} \end{bmatrix} \;. \tag{6.2.4}$$

i.e.,

$$A\underline{u} = \underline{b} \;,$$

which can be easily converted into n-point group iterative scheme, where $n=2,3,4,6,8$ or 12, as mentioned before. These methods are described in the following subsections.

6.2.1 THE 2-POINT GROUP ITERATIVE SCHEME

Consider the system (6.2.4) defined in the interval $0 \leq x \leq 1$, with m internal mesh points with m even. Suppose that the mesh points are ordered in groups of two as shown in Figure (6.2.1).

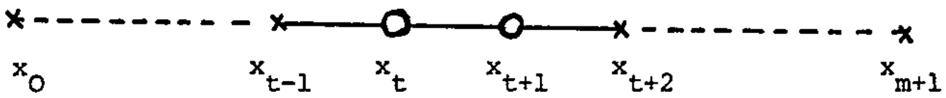


FIGURE (6.2.1)

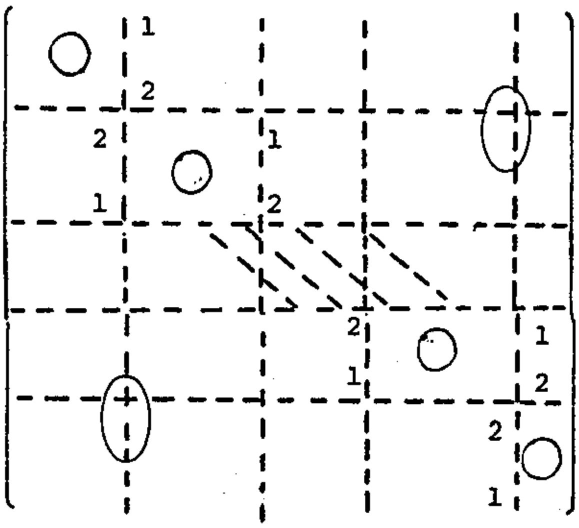
where t=1,(2),m-1. Hence, the system (6.2.4) is converted into the 2-point group iterative scheme, i.e.,

$$\begin{bmatrix} 1 & -\frac{1}{2} & & & & & \\ -\frac{1}{2} & 1 & & & & & \\ & & 1 & -\frac{1}{2} & & & \\ & & -\frac{1}{2} & 1 & & & \\ & & & & \ddots & & \\ & & & & & 1 & -\frac{1}{2} \\ & & & & & -\frac{1}{2} & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ \vdots \\ \vdots \\ u_{m-1} \\ u_m \end{bmatrix}^{(k+1)} = \begin{bmatrix} 0 & 0 & & & & & \\ 0 & 0 & \frac{1}{2} & & & & \\ & \frac{1}{2} & 0 & 0 & & & \\ & & 0 & 0 & \frac{1}{2} & & \\ & & & \ddots & \ddots & \frac{1}{2} & \\ & & & & \frac{1}{2} & 0 & 0 \\ & & & & & 0 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ \vdots \\ \vdots \\ u_{m-1} \\ u_m \end{bmatrix}^{(k)} + \begin{bmatrix} \frac{1}{2} \\ 0 \\ 0 \\ \vdots \\ \vdots \\ 0 \\ \frac{1}{2} \end{bmatrix} , \qquad (6.2.5)$$

i.e, $$B_1\underline{u}^{(k+1)} = B_2\underline{u}^{(k)} + \underline{b} ,$$

which can be written in the explicit form (6.1.13), where B_1^{-1} can be explicitly obtained, also $B_1^{-1}B_2$ and $B_1^{-1}\underline{b}$ can be determined explicitly to give,

$$\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{m-1} \\ u_m \end{bmatrix}^{(k+1)} = \frac{1}{3}$$



$$\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{m-1} \\ u_m \end{bmatrix}^{(k)} + \frac{1}{3}\begin{bmatrix} 2 \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 2 \end{bmatrix}$$

(6.2.6)

i.e. $$\underline{u}^{(k+1)} = B_1^{-1}B_2\underline{u}^{(k)} + B_1^{-1}\underline{b} .$$

Hence, the 2-point group Gauss-Seidel (G.S.) iterative method is given by,

$$u_t^{(k+1)} = (2u_{t-1}^{(k+1)} + u_{t+2}^{(k)})/3 ,$$

and $$u_{t+1}^{(k+1)} = (u_{t-1}^{(k+1)} + 2u_{t+2}^{(k)})/3 , \qquad (6.2.7)$$

where $t=1,(2),m-1$ and the points x_t, where $t=0$ and $t=m+1$ lies on the boundary. Related to the G.S. method is the 2-point group explicit S.O.R. iterative method which replaces (6.2.7) by,

$$u_t^{(k+1)} = \omega(2u_{t-1}^{(k+1)} + u_{t+2}^{(k)})/3 + (1-\omega)u_t^{(k)},$$

$$u_{t+1}^{(k+1)} = \omega(u_{t-1}^{(k+1)} + 2u_{t+2}^{(k)})/3 + (1-\omega)u_{t+1}^{(k)} , \qquad (6.2.8)$$

where ω is the relaxation factor. Clearly similar sets of equations can be written down for the 2-point explicit Jacobi and J.O.R. iterative methods.

6.2.2 THE 3-POINT GROUP ITERATIVE SCHEME

Similar to the 2-point group an explicit set of equations can be derived for this scheme in which each group is formed from 3 points of the mesh interval Figure (6.1.1) in accordance with Figure (6.2.2).

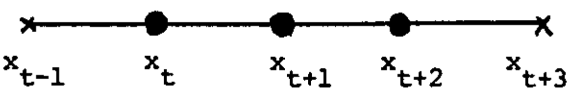


FIGURE (6.2.2)

where t=1,(3),m-2 and m is divisible by 3. The system (6.2.4) can now be converted into a 3-point group iterative scheme similar to (6.2.5) with B_1 and B_2 given by,

$$B_1=\left[\begin{array}{c|c|c|c} \begin{matrix}1&-\frac12&0\\-\frac12&1&-\frac12\\0&-\frac12&1\end{matrix} & & & \bigcirc \\ \hline & \begin{matrix}1&-\frac12&0\\-\frac12&1&-\frac12\\0&-\frac12&1\end{matrix} & & \\ \hline & & \ddots & \\ \hline \bigcirc & & & \begin{matrix}1&-\frac12&0\\-\frac12&1&-\frac12\\0&-\frac12&1\end{matrix} \end{array}\right], \quad B_2=\left[\begin{array}{c|c|c|c} \bigcirc & {}^{\frac12} & & \bigcirc \\ \hline {}_{\frac12} & \bigcirc & {}^{\frac12} & \\ \hline & {}_{\frac12} & \ddots & {}^{\frac12} \\ \hline \bigcirc & & {}_{\frac12} & \bigcirc \end{array}\right]$$

(6.2.9)

B_1^{-1} can be obtained by inverting the sub-diagonal matrices of B_1, D_i (say), $i=1,2,\ldots,m/3$, but before working out the inverse, we state the following method to find the inverse of a special tridiagonal matrix [see Gregory and Karney (1969), p.48].

[If M is an n×n matrix of the form,

$$M=\begin{bmatrix} (x+b) & 1 & & & & \\ 1 & x & 1 & & \bigcirc & \\ & 1 & x & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & \bigcirc & & 1 & x & 1 \\ & & & & 1 & (x+a) \end{bmatrix}, \qquad (6.2.10)$$

then, $M^{-1}=[c_{ij}]$ is the (n×n) matrix defined by,

$$c_{ij} = c_{ji} = \frac{(-1)^{i+j} r_{j-1} s_{n-i}}{(x+a)r_{n-1} - r_{n-2}} \quad , \text{ if } j \leq i \; , \tag{6.2.11}$$

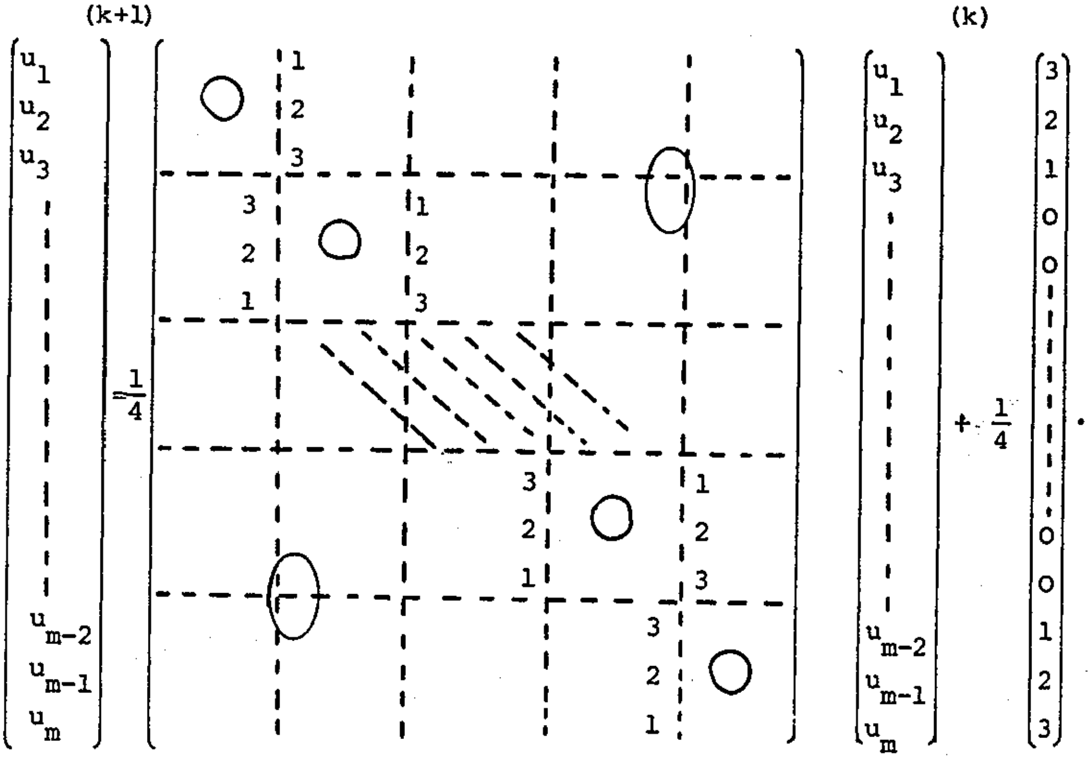
where,
$$\left.\begin{aligned} & r_0 = 1, \quad r_1 = x+b \; , \\ & r_\ell = x r_{\ell-1} - r_{\ell-2} \; , \quad \ell=2,3,\ldots,n-1, \end{aligned}\right.$$

and
$$\left.\begin{aligned} & s_0 = 1, \quad s_1 = x+a \; , \\ & s_\ell = x s_{\ell-1} - s_{\ell-2} \quad , \ell=2,3,\ldots,n-1 \;] \; . \end{aligned}\right. \tag{6.2.12}$$

Hence, if we multiply the (3×3) submatrix D by -2, we get a matrix of the form given in (6.2.10) with x=-2 and a=b=0, which can be inverted by using (6.2.11) and (6.2.12) and by multiplying the resulting inverse matrix by -2 again we have D_i^{-1}, i=1,2,...,m/3 and hence B_1^{-1} is given by,

$$B_1^{-1} = \frac{1}{2} \left[\begin{array}{ccc|ccc|c|ccc} 3 & 2 & 1 & & & & & & & \\ 2 & 4 & 2 & & & & & & & \\ 1 & 2 & 3 & & & & & & & \\ \hline & & & 3 & 2 & 1 & & & & \\ & & & 2 & 4 & 2 & & & & \\ & & & 1 & 2 & 3 & & & & \\ \hline & & & & & & \ddots & & & \\ \hline & & & & & & & 3 & 2 & 1 \\ & & & & & & & 2 & 4 & 2 \\ & & & & & & & 1 & 2 & 3 \end{array}\right] \quad , \tag{6.2.13}$$

from which $B_1^{-1}B_2$ and $B_1^{-1}\underline{b}$ can be obtained explicitly and we have the 3-point iterative scheme,



(6.2.14)

Hence the 3-point group Jacobi iterative method is given by,

$$u_t^{(k+1)} = (3u_{t-1}^{(k)}+u_{t+3}^{(k)})/4 \ ,$$

$$u_{t+1}^{(k+1)} = (2u_{t-1}^{(k)}+2u_{t+3}^{(k)})/4 \ , \qquad (6.2.15)$$

and

$$u_{t+2}^{(k+1)} = (u_{t-1}^{(k)}+3u_{t+3}^{(k)})/4 \ ,$$

where $t=1,(3),m-2$.

Another set of equations can be written down for the 3-point group J.O.R. iterative method which is related to this method, and similar sets of equations can be obtained for the 3-point group G.S. and S.O.R. iterative methods.

6.2.3 THE 4-POINT GROUP ITERATIVE SCHEME

In this scheme, the mesh points of the interval of Figure (6.1.1) are ordered in groups of four as illustrated in Figure (6.2.3).

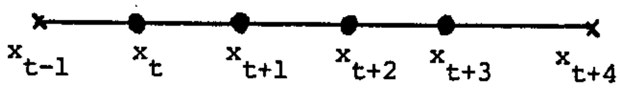


FIGURE (6.2.3)

where t=1,(4),m-3 and m is divisible by 4. Similar to the 2 and 3-point groups, the system (6.2.4) can be converted into a 4-point group iterative scheme with the (m×m) block diagonal matrix B_1 having the form (6.1.11a) such that each subdiagonal matrix D_i, $i=1,2,\ldots,m/4$, is a 4×4 matrix of the form,

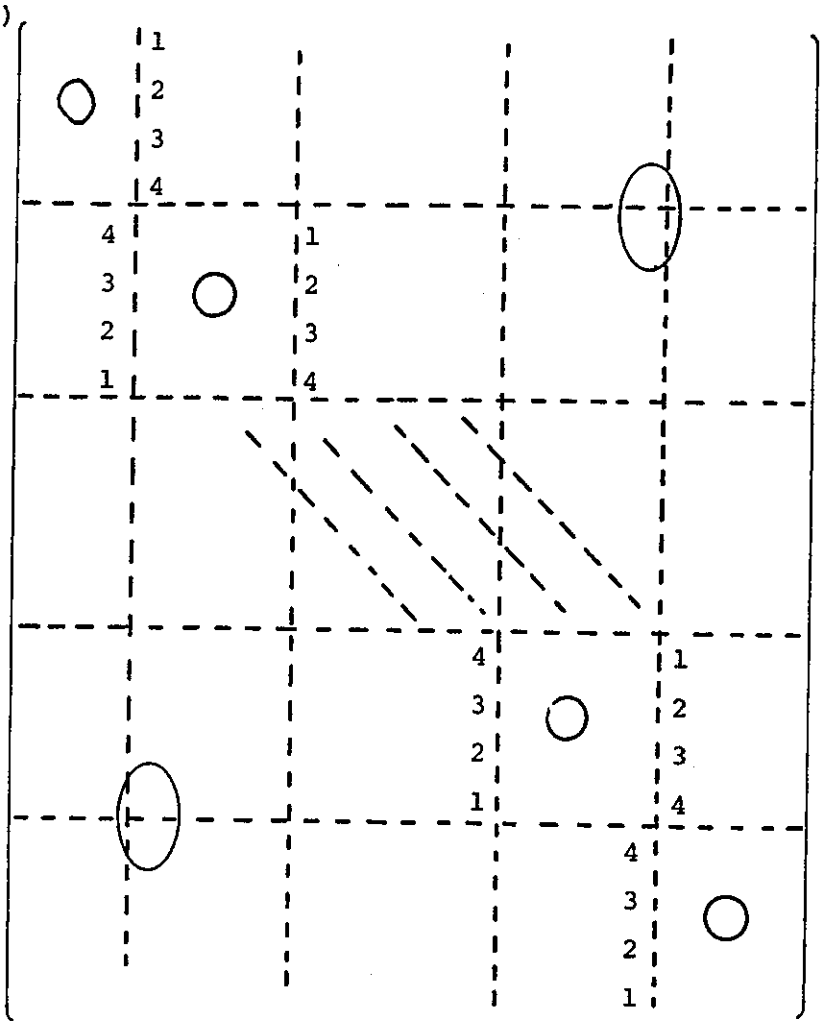
$$D_i = \begin{bmatrix} 1 & -\frac{1}{2} & & 0 \\ -\frac{1}{2} & 1 & -\frac{1}{2} & \\ & -\frac{1}{2} & 1 & -\frac{1}{2} \\ 0 & & -\frac{1}{2} & 1 \end{bmatrix} \qquad , \; i=1,2,\ldots,m/4.$$

By using the method given in sub-section (6.2.2), B_1^{-1} can be determined and is given by,

$$B_1^{-1} = \frac{2}{5}\left[\begin{array}{cccc:c:cccc} 4 & 3 & 2 & 1 & & & & & \\ 3 & 6 & 4 & 2 & & & & 0 & \\ 2 & 4 & 6 & 3 & & & & & \\ 1 & 2 & 3 & 4 & & & & & \\ \hdashline & & & & \ddots & & & & \\ \hdashline & & & & & 4 & 3 & 2 & 1 \\ & 0 & & & & 3 & 6 & 4 & 2 \\ & & & & & 2 & 4 & 6 & 3 \\ & & & & & 1 & 2 & 3 & 4 \end{array}\right] , \qquad (6.2.16)$$

and B_2 has the form (6.1.11b). Now, $B_1^{-1}B_2$ and $B_1^{-1}\underline{b}$ can be determined explicitly and we have the 4-point iterative scheme, i.e.,

$$\begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_{m-1} \\ u_m \end{pmatrix}^{(k+1)} = \frac{1}{5}$$



$$\begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_{m-1} \\ u_m \end{pmatrix}^{(k)}$$

$$+\ \frac{1}{5}\begin{pmatrix} 4 \\ 3 \\ 2 \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{pmatrix} \qquad (6.2.17)$$

Clearly, by using this iterative system, the 4-point group Jacobi iterative method can be written in explicit form, and related to the Jacobi method is the 4-point group J.O.R. method. Also, by using the $(k+1)^{th}$ iterative values as soon as they are available we can obtain the 4-point group G.S. method and related to it is the group S.O.R. method. Hence, the explicit 4-point group G.S. iterative method is given by,

$$
\begin{aligned}
u_t^{(k+1)} &= \frac{1}{5}(4u_{t-1}^{(k+1)} + u_{t+4}^{(k)}) , \\
u_{t+1}^{(k+1)} &= \frac{1}{5}(3u_{t-1}^{(k+1)} +2u_{t+4}^{(k)}) , \\
u_{t+2}^{(k+1)} &= \frac{1}{5}(2u_{t-1}^{(k+1)} +3u_{t+4}^{(k)}) , \qquad (6.2.18)
\end{aligned}
$$

and

$$u_{t+3}^{(k+1)} = \frac{1}{5}(u_{t-1}^{(k+1)} +4u_{t+4}^{(k)}) .$$

where $t=1,(4),m-3$.

6.2.4 THE 6-POINT GROUP ITERATIVE SCHEME

The points of the interval of Figure (6.1.1) are ordered here in groups of six, where each subset (group) G_ℓ of Definition (3.11.1), $\ell=1,2,\ldots,m/6$, consists of six elements $\{x_t,x_{t+1},x_{t+2},x_{t+3},x_{t+4},x_{t+5}\}$, and m must be divisible by 6. By converting the system (6.2.4) into a 6-point group iterative scheme, the subdiagonal matrices D_i, $i=1,2,\ldots,m/6$, of the $(m\times m)$ block diagonal matrix B_1 are 6×6 tridiagonal matrices given by,

$$
D_i = \begin{bmatrix}
1 & -\frac{1}{2} & & & & \\
-\frac{1}{2} & 1 & -\frac{1}{2} & & \bigcirc & \\
 & -\frac{1}{2} & 1 & -\frac{1}{2} & & \\
 & & -\frac{1}{2} & 1 & -\frac{1}{2} & \\
 & \bigcirc & & -\frac{1}{2} & 1 & -\frac{1}{2} \\
 & & & & -\frac{1}{2} & 1
\end{bmatrix} , \quad i=1,2,\ldots,m/6,
$$

hence, by multiplying B_1 by -2 and then use the method given in sub-section (6.2.2), B_1^{-1} can be obtained and is given by,

$$B_1^{-1} = \frac{2}{7}\left[\begin{array}{c:c:c}
\begin{matrix} 6 & 5 & 4 & 3 & 2 & 1 \\ 5 & 10 & 8 & 6 & 4 & 2 \\ 4 & 8 & 12 & 9 & 6 & 3 \\ 3 & 6 & 9 & 12 & 8 & 4 \\ 2 & 4 & 6 & 8 & 10 & 5 \\ 1 & 2 & 3 & 4 & 5 & 6 \end{matrix} & & \bigcirc \\ \hdashline
 & \ddots & \\ \hdashline
\bigcirc & & \begin{matrix} 6 & 5 & 4 & 3 & 2 & 1 \\ 5 & 10 & 8 & 6 & 4 & 2 \\ 4 & 8 & 12 & 9 & 6 & 3 \\ 3 & 6 & 9 & 12 & 8 & 4 \\ 2 & 4 & 6 & 8 & 10 & 5 \\ 1 & 2 & 3 & 4 & 5 & 6 \end{matrix}
\end{array}\right] \qquad (6.2.19)$$

B_2 has the form (6.1.11b) and $\underline{b}$ is given by,

$$\underline{b} = (\tfrac{1}{2},0,0,\ldots,0,0,\tfrac{1}{2})^T \ .$$

Hence, $B_1^{-1}B_2$ and $B_1^{-1}\underline{b}$ can be obtained explicitly, and the 6-point group G.S. iterative method can be written as,

$$\left.\begin{aligned}
u_t^{(k+1)} &= \tfrac{1}{7}(6u_{t-1}^{(k+1)} + u_{t+6}^{(k)}) \ , \\
u_{t+1}^{(k+1)} &= \tfrac{1}{7}(5u_{t-1}^{(k+1)} + 2u_{t+6}^{(k)}) \ , \\
u_{t+2}^{(k+1)} &= \tfrac{1}{7}(4u_{t-1}^{(k+1)} + 3u_{t+6}^{(k)}) \ , \\
u_{t+3}^{(k+1)} &= \tfrac{1}{7}(3u_{t-1}^{(k+1)} + 4u_{t+6}^{(k)}) \ , \\
u_{t+4}^{(k+1)} &= \tfrac{1}{7}(2u_{t-1}^{(k+1)} + 5u_{t+6}^{(k)}) \ , \\
\text{and} \quad u_{t+5}^{(k+1)} &= \tfrac{1}{7}(u_{t-1}^{(k+1)} + 6u_{t+6}^{(k)}) \ .
\end{aligned}\right\} \qquad (6.2.20)$$

where $t=1,(6),m-5$.

6.2.5 THE 8-POINT GROUP ITERATIVE SCHEME

In this scheme, the mesh points of the interval of Figure (6.1.1) are ordered in groups of eight, where each group G_ℓ of Definition (3.11.1), $\ell=1,2,\ldots,m/8$, consists of 8 elements $\{x_t,x_{t+1},x_{t+2},x_{t+3},x_{t+4},x_{t+5},x_{t+6},x_{t+7}\}$ and m must be divisible by 8. Similar to the previous schemes, the system (6.2.4) can be converted into an 8-point group iterative scheme, the subdiagonal matrices D_i, i=1,2,...,m/8, of the matrix B_1 are of the form,

$$D_i = \begin{pmatrix} 1 & -\frac{1}{2} & & & & & & \\ -\frac{1}{2} & 1 & -\frac{1}{2} & & & & 0 & \\ & -\frac{1}{2} & 1 & -\frac{1}{2} & & & & \\ & & -\frac{1}{2} & 1 & -\frac{1}{2} & & & \\ & & & -\frac{1}{2} & 1 & -\frac{1}{2} & & \\ & 0 & & & -\frac{1}{2} & 1 & -\frac{1}{2} & \\ & & & & & -\frac{1}{2} & 1 & -\frac{1}{2} \\ & & & & & & -\frac{1}{2} & 1 \end{pmatrix}, \quad i=1,2,\ldots,m/8,$$

and again by using the method given in subsection (6.2.2), B_1^{-1} can be determined and is given by,

$$B_1^{-1} = \frac{2}{9}\left(\begin{array}{c:c:c} \begin{matrix} 8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 \\ 7 & 14 & 12 & 10 & 8 & 6 & 4 & 2 \\ 6 & 12 & 18 & 15 & 12 & 9 & 6 & 3 \\ 5 & 10 & 15 & 20 & 16 & 12 & 8 & 4 \\ 4 & 8 & 12 & 16 & 20 & 15 & 10 & 5 \\ 3 & 6 & 9 & 12 & 15 & 18 & 12 & 6 \\ 2 & 4 & 6 & 8 & 10 & 12 & 14 & 7 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{matrix} & & 0 \\ \hdashline & \ddots & \\ \hdashline 0 & & \ddots \end{array}\right) \qquad (6.2.21)$$

Hence, we can determine $B_1^{-1}B_2$ and $B_1^{-1}\underline{b}$ explicitly, and the 8-point group explicit G.S. iterative method is given by,

$$u_{t-1+i}^{(k+1)} = \frac{1}{9}[(9-i)u_{t-1}^{(k+1)} + iu_{t+8}^{(k)}] \ , \quad i=1,2,\ldots,8, \tag{6.2.22}$$

where t=1,(8),m-7.

Related to this method is the 8-point group explicit S.O.R. iterative method which replaces (6.2.22) by,

$$u_{t-1+i}^{(k+1)} = \frac{1}{9}\omega[(9-i)u_{t-1}^{(k+1)} + iu_{t+8}^{(k)}] + (1-\omega)u_{t-1+i}^{(k)} \ , \ i=1,2,\ldots,8. \tag{6.2.23}$$

Similar sets of equations can be written down for the Jacobi and J.O.R. iterative schemes.

6.2.6 THE 12-POINT GROUP ITERATIVE SCHEME

Finally, we have the 12-point group, where the mesh points of the interval of Figure (6.1.1) are ordered in groups of 12. By converting the system (6.2.4) into a 12-point group iterative scheme and following the same technique explained before, D_i^{-1}, i=1,2,...,m/12 with m is divisible by 12, can be obtained and can be written as,

$$D_i^{-1} = \frac{2}{13}\begin{pmatrix} 12 & 11 & 10 & 9 & 8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 \\ 11 & 22 & 20 & 18 & 16 & 14 & 12 & 10 & 8 & 6 & 4 & 2 \\ 10 & 20 & 30 & 27 & 24 & 21 & 18 & 15 & 12 & 9 & 6 & 3 \\ 9 & 18 & 27 & 36 & 32 & 28 & 24 & 20 & 16 & 12 & 8 & 4 \\ 8 & 16 & 24 & 32 & 40 & 35 & 30 & 25 & 20 & 15 & 10 & 5 \\ 7 & 14 & 21 & 28 & 35 & 42 & 36 & 30 & 24 & 18 & 12 & 6 \\ 6 & 12 & 18 & 24 & 30 & 36 & 42 & 35 & 28 & 21 & 14 & 7 \\ 5 & 10 & 15 & 20 & 25 & 30 & 35 & 40 & 32 & 24 & 16 & 8 \\ 4 & 8 & 12 & 16 & 20 & 24 & 28 & 32 & 36 & 27 & 18 & 9 \\ 3 & 6 & 9 & 12 & 15 & 18 & 21 & 24 & 27 & 30 & 20 & 10 \\ 2 & 4 & 6 & 8 & 10 & 12 & 14 & 16 & 18 & 20 & 22 & 11 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \end{pmatrix} ,i=1,2,\ldots,m/12,$$

B_1^{-1} follows, since,

$$B_1^{-1} = \begin{bmatrix} D_1^{-1} & & & \\ & D_2^{-1} & & O \\ & & \ddots & \\ O & & & D_{\frac{m}{12}}^{-1} \end{bmatrix} ,$$

$B_1^{-1}B_2$ and $B_1^{-1}\underline{b}$ can then be determined explicitly, and the 12-point group explicit G.S. iterative method is given by,

$$u_{t-1+i}^{(k+1)} = \frac{1}{13}\,[(13-i)u_{t-1}^{(k+1)} + iu_{t+12}^{(k)}] \;,\; i=1,2,\ldots,12, \qquad (6.2.24)$$

where t=1,(12),m-11. Finally, the group S.O.R. iterative method can be written as,

$$u_{t-1+i}^{(k+1)} = \frac{1}{13}\,\omega[(13-i)u_{t-1}^{(k+1)} + iu_{t+12}^{(k)}] + (1-\omega)u_{t-1+i}^{(k)},$$
$$i=1,2,\ldots,12. \qquad (6.2.25)$$

6.2.7 EXPERIMENTAL RESULTS AND COMPUTATIONAL COMPLEXITY

We present here some numerical experiments in order to choose the most efficient group explicit iterative method which were described in subsections (6.2.1), (6.2.2),...,(6.2.6).

The problem is that of solving the two-point boundary value problem (6.2.1)-(6.2.2), where the n-point group explicit Jacobi, Gauss-Seidel and S.O.R. iterative methods were carried out. If we assume that $\rho(L_n)$ is the spectral radius of the n-point group Gauss-Seidel iterative scheme then the optimum relaxation factor, ω_b, for the n-point group S.O.R. method can be calculated using the formula

$$\omega_b = \frac{2}{1+\sqrt{1-\rho(L_n)}} \;, \qquad (6.2.26)$$

where $\rho(L_n)$ can be estimated using the power method (see (3.9.7)-(3.9.9)).

The formula (4.3.4) can also be used to estimate the spectral radius for the n-point explicit group Jacobi scheme $\rho(J)$.

The convergence test used in the numerical experiments was the average test, i.e. the iteration was continued until the following conditions were satisfied,

$$\frac{||u_i^{(k+1)}-u_i^{(k)}||}{1+||u_i^{(k)}||} < \varepsilon \quad \text{for all } i, \tag{6.2.27}$$

where $\varepsilon=10^{-6}$.

The experimental optimum values of ω for the group S.O.R. iterative method are determined to within ±0.001 by solving the problem for a range of values of ω and choosing those which give the minimum number of iterations. The theoretical number of iterations can be obtained by the formula,

$$k \simeq \frac{\ln\varepsilon}{\ln(\omega_b-1)} \quad . \tag{6.2.28}$$

The results obtained are recorded in Tables (6.2.1),(6.2.2) and (6.2.3).

For each method, the graph plot of log(k) versus log(h^{-1}), where k is the minimum number of iterations and h is the mesh size, are shown to be straight lines with a slope of unity, see Figure (6.2.1), thus verifying the S.O.R. theory.

Method \ h^{-1}	25			49		
	$\rho(J_n)$		No.of iterations	$\rho(J_n)$		k
	Theory	Experiment	k	Th.	Exp.	
2-Point Group	0.98421	0.98453	482	0.99588	0.99545	1524
3-Point Group	0.97631	0.97579	340	0.99383	0.99217	1084
4-Point Group	0.96842	0.97001	266	0.99178	0.99281	849
6-Point Group	0.95263	0.95437	190	0.98767	0.98791	603
8-Point Group	0.93683	0.93386	161	0.98356	0.98479	472
12-Point Group	0.90525	0.9042	113	0.97534	0.97645	338

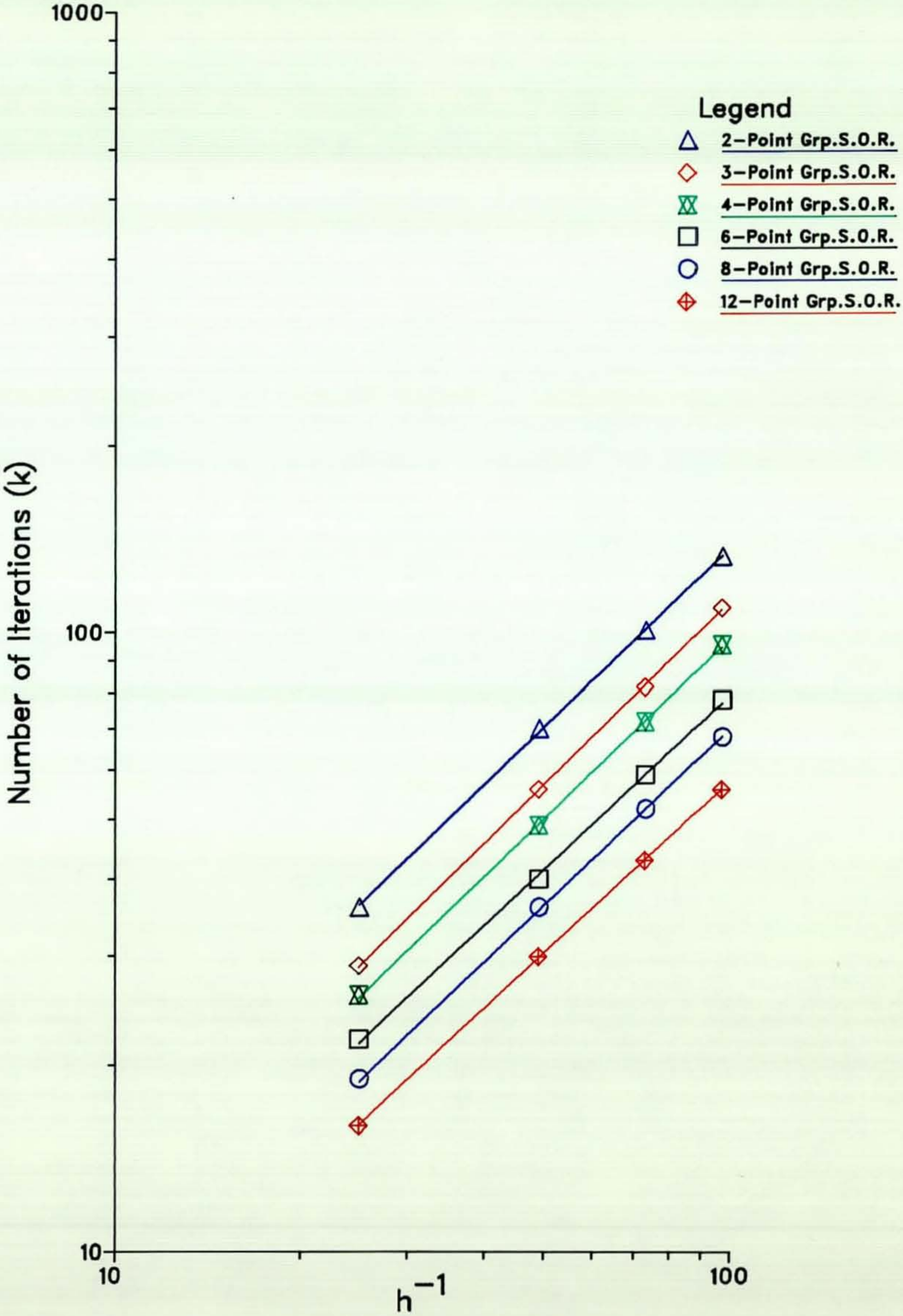
TABLE 6.2.1: The group Jacobi iterative method

Method \ h^{-1}	25			49		
	$\rho(L_n)$		k	$\rho(L_n)$		k
	Th.	Exp.		Th.	Exp.	
2-Point	0.96867	0.96841	264	0.99178	0.99189	847
3-Point	0.95318	0.95585	186	0.9877	0.98873	599
4-Point	0.93784	0.94093	145	0.98363	0.9843	468
6-Point	0.9075	0.91234	103	0.97549	0.97516	330
8-Point	0.87765	0.88901	81	0.96739	0.96782	259
12-Point	0.81948	0.85034	62	0.95129	0.95591	185

TABLE 6.2.2: The group Gauss-Seidel iterative method

Method \ h^{-1}	25				49				73				97			
	Optimum ω (ω_b)		No.of iterations k		ω_b		k		ω_b		k		ω_b		k	
	Th.	Exp.	Th.	Exp.	Th.	Exp.	Th.	Exp.	Th.	Exp.	Th.	Exp.	Th.	Exp.	Th.	Exp.
2-Point Group	1.699	1.707 & 1.708	39	36	1.834	1.84	76	70	1.885	1.887	113	101	1.912	1.914	151	133
3-Point Group	1.644	1.658	31	29	1.8	1.805 & 1.806	62	56	1.861	1.865	92	82	1.894	1.897 & 1.898	123	110
4-Point Group	1.601	1.612 ⋮ 1.619	27	26	1.773	1.778 ⋮ 1.781	54	49	1.842	1.844 ⋮ 1.846	80	72	1.879	1.88 & 1.881	107	96
6-Point Group	1.534	1.551 ⋮ 1.562	22	22	1.73	1.736 & 1.737	44	40	1.81	1.813	66	59	1.853	1.855 & 1.856	87	78
8-Point Group	1.482	1.509 ⋮ 1.517	19	19	1.694	1.701 ⋮ 1.709	38	36	1.783	1.787 ⋮ 1.79	56	52	1.832	1.835 & 1.836	75	68
12-Point Group	1.404	1.454 ⋮ 1.455	13	16	1.639	1.651 ⋮ 1.661	31	30	1.741	1.747 ⋮ 1.751	46	43	1.799	1.802 & 1.803	61	56

TABLE 6.2.3: The group S.O.R. iterative method



1000
100
10
Number of Iterations (k)
10
100
h^{-1}
Legend
2-Point Grp.S.O.R.
3-Point Grp.S.O.R.
4-Point Grp.S.O.R.
6-Point Grp.S.O.R.
8-Point Grp.S.O.R.
12-Point Grp.S.O.R.

Figure (6.2.1)

We now consider the relative efficiency of the various iterative methods which we have discussed for solving the two-point boundary value problem (6.2.1)-(6.2.2). In order to estimate the amount of computational work to obtain the solution, it is necessary to measure the *computational complexity* for each iteration and the number of iterations needed to obtain the solution.

First, we determine the amount of computational work required to obtain the inverse of the matrix B_1 given in (6.1.11a), B_1^{-1} has the form,

$$B_1^{-1} = \begin{bmatrix} D_1^{-1} & & & \\ & D_2^{-1} & & O \\ & & \ddots & \\ O & & & D_{\frac{m}{n}}^{-1} \end{bmatrix} \tag{6.2.29}$$

where n is the size of the group and m is a multiple of n

$$D_1^{-1} = D_2^{-1} = \ldots = D_{\frac{m}{n}}^{-1} = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1n} \\ \alpha_{21} & \alpha_{22} & \cdots & \alpha_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{n1} & \alpha_{n2} & \cdots & \alpha_{nn} \end{bmatrix} . \tag{6.2.30}$$

From the analysis given earlier, it can be seen that we only need to calculate the first and the last columns of D_i^{-1}, moreover, since $\alpha_{11}=\alpha_{nn}$, $\alpha_{21}=\alpha_{n-1,n}$, $\ldots,\alpha_{n1}=\alpha_{1n}$, then only one column of D_i^{-1} is needed to be calculated for this problem. By applying the method illustrated on page(266), we require per iteration,

$$(1 \text{ multiplication} + 4 \text{ additions}) \times n \quad . \tag{6.2.31}$$

We now calculate the number of operations per iteration needed to solve the problem by using the 2,3,4,6,8 and 12 point group S.O.R. iterative methods assuming that there are m internal mesh points.

For the *2-point group*, the $(k+1)^{th}$ iterate of the S.O.R. iterative scheme is given by,

$$\left.\begin{aligned} u_i^{(k+1)} &= \omega(\tilde{u}_i^{(k+1)} - u_i^{(k)}) + u_i^{(k)} \quad , \\ \text{and} \quad u_{i+1}^{(k+1)} &= \omega(\tilde{u}_{i+1}^{(k+1)} - u_{i+1}^{(k)}) + u_{i+1}^{(k)} \quad , \end{aligned}\right\} i=1,(2),m-1, \tag{6.2.32}$$

where the $\tilde{u}_i^{(k+1)}$ and $\tilde{u}_{i+1}^{(k+1)}$ are the components of the $(k+1)^{th}$ G.S. iteration (6.2.7) which can be written more efficiently as,

$$\left.\begin{aligned} \tilde{u}_i^{(k+1)} &= \frac{1}{3}(u_{i-1}^{(k+1)} + u_{i-1}^{(k+1)} + u_{i+2}^{(k)}) \quad , \\ \tilde{u}_{i+1}^{(k+1)} &= \frac{1}{3}(u_{i-1}^{(k+1)} + u_{i+2}^{(k)} + u_{i+2}^{(k)}) \quad . \end{aligned}\right\} i=1,(2),m-1 \tag{6.2.33}$$

Hence, the number of operations required per iteration, assuming that the constant 1/3 is stored beforehand, is

$$2m \text{ multiplications} + 4m \text{ additions} . \tag{6.2.34}$$

The S.O.R. iterative method for the *3-point group* is defined by,

$$u_j^{(k+1)} = (\tilde{u}_j^{(k+1)} - u_j^{(k)}) + u_j^{(k)}, \; j=i,i+1 \text{ and } i+2 , \tag{6.2.35}$$
$$i=1,(3),m-2.$$

where $\tilde{u}_j^{(k+1)}$ are the components of the $(k+1)^{th}$ G.S. iteration which is given by,

$$\left.\begin{aligned} \tilde{u}_i^{(k+1)} &= 0.25(3u_{i-1}^{(k+1)} + u_{i+3}^{(k)}), \\ \tilde{u}_{i+1}^{(k+1)} &= 0.25(u_{i-1}^{(k+1)} + u_{i-1}^{(k+1)} + u_{i+3}^{(k)} + u_{i+3}^{(k)}) \quad , \\ \text{and} \quad \tilde{u}_{i+2}^{(k+1)} &= 0.25(u_{i-1}^{(k+1)} + 3u_{i+3}^{(k)}) \quad . \end{aligned}\right\} \tag{6.2.36}$$

Hence, this process requires for each iteration,

$$\frac{8}{3} m \text{ multiplications} + \frac{11}{3} m \text{ additions} . \tag{6.2.37}$$

For the remaining groups, i.e. the 4,6,8 and 12 point and from the equation given in (6.2.18), (6.2.20), (6.2.22) and (6.2.24) which represent the $(k+1)^{th}$ iterate of the G.S. method respectively, it can be seen that the method requires per iteration,

$$2m \text{ multiplications} + 1m \text{ additions.}$$

In addition, since the S.O.R. iterative scheme requires an extra 1 multiplication and 2 additions for each point per iteration. Hence each of the 4,6,8 and 12 point group S.O.R. iterative schemes requires for each iteration,

$$3m \text{ multiplications} + 3m \text{ additions.} \tag{6.2.38}$$

Hence, the total number of operations required to solve the problem can be obtained by adding (6.2.31) to (6.2.34), (6.2.37) and (6.2.38) and the final result given in Table (6.2.4).

Method	No. of operations in each iteration (excluding conv.test)	
	Multiplications	Additions
2-Point Group	$2m+2$	$4m+8$
3-Point Group	$\frac{8}{3}m+3$	$\frac{11}{3}m+12$
4-Point Group	$3m+4$	$3m+16$
6-Point Group	$3m+6$	$3m+24$
8-Point Group	$3m+8$	$3m+32$
12-Point Group	$3m+12$	$3m+48$

TABLE 6.2.4

The experimental results for the number of iterations given in Table (6.2.3) can be combined with the number of operations in each iteration required by each of the methods to give the total number of arithmetic operations required for a solution, these are recorded in Table (6.2.5). The execution times of the different groups are recorded in Table (6.2.6).

Method \ h^{-1}	25		49		73		97	
	Multiplications	Additions	Multiplications	Additions	Multiplications	Additions	Multiplications	Additions
2-Point Group SOR	72m+72	144m+288	140m+140	280m+560	202m+202	404m+808	266m+266	532m+1064
3-Point Group SOR	77.3m+87	106.3m+348	149.3m+168	205.3m+672	218.68m+246	300.67m+984	293.3m+330	403.3m+1320
4-Point Group SOR	78m+104	78m+416	147m+196	147m+784	216m+288	216m+1152	288m+384	288m+1536
6-Point Group SOR	66m+132	66m+528	120m+240	120m+960	177m+354	177m+1416	234m+468	234m+1872
8-Point Group SOR	57m+152	57m+608	108m+288	108m+1152	156m+416	156m+1664	204m+544	204m+2176
12-Point Group SOR	48m+192	48m+768	90m+360	90m+1440	129m+516	129m+2064	168m+672	168m+2688

TABLE 6.2.5

Method / h^{-1}	2-Point Group SOR	3-Point Group SOR	4-Point Group SOR	6-Point Group SOR	8-Point Group SOR	12-Point Group SOR
25	9	9	12	10	9	8
49	36	31	45	36	32	27
73	76	69	99	79	70	58
97	142	130	180	144	131	103

TABLE 6.2.6: The execution time in centisec.for the optimum value of ω

This table indicates that the 3,8 and 12 point groups require less computer time than the other groups to solve the two-point boundary value problem (6.2.1). However, if a more general problem with variable coefficients depending on the position x was considered, then the extra work involved in group inversions would present an entirely different situation (Sojoodi-Haghighi, (1981)).

On the other hand, if we confine our attention to methods involving the inversion of small groups and the determination of new explicit iterative methods then it can be verified that the 3-point group approach is more attractive than the 8 or 12 point groups since it is easier to program and obviously simpler to invert. Also it offers some gains in computational efficiency over the 2,4 and 6 point groups.

6.3 NON-LINEAR BOUNDARY VALUE PROBLEMS

6.3.1 THE 2,3 AND 4-POINT GROUP ITERATIVE METHODS

In this section, some non-linear boundary value problems are solved by transforming them into an iterative sequence of equations which can then be solved by applying the 2,3 and 4 point explicit groups methods discussed earlier in subsections (6.2.1), (6.2.2) and (6.2.3) respectively.

Problem 1

The first non-linear problem we consider is

$$\frac{d^2U}{dx^2} = \frac{1}{8}(32 + 2x^3 - U\frac{dU}{dx}) \;,\; 1\leq x\leq 3 \tag{6.3.1}$$

subject to the boundary conditions

$$U(1) = 17 \;,\quad U(3) = 43/3 \;. \tag{6.3.2}$$

This problem has the exact solution,

$$U(x) = x^2 + \frac{16}{x} \;. \tag{6.3.3}$$

By following the finite difference procedure of Section (6.1) to approximate equation (6.3.1), we obtain the non-linear difference equation,

$$\frac{u_{i-1}-2u_i+u_{i+1}}{h^2} = \frac{1}{8}[32+2x_i^3-u_i(\frac{u_{i+1}-u_{i-1}}{2h})] \;,\; i=1,2,\ldots,m \tag{6.3.4}$$

and the boundary conditions are replaced by,

$$u_0 = 17, \quad \text{and} \quad u_{m+1} = 43/3 \;, \tag{6.3.5}$$

where h is the grid spacing given by (see equation 6.1.3)

$$h = \frac{2}{m+1} \;. \tag{6.3.6}$$

Equation (6.3.4) can be written in the form,

$$-\frac{1}{2}(1-\frac{h}{16}u_i)u_{i-1}+u_i-\frac{1}{2}(1+\frac{h}{16}u_i)u_{i+1} = -2h^2(1+\frac{1}{16}x_i^3),\; i=1,2,\ldots,m \tag{6.3.7}$$

this can be rewritten iteratively by the matrix system,

$$\begin{bmatrix} 1 & -g_1 & & & & \\ -f_2 & 1 & -g_2 & & \bigcirc & \\ & -f_3 & 1 & -g_3 & & \\ & & \ddots & \ddots & \ddots & \\ & \bigcirc & & -f_{m-1} & 1 & -g_{m-1} \\ & & & & -f_m & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{m-1} \\ u_m \end{bmatrix}^{(k+1)} = \begin{bmatrix} 17f_1+c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_{m-1} \\ \frac{43}{3}g_m+c_m \end{bmatrix}, \qquad (6.3.8a)$$

where $g_i = g(u_i)=\frac{1}{2}(1+\frac{h}{16}u_i^{(k)})$, $f_i=f(u_i)=\frac{1}{2}(1-\frac{h}{16}u_i^{(k)})$ and $c_i=-2h^2(1+\frac{1}{16}x_i^3)$,

$$i=1,2,\ldots,m. \qquad (6.3.8b)$$

Each iteration of this nonlinear system can be directly solved using Picard's algorithm. The iteration proceeds until the following condition was achieved,

$$\frac{||u_i^{(k+1)}-u_i^{(k)}||}{1+||u_i^{(k)}||} < \varepsilon \qquad \text{for all } i, \qquad (6.3.9)$$

where $\varepsilon=0.00001$.

Alternatively, the system (6.3.8a) can be decoupled into a 2 point group system or into a 3 or 4 point group system according to the partitioning strategy chosen.

The 2-Point Group Iterative Strategy

We now decouple the system (6.3.8a) into a two point group iterative strategy in the following manner,

$$\begin{bmatrix} 1 & -g_1 & & & & & & \\ -f_2 & 1 & & & & & 0 & \\ & & 1 & -g_3 & & & & \\ & & -f_4 & 1 & & & & \\ & & & & \ddots & & & \\ & 0 & & & & \ddots & & \\ & & & & & & 1 & -g_{m-1} \\ & & & & & & -f_m & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ \vdots \\ u_{m-1} \\ u_m \end{bmatrix}^{(k+1)} = \begin{bmatrix} 0 & 0 & & & & & & 0 \\ 0 & g_2 & 0 & & & & & \\ 0 & f_3 & & & & & & \\ 0 & 0 & 0 & g_4 & & & & \\ & & & f_5 & \ddots & & & \\ & & & & \ddots & & g_{m-1} & \\ & 0 & & & & f_{m-1} & & 0 \end{bmatrix}$$

$$\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ \vdots \\ u_{m-1} \\ u_m \end{bmatrix}^{(k)} + \begin{bmatrix} 17f_1+c_1 \\ c_2 \\ \vdots \\ \vdots \\ c_{m-1} \\ \frac{43}{3}\, g_m+f_m \end{bmatrix} , \qquad (6.3.10)$$

which in matrix form is,

$$B_1\underline{u}^{(k+1)} = B_2\underline{u}^{(k)} + \underline{b} \, . \qquad (6.3.11)$$

Since the matrix B_1 consists of a series of 2×2 block subsystems. Then each can be easily converted to explicit form in the following way,

$$\underline{u}^{(k+1)} = B_1^{-1}B_2\underline{u}^{(k)} + B_1^{-1}\underline{b} \, , \qquad (6.3.12)$$

where,

$$B_1^{-1} = \begin{bmatrix} d_1 & g_1d_1 & & & & \\ f_2d_1 & d_1 & & & 0 & \\ & & d_2 & g_3d_2 & & \\ & & f_4d_2 & d_2 & & \\ & & & & \ddots & \\ & 0 & & & d_{\frac{m}{2}} & g_{m-1}d_{\frac{m}{2}} \\ & & & & f_m d_{\frac{m}{2}} & d_{\frac{m}{2}} \end{bmatrix} \qquad (6.3.13a)$$

and $\quad d_j=1/(1-g_{2j-1}f_{2j}) \quad , \quad j=1,2,\ldots,\frac{m}{2}$ for m even. (6.3.13b)

In explicit form, the system is now

$$
\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ \vdots \\ \vdots \\ u_{m-1} \\ u_m \end{pmatrix}^{(k+1)}
=
\left(\begin{array}{cc|cc|cc|c|cc|cc}
O & & g_1g_2d_1 & 0 & & & & & & & \\
 & & g_2d_1 & 0 & & & & & & & \\
\hline
0 & f_3d_2 & O & & g_3g_4d_2 & 0 & & & & & \\
0 & f_3f_4d_2 & & & g_4d_2 & 0 & & & & & \\
\hline
 & & 0 & f_5d_3 & O & & \ddots & & & & \\
 & & 0 & f_5f_6d_3 & & & & \ddots & & & \\
\hline
 & & & & \ddots & & & \ddots & & & \\
 & & & & & \ddots & & & \ddots & g_{m-3}g_{m-2}d_{m/2-1} & 0 \\
 & & & & & & & & \ddots & g_{m-2}d_{m/2-1} & 0 \\
\hline
 & & & & & & & & f_{m-1}d_{m/2} & O & \\
 & & & & & & & & f_{m-1}f_md_{m/2} & &
\end{array}\right)
\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ \vdots \\ \vdots \\ u_{m-1} \\ u_m \end{pmatrix}^{(k)}
+
\begin{pmatrix}
(c_1+17f_1+g_1c_2)d_1 \\
(f_2(c_1+17f_1)+c_2)d_1 \\
(c_3+g_3c_4)d_2 \\
(f_4c_3+c_4)d_2 \\
\vdots \\
\vdots \\
[c_{m-1}+g_{m-1}(c_m+\frac{43}{3}g_m)]d_{m/2} \\
(f_mc_{m-1}+c_m+\frac{43}{3}g_m)d_{m/2}
\end{pmatrix}
\qquad (6.3.14)
$$

which can be solved by the Jacobi, J.O.R., Gauss-Seidel or N.L.O.R. iterative methods.

<u>The 3-Point Group Iterative Strategy</u>

For this strategy we convert the system (6.3.8a) into a 3-point group iterative scheme as follows,

$$
\left[\begin{array}{ccc|ccc|c|ccc}
1 & -g_1 & & & & & & & & \\
-f_2 & 1 & -g_2 & & & & & & 0 & \\
 & -f_3 & 1 & & & & & & & \\
\hline
 & & & 1 & -g_4 & & & & & \\
 & & & -f_5 & 1 & -g_5 & & & & \\
 & & & & -f_6 & 1 & & & & \\
\hline
 & & & & & & \ddots & & & \\
\hline
 & 0 & & & & & & 1 & -g_{m-2} & \\
 & & & & & & & -f_{m-1} & 1 & -g_{m-1} \\
 & & & & & & & & -f_m & 1
\end{array}\right]
\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{m-2} \\ u_{m-1} \\ u_m \end{bmatrix}^{(k+1)}
=
$$

$$
\left[\begin{array}{ccc|ccc|c|ccc}
 & & & 0 & 0 & 0 & & & & \\
 & 0 & & 0 & 0 & 0 & & & 0 & \\
 & & & g_3 & 0 & 0 & & & & \\
\hline
0 & 0 & f_4 & & & & & & & \\
0 & 0 & 0 & & 0 & & & & & \\
0 & 0 & 0 & & & & g_6 & & & \\
\hline
 & & & & & f_7 & \ddots & & & \\
 & & & & & & \ddots & g_{m-3} & & \\
\hline
 & 0 & & & & & f_{m-2} & & & \\
 & & & & & & & & 0 & \\
 & & & & & & & & &
\end{array}\right]
\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{m-2} \\ u_{m-1} \\ u_m \end{bmatrix}^{(k)}
+
\begin{bmatrix} 17f_1+c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_{m-2} \\ c_{m-1} \\ \frac{43}{3} g_m+c_m \end{bmatrix}
\qquad (6.3.15)
$$

This system has the matrix form (6.3.11), where the matrix B_1 can easily be inverted, and is given by,

$$B_1^{-1}=\left[\begin{array}{c|c|c}
\begin{matrix}(1-g_2f_3)d_1 & g_1d_1 & g_1g_2d_1\\ f_2d_1 & d_1 & g_2d_1\\ f_2f_3d_1 & f_3d_1 & (1-g_1f_2)d_1\end{matrix} & & O\\ \hline
 & \ddots & \\ \hline
O & & \begin{matrix}(1-g_{m-1}f_m)d_{\frac{m}{3}} & g_{m-2}d_{\frac{m}{3}} & g_{m-2}g_{m-1}d_{\frac{m}{3}}\\ f_{m-1}d_{\frac{m}{3}} & d_{\frac{m}{3}} & g_{m-1}d_{\frac{m}{3}}\\ f_{m-1}f_md_{\frac{m}{3}} & f_md_{\frac{m}{3}} & (1-g_{m-2}g_{m-1})d_{\frac{m}{3}}\end{matrix}
\end{array}\right]$$

(6.3.16)

and $d_j=1/(1-g_{3j-1}f_{3j}-g_{3j-2}f_{3j-1})$, $j=1,2,\ldots,\frac{m}{3}$ (m is divisible by 3). Hence the system (6.3.15) can be converted to an explicit form, which again can be solved by Jacobi, J.O.R., Gauss-Seidel or N.L.O.R. iterative methods. The converted system has the matrix form (6.3.12) where $B_1^{-1}B_2$ and $B_1^{-1}\underline{b}$ are given by,

$$B_1^{-1}B_2=\left[\begin{array}{c|c|c|c}
O & \begin{matrix}g_1g_2g_3d_1 & 0 & 0\\ g_2g_3d_1 & 0 & 0\\ (1-g_1f_2)g_3d_1 & 0 & 0\end{matrix} & & O\\ \hline
\begin{matrix}0 & 0 & (1-g_5f_6)f_4d_2\\ 0 & 0 & f_4f_5d_2\\ 0 & 0 & f_4f_5f_6d_2\end{matrix} & O & \begin{matrix}g_4g_5g_6d_2\\ g_5g_6d_2\\ (1-g_4f_5)g_6d_2\end{matrix} & O\\ \hline
 & & \ddots & \\ \hline
O & & \begin{matrix}(1-g_{m-1}f_m)f_{m-2}d_{\frac{m}{3}}\\ f_{m-1}f_{m-2}d_{\frac{m}{3}}\\ f_mf_{m-1}f_{m-2}d_{\frac{m}{3}}\end{matrix} & O
\end{array}\right]$$

(6.3.17a)

$$
B_1^{-1}\underline{b} = \begin{bmatrix}
[(1-g_2f_3)(17f_1+c_1)+g_1c_2+g_1g_2c_3]d_1 \\
[f_2(17f_1+c_1)+c_2+g_2c_3]d_1 \\
[f_2f_3(17f_1+c_1)+f_3c_2+(1-g_1f_2)c_3]d_1 \\
[(1-g_5f_6)c_4+g_4c_5+g_4g_5c_6]d_2 \\
(f_5c_4+c_5+g_5c_6)d_2 \\
[f_5f_6c_4+f_6c_5+(1-g_4f_5)c_6]d_2 \\
\vdots \\
[(1-g_{m-1}f_m)c_{m-2}+g_{m-2}c_{m-1}+(g_{m-2}g_{m-1})(\frac{43}{3}g_m+c_m)]d_{\frac{m}{3}} \\
[f_{m-1}c_{m-2}+c_{m-1}+g_{m-1}(\frac{43}{3}\,g_m+c_m)]d_{\frac{m}{3}} \\
[f_{m-1}f_mc_{m-2}+f_mc_{m-1}+(1-g_{m-2}f_{m-1})(\frac{43}{3}\,g_m+c_m)]d_{\frac{m}{3}}
\end{bmatrix} \qquad (6.3.17b)
$$

<u>The 4-Point Group Iterative Strategy</u>

The system (6.3.8a), in this case, is converted into a 4 point group iterative strategy and we have the system,

$$
\left[\begin{array}{cccc:c:cccc}
1 & -g_1 & & 0 & & & & & \\
-f_2 & 1 & -g_2 & & & & & 0 & \\
& -f_3 & 1 & -g_3 & & & & & \\
0 & & -f_4 & 1 & & & & & \\
\hdashline
& & & & \ddots & & & & \\
\hdashline
& & & & & 1 & -g_{m-3} & & 0 \\
& & 0 & & & -f_{m-2} & 1 & -g_{m-2} & \\
& & & & & & -f_{m-1} & 1 & -g_{m-1} \\
& & & & & 0 & & -f_m & 1
\end{array}\right]
\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{m-2} \\ u_{m-1} \\ u_m \end{bmatrix}^{(k+1)} =
$$

$$\left[\begin{array}{c|c|c} 0 & 0 & 0 \\ & g_4 & \\ \hline f_5 & \ddots\quad\ddots & \\ 0 & \ddots\quad\ddots & 0 \\ & & g_{m-4} \\ \hline & f_{m-3} & \\ 0 & 0 & 0 \end{array}\right] \begin{bmatrix} u_1\\ u_2\\ u_3\\ \vdots \\ u_{m-2}\\ u_{m-1}\\ u_m\end{bmatrix}^{(k)} + \begin{bmatrix}17f_1+c_1\\ c_2\\ c_3\\ \vdots\\ c_{m-2}\\ c_{m-1}\\ \frac{43}{3}g_m+c_m\end{bmatrix} \tag{6.3.18}$$

which has the matrix form (6.3.11). The matrix B_1 can be inverted where B_1^{-1} has the form,

$$B_1^{-1} = \begin{bmatrix} D_1 & & & \\ & D_2 & & 0 \\ & & \ddots & \\ 0 & & & D_n \end{bmatrix} \tag{6.3.19}$$

where $n=\frac{m}{4}$, and each submatrix D_j $(j=1,2,\ldots,n)$ is a (4×4) matrix given by,

$$D_j = d_j \begin{bmatrix} 1-g_{4j-2}f_{4j-1}-g_{4j-1}f_{4j} & g_{4j-3}(1-g_{4j-1}f_{4j}) & g_{4j-3}g_{4j-2} & g_{4j-3}g_{4j-2}g_{4j-1} \\ f_{4j-2}(1-g_{4j-1}f_{4j}) & 1-g_{4j-1}f_{4j} & g_{4j-2} & g_{4j-2}g_{4j-1} \\ f_{4j-2}f_{4j-1} & f_{4j-1} & 1-g_{4j-3}f_{4j-2} & g_{4j-1}(1-g_{4j-3}f_{4j-2}) \\ f_{4j-2}f_{4j-1}f_{4j} & f_{4j-1}f_{4j} & f_{4j}(1-g_{4j-3}f_{4j-2}) & 1-g_{4j-3}f_{4j-2}-g_{4j-2}f_{4j-1} \end{bmatrix}$$

$$\text{for } j=1,2,\ldots,n, \tag{6.3.20a}$$

where,

$$d_j = 1/[1-g_{4j-3}f_{4j-2}-g_{4j-2}f_{4j-1}-g_{4j-1}f_{4j}(1-g_{4j-3}f_{4j-2})]$$

$$\text{for } j=1,2,\ldots,n. \qquad (6.3.20b)$$

Hence, an explicit form of the system (6.3.17) can be obtained which has the matrix form (6.3.12) where $B_1^{-1}B_2$ and $B_1^{-1}\underline{b}$ are given by,

$$
B_1^{-1}B_2 = \left(\begin{array}{cccc:cccc:cccc:c:cccc:c}
 & & & & g_1g_2g_3g_4d_1 & 0 & 0 & 0 & & & & & & & & & \\
 & 0 & & & g_2g_3g_4d_1 & 0 & 0 & 0 & & & & & & & & & \\
 & & & & g_3g_4(1-g_1f_2)d_1 & 0 & 0 & 0 & & & 0 & & & & & & \\
 & & & & g_4(1-g_1f_2-g_2f_3)d_1 & 0 & 0 & 0 & & & & & & & & & \\
\hdashline
0 & 0 & 0 & f_5(1-g_6f_7-g_7f_8)d_2 & & & & & g_5g_6g_7g_8d_2 & & & & & & & & \\
0 & 0 & 0 & f_5f_6(1-g_7f_8)d_2 & & 0 & & & g_6g_7g_8d_2 & & 0 & & & & & & \\
0 & 0 & 0 & f_5f_6f_7d_2 & & & & & g_7g_8(1-g_5f_6)d_2 & & & & & & & & \\
0 & 0 & 0 & f_5f_6f_7f_8d_2 & & & & & g_8(1-g_5f_6-g_6f_7)d_2 & & & & & & & & \\
\hdashline
 & & & & & & & & & & \ddots & & & & & & \\
\hdashline
 & & & & & & & & & & & & & f_{m-3}(1-g_{m-2}f_{m-1}-g_{m-1}f_m)d_{\frac{m}{4}} & & & \\
 & & & & & & & & & 0 & & & & f_{m-3}f_{m-2}(1-g_{m-1}f_m)d_{\frac{m}{4}} & & 0 & \\
 & & & & & & & & & & & & & f_{m-3}f_{m-2}f_{m-1}d_{\frac{m}{4}} & & & \\
 & & & & & & & & & & & & & f_{m-3}f_{m-2}f_{m-1}f_md_{\frac{m}{4}} & & & \\
\end{array}\right)
$$

(6.3.21a)

and

$$
B_1^{-1}\underline{b} = \begin{pmatrix}
[(c_1+17f_1)(1-g_2f_3-g_3f_4)+g_1c_2(1-g_3f_4)+g_1g_2c_3+g_1g_2g_3c_4]d_1 \\
[f_2(1-g_3f_4)(c_1+17f_1)+(1-g_3f_4)c_2+g_2c_3+g_2g_3c_4]d_1 \\
[f_2f_3(c_1+17f_1)+f_3c_2+c_3(1-g_1f_2)+g_3c_4(1-g_1f_2)]d_1 \\
[f_2f_3f_4(c_1+17f_1)+f_3f_4c_2+f_4c_3(1-g_1f_2)+c_4(1-g_1f_2-g_2f_3)]d_1 \\
\vdots \\
[c_{m-3}(1-g_{m-2}f_{m-1}-g_{m-1}f_m)+g_{m-3}c_{m-2}(1-g_{m-1}f_m)+g_{m-2}g_{m-3}c_{m-1}+(c_m+\frac{43}{3}g_m)g_{m-3}g_{m-2}g_{m-1}]d_{\frac{m}{4}} \\
[f_{m-2}c_{m-3}(1-g_{m-1}f_m)+c_{m-2}(1-g_{m-1}g_m)+g_{m-2}c_{m-1}+(c_m+\frac{43}{3}g_m)g_{m-2}g_{m-1}]d_{\frac{m}{4}} \\
[f_{m-2}f_{m-1}c_{m-3}+f_{m-1}c_{m-2}+c_{m-1}(1-g_{m-3}f_{m-2})+(c_m+\frac{43}{3}g_m)(1-g_{m-3}f_{m-2})g_{m-1}]d_{\frac{m}{4}} \\
[f_{m-2}f_{m-1}f_mc_{m-3}+f_{m-1}f_mc_{m-2}+f_mc_{m-1}(1-g_{m-3}f_{m-2})+(c_m+\frac{43}{3}g_m)(1-g_{m-3}f_{m-2}-g_{m-2}f_{m-1})]d_{\frac{m}{4}}
\end{pmatrix} \qquad (6.3.21b)
$$

The converted system can now be solved by the Jacobi, J.O.R., Gauss-Seidel or N.L.O.R. iterative methods.

Problem 2

We now consider the non-linear problem,

$$U_1' = U_2 \tag{6.3.22a}$$

$$U_2' = U_1^3 - \sin x(1+\sin^2 x) \ , \tag{6.3.22b}$$

subject to the boundary conditions,

$$U_1(0) = U_1(\pi) = 0 \ . \tag{6.3.22c}$$

The exact solution for this problem is given by,

$$U_1(x) = \sin x \ , \qquad U_2(x) = \cos x \ . \tag{6.3.23}$$

From (6.3.22a) and (6.3.22b), we have,

$$U_1'' = U_1^3 - \sin x(1+\sin^2 x) \ , \tag{6.3.24}$$

this non-linear equation can be approximated by applying the finite difference procedure of Section (6.1) to obtain the following non-linear difference equation (assuming that $u=u_1$),

$$\frac{u_{i-1}-2u_i+u_{i+1}}{h^2} = u_i^3-\sin x_i(1+\sin^2 x_i), \ i=1,2,\ldots,m. \tag{6.3.25}$$

the boundary conditions are replaced by,

$$u_0 = 0 \text{ and } u_{m+1} = 0 \ , \tag{6.3.26}$$

and where $h = \frac{\pi}{m+1}$.

Equation (6.3.25) can be simplified to the form,

$$-\frac{1}{2}u_{i-1} + (1+\frac{h^2}{2}u_i^2)u_i - \frac{1}{2}u_{i+1} = \frac{h^2}{2}\sin x_i\ (1+\sin^2 x_i) \ ,$$

$$i=1,2,\ldots,m, \tag{6.3.27}$$

which can be rewritten in the non-linear matrix iterative form

$$\begin{bmatrix} g_1 & -1/2 & & & & \\ -1/2 & g_2 & -1/2 & & \bigcirc & \\ & -1/2 & g_3 & -1/2 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1/2 & g_{m-1} & -1/2 \\ & \bigcirc & & & -1/2 & g_m \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{m-1} \\ u_m \end{bmatrix}^{(k+1)} = \frac{h^2}{2}\begin{bmatrix} \sin x_1(1+\sin^2 x_1) \\ \sin x_2(1+\sin^2 x_2) \\ \sin x_3(1+\sin^2 x_3) \\ \vdots \\ \sin x_{m-1}(1+\sin^2 x_{m-1}) \\ \sin x_m(1+\sin^2 x_m) \end{bmatrix} ,$$

(6.3.28)

where $g_i = g(u_i^{(k)}) = 1 + \frac{h^2}{2}(u_i^{(k)})^2$, $i=1,2,\ldots,m$.

Similar to Problem 1, each iteration of this system can be directly solved using Picard's algorithm. Alternatively, we can convert the above system into a 2,3 or 4 point group iterative strategy, in which the converted system has the matrix form, (6.3.11), where B_1 has the form,

$$B_1 = \begin{bmatrix} R_1 & & & \\ & R_2 & & O \\ & & \ddots & \\ O & & & R_n \end{bmatrix} \qquad (6.3.29a)$$

where,

$$n = \begin{cases} \frac{m}{2}, & \text{for the 2-point group strategy, } m \text{ is even} \\ \frac{m}{3}, & \text{" " 3 " " " , } m \text{ is divisible by 3} \\ \frac{m}{4}, & \text{" " 4 " " " , } m \text{ is divisible by 4} \end{cases}$$

and each R_j, $(j=1,2,\ldots,n)$ is given by,

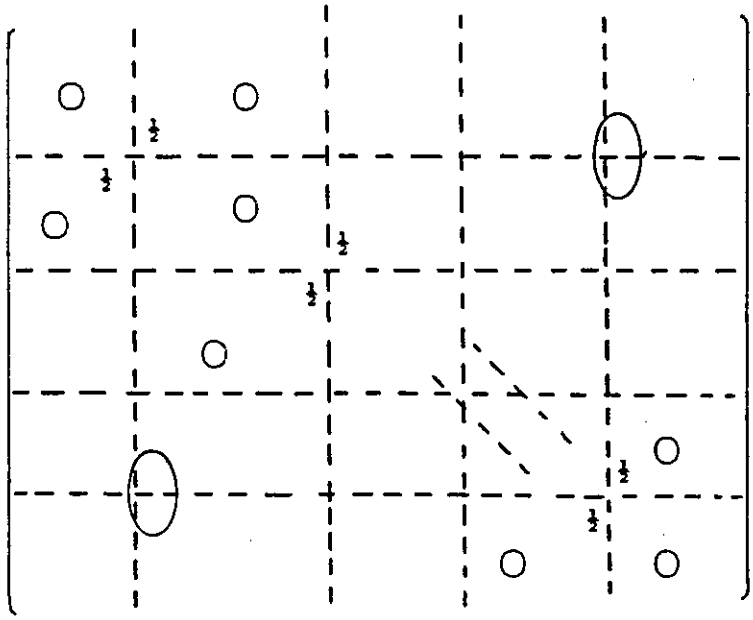
$$R_j = \begin{bmatrix} g_{2j-1} & -1/2 \\ -1/2 & g_{2j} \end{bmatrix}, \quad R_j = \begin{bmatrix} g_{3j-2} & -1/2 & 0 \\ -1/2 & g_{3j-1} & -1/2 \\ 0 & -1/2 & g_{3j} \end{bmatrix}$$

and

$$R_j = \begin{bmatrix} g_{4j-3} & -1/2 & & O \\ -1/2 & g_{4j-2} & -1/2 & \\ & -1/2 & g_{4j-1} & -1/2 \\ O & & -1/2 & g_{4j} \end{bmatrix} \qquad (6.3.29b)$$

for the 2,3 and 4 point group strategies respectively.

The matrix B_2 of (6.3.11) for this problem is of the form,

$B_2 =$  (6.3.30a)

and the r.h.s. vector is given by,

$$\underline{b} = \frac{h^2}{2}\begin{pmatrix} \sin x_1(1+\sin^2 x_1) \\ \sin x_2(1+\sin^2 x_2) \\ \vdots \\ \sin x_{m-1}(1+\sin^2 x_{m-1}) \\ \sin x_m(1+\sin^2 x_m) \end{pmatrix} \tag{6.3.30b}$$

The matrix B_1^{-1} can be easily obtained and as before it is a block diagonal matrix with block entries R_j^{-1}, $(j=1,2,\ldots,n)$.

For the 2,3 and 4-point group iterative strategies, we have respectively,

$$R_j^{-1} = d_j \begin{pmatrix} g_{2j} & 1/2 \\ 1/2 & g_{2j-1} \end{pmatrix} . \tag{6.3.31a}$$

where $d_j=1/(g_{2j}g_{2j-1}-\frac{1}{4})$, for $j=1,2,\ldots,n$.

$$R_j^{-1} = d_j \begin{pmatrix} g_{3j-1}g_{3j}-\frac{1}{4} & \frac{1}{2}g_{3j} & \frac{1}{4} \\ \frac{1}{2}g_{3j} & g_{3j-2}g_{3j} & \frac{1}{2}g_{3j-2} \\ \frac{1}{4} & \frac{1}{2}g_{3j-2} & g_{3j-2}g_{3j-1}-\frac{1}{4} \end{pmatrix}, \tag{6.3.31b}$$

where $d_j=1/[g_{3j-2}g_{3j-1}g_{3j}-\frac{1}{4}(g_{3j-2}+g_{3j})]$, for $j=1,2,\ldots,n$

$$R_j^{-1} = d_j \begin{pmatrix} g_{4j-2}(g_{4j-1}g_{4j}-\frac{1}{4})-\frac{1}{4}g_{4j} & \frac{1}{2}(g_{4j-1}g_{4j}-\frac{1}{4}) & \frac{1}{4}g_{4j} & \frac{1}{8} \\ \frac{1}{2}(g_{4j-1}g_{4j}-\frac{1}{4}) & g_{4j-1}(g_{4j-1}g_{4j}-\frac{1}{4}) & \frac{1}{2}g_{4j-3}g_{4j} & \frac{1}{4}g_{4j-3} \\ \frac{1}{4}g_{4j} & \frac{1}{2}g_{4j-3}g_{4j} & g_{4j}(g_{4j-3}g_{4j-2}-\frac{1}{4}) & \frac{1}{2}(g_{4j-3}g_{4j-2}-\frac{1}{4}) \\ \frac{1}{8} & \frac{1}{4}g_{4j-3} & \frac{1}{2}(g_{4j-3}g_{4j-2}-\frac{1}{4}) & g_{4j-3}(g_{4j-2}g_{4j-1}-\frac{1}{4})-\frac{1}{4}g_{4j-1} \end{pmatrix} \tag{6.3.31c}$$

where $d_j=1/[(g_{4j-1}g_{4j}-\frac{1}{4})(g_{4j-3}g_{4j-2}-\frac{1}{4})-\frac{1}{4}g_{4j-3}g_{4j}]$, $j=1,2,\ldots,n$.

Hence, $B_1^{-1}B_2$ and $B_1^{-1}\underline{b}$ can be determined explicitly to give the iterative scheme,

$$\underline{u}^{(k+1)} = B_1^{-1}B_2\underline{u}^{(k)} + B_1^{-1}\underline{b}\ , \tag{6.3.32}$$

which can be solved by one of the iterative methods, like the Jacobi, J.O.R., Gauss-Seidel or N.L.O.R.methods.

Problem 3

$$U_1' = U_2\ , \tag{6.3.33a}$$

$$U_2' = U_1^{10} - \sin x(1+\sin^9 x)\ , \tag{6.3.33b}$$

subject to the boundary condition (6.3.23). The exact solution is the same as Problem 2 which is given in (6.3.24).

By following the same steps as the previous problem, we obtain the nonlinear difference equation which can be written in the matrix iterative form,

$$\begin{bmatrix} g_1 & -1/2 & & & & \\ -1/2 & g_2 & -1/2 & & O & \\ & \ddots & \ddots & \ddots & & \\ & & \ddots & \ddots & \ddots & \\ & O & & -1/2 & g_{m-1} & -1/2 \\ & & & & -1/2 & g_m \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ \vdots \\ u_{m-1} \\ u_m \end{bmatrix}^{(k+1)}$$

$$= \frac{h^2}{2} \begin{bmatrix} \sin x_1(1+\sin^9 x_1) \\ \sin x_2(1+\sin^9 x_2) \\ \vdots \\ \vdots \\ \sin x_{m-1}(1+\sin^9 x_{m-1}) \\ \sin x_m(1+\sin^9 x_m) \end{bmatrix}, \qquad (6.3.34)$$

where $g_i = g(u_i^{(k)}) = 1 + \frac{h^2}{2}(u_i^{(k)})^9$, $i=1,2,\ldots,m$.

The same discussion for Problem 2 can be applied to this problem.

6.3.2 EXPERIMENTAL RESULTS

The experimental results presented here are for solving the three non-linear problems using Picard's method and the 2,3 and 4 point explicit groups Jacobi, Gauss-Seidel and non-linear successive overrelaxation iterative methods. All the results were obtained using the Prime Computer at Loughborough University.

The convergence test used in the numerical experiments was the average test, see (6.3.9), where $\varepsilon=10^{-5}$. The experimental optimum value of ω for the group non-linear overrelaxation are determined to within ± 0.005. In each method we assume that there are m internal mesh points where m=24.

The results obtained are recorded in Tables (6.3.1), (6.3.2) and (6.3.3).

METHOD	Jacobi	G.S.	N.L.O.R.	
	No.of iterations (k)	k	ω_b	k
Picard	5	5	1	5
2-Point Group	340	179	1.68 - 1.681	29
3-Point Group	241	127	1.623- 1.625	23
4-Point Group	190	100	1.584	19

TABLE 6.3.1 Problem 1

METHOD	Jacobi	G.S.	N.L.O.R.	
	k	k	ω_b	k
Picard	No conv.	64(*)	0.69- 0.73	10
2-Point Group	140	54	1.386 1.406	21
3-Point Group	101	38	1.305- 1.321	17
4-Point Group	80	28	1.255 1.275	15

TABLE 6.3.2 Problem 2

(*) *using double precision.*

METHOD	Jacobi	G.S.	N.L.O.R.	
	k	k	ω_b	k
Picard	No.conv.	No.conv.	0.4-0.41	21
2-Point Group	128	51	1.43 - 1.444	20
3-Point Group	96	36	1.352-1.376	16
4-Point Group	74	29	1.27-1.34	14

TABLE 6.3.3 Problem 3

6.3.3 THE COMPUTATIONAL COMPLEXITY

We now consider the relative efficiencies of the various iterative methods which we have introduced.

To solve the non-linear system using Picard's algorithm then at each iteration, the number of operations required for the m internal mesh points is

$$5m \text{ multiplications} + 3m \text{ additions} + \text{R.H.S. unit} , \qquad (6.3.35a)$$

In addition, since the N.L.O.R. iterative scheme requires an extra 1 multiplication and 2 additions for each point per iteration. Hence, the number of operations required is

$$6m \text{ multiplications} + 5m \text{ additions} + \text{R.H.S. unit} , \qquad (6.3.35b)$$

per point.

For the 2,3 and 4-point group iterative methods for solving the non-linear problems, the number of operations required can be determined for each problem as follows.

Problem 1

The 2-point group

It can be seen, from the system (6.3.14), that the $(k+1)^{th}$ iterate of the G.S. iterative scheme is given by,

$$\left.\begin{aligned} u_1^{(k+1)} &= g_1 g_2 u_3^{(k)} d_1 + \text{R.H.S.} \\ u_2^{(k+1)} &= g_2 u_3^{(k)} d_1 + \text{R.H.S.} \end{aligned}\right\} \qquad (6.3.36a)$$

then for i=3,(2),m-3

$$\left.\begin{aligned} u_i^{(k+1)} &= (f_i u_{i-1}^{(k+1)} + g_i g_{i+1} u_{i+2}^{(k)}) d_{\frac{i+1}{2}} + \text{R.H.S.} \\ u_{i+1}^{(k+1)} &= (f_i f_{i+1} u_{i-1}^{(k+1)} + g_{i+1} u_{i+2}^{(k)}) d_{\frac{i+1}{2}} + \text{R.H.S.} \end{aligned}\right\} \qquad (6.3.36b)$$

and finally,

$$\left.\begin{aligned} u_{m-1}^{(k+1)} &= f_{m-1} u_{m-2}^{(k+1)} d_{\frac{m}{2}} + \text{R.H.S.} \\ u_m^{(k+1)} &= f_{m-1} f_m u_{m-2}^{(k+1)} d_{\frac{m}{2}} + \text{R.H.S.} \end{aligned}\right\} \qquad (6.3.36c)$$

where g_i and f_i $(i=1,2,\ldots,m)$ are defined in (6.3.8b) and d_j $(j=1,2,\ldots,\frac{m}{2})$ is defined in (6.3.13b).

Hence the number of operations required for m mesh points per iteration for the 2 point group G.S. iterative method is

$$(7m-6) \text{ multiplications} + (\tfrac{7}{2}m-4) \text{ additions} + \text{R.H.S. unit} \qquad (6.3.37)$$

To proceed to the N.L.O.R. iterative scheme an extra 1 multiplication and 2 additions for each point per iteration is required. Hence, to solve this problem by applying the 2-point group N.L.O.R. iterative method, the number of operations required is,

$$(8m-6) \text{ multiplications} + (6m-4) \text{ additions} + \text{R.H.S. unit} \qquad (6.3.38)$$

The 3-point group

In a similar manner to the 2-point group method, the (k+1)th iterate of the G.S. iterative scheme can be obtained for the 3-point group, in which the system to be solved has the matrix form (6.3.12), where $B_1^{-1}B_2$ and $B_1^{-1}\underline{b}$ are given in (6.3.17a) and (6.3.17b) respectively. Hence, we can show that the 3-point group G.S. iterative method requires,

$$\frac{28}{3}\text{ m multiplications} + \frac{22}{3}\text{ m additions + R.H.S. unit} \qquad (6.3.39)$$

for m mesh points per iteration and hence, for the 3-point group N.L.O.R. iterative scheme, the number of operations required is,

$$\frac{31}{3}\text{ m multiplications} + \frac{28}{3}\text{ m additions + R.H.S. unit} \qquad (6.3.40)$$

The 4-point group

For this grouping strategy, it can be seen that the number of operations needed per iteration for the G.S. iterative method for the m mesh points is

$$12m\text{ multiplications} + 10m\text{ additions + R.H.S. unit .} \qquad (6.3.41)$$

Hence, the 4-point group N.L.O.R. iterative method requires

$$13m\text{ multiplications} + 12m\text{ additions + R.H.S. unit} \qquad (6.3.42)$$

per iteration.

Problem 2

The 2-point group

From the analysis given earlier for solving this problem, it can be seen that the $(k+1)^{th}$ iterate of the 2-point group G.S. iterative method is,

$$\left.\begin{aligned} u_1^{(k+1)} &= 0.25d_1u_3^{(k)} + \text{R.H.S.} \\ u_2^{(k+1)} &= 0.5d_1g_1u_3^{(k)} + \text{R.H.S.} \end{aligned}\right\} \qquad (6.3.43a)$$

then, for i=3,(2),m-3

$$\left.\begin{aligned} u_i^{(k+1)} &= 0.5d_{\frac{i+1}{2}}(g_{i+1}u_{i-1}^{(k+1)}+0.5u_{i+2}^{(k)}) + \text{R.H.S.} \\ u_{i+1}^{(k+1)} &= 0.5d_{\frac{i+1}{2}}(0.5u_{i-1}^{(k+1)}+g_iu_{i+2}^{(k)}) + \text{R.H.S.} \end{aligned}\right\} \qquad (6.3.43b)$$

and
$$\left.\begin{aligned} u_{m-1}^{(k+1)} &= 0.5d_{\frac{m}{2}}\, g_m u_{m-2}^{(k+1)} + \text{R.H.S.} \\ u_m^{(k+1)} &= 0.25d_{\frac{m}{2}}\, u_{m-2}^{(k+1)} + \text{R.H.S.} \end{aligned}\right\} \qquad (6.3.43c)$$

where,
$$g_i = g(u_i^{(k)}) = 1 + \frac{h^2}{2}(u_i^{(k)})^2 \ , \ i=1,2,\ldots,m \qquad (6.3.44)$$

Hence, for m mesh points, this method requires

$$(7m-6) \text{ multiplications} + (\frac{5}{2}m-4) \text{ additions} + \text{R.H.S. unit} \qquad (6.3.45)$$

per iteration.

In addition, since the N.L.O.R. iterative scheme requires an extra 1 multiplication and 2 additions for each point per iteration. Hence, the 2-point group N.L.O.R. iterative method requires

$$(8m-6) \text{ multiplications} + (\frac{9}{2}m-4) \text{ additions} + \text{R.H.S. unit} \qquad (6.3.46)$$

The 3-point group

From (6.3.30a), (6.3.30b) and (6.3.31b), we can obtain the 3-point iterative system (6.3.32), where a set of G.S. iterative equations can be written down in a similar manner to the 2-point group strategy. This technique requires,

$$(8m-10) \text{ multiplications} + (\frac{10}{3}m-8) \text{ additions} + \text{R.H.S. unit} \qquad (6.3.47)$$

per iteration.

Hence, for the 3-point group N.L.O.R. iterative method, the number of operations required is,

$$(9m-10) \text{ multiplications} + (\frac{16}{3}m-8) \text{ additions} + \text{R.H.S. unit} \qquad (6.3.48)$$

per iteration.

The 4-point group

In a similar way to the two previous groupings, it can be shown that the 4-point group G.S. iterative method requires

$$(10m-22) \text{ multiplications} + (3m-4) \text{ additions} + \text{R.H.S. unit} \qquad (6.3.49)$$

per iteration.

The N.L.O.R. scheme for this grouping requires

$$(11m-22) \text{ multiplications} + (5m-4) \text{ additions} + \text{R.H.S. unit} \qquad (6.3.50)$$

per iteration.

Problem 3

For this problem, it can be seen that the number of operations required for the 2,3 and 4-point group G.S. and N.L.O.R. iterative methods are the same as that given for Problem 2.

The total number of arithmetic operations required to solve the three problems by each of the Gauss-Seidel and N.L.O.R. methods can be obtained by combining the results shown in Tables (6.3.1), (6.3.2) and (6.3.3) with the number of operations per iteration required by each method given in (6.3.35a), (6.3.35b), (6.3.37), (6.3.38), (6.3.39), (6.3.40), (6.3.41) and (6.3.42) for Problem 1, and (6.3.45), (6.3.46), (6.3.47), (6.3.48), (6.3.49) and (6.3.50) for Problems 2 and 3. These are recorded in Tables (6.3.4), (6.3.5) and (6.3.6).

Method	G.S.		N.L.O.R.	
	Multiplications	Additions	M	A
Picard	25m	15m	30m	25m
2-Point Group	1253m-1074	626.5m-716	232m-174	174m-116
3-Point Group	1185.33m	931.33m	237.6m	214.6m
4-Point Group	1200m	1000m	247m	228m

TABLE 6.3.4 Problem 1

Method	G.S.		N.L.O.R.	
	M	A	M	A
Picard	320m(*)	192m(*)	60m	50m
2-Point Group	378m-324	135m-216	168m-126	94.5m-84
3-Point Group	304m-380	126.67m-304	153m-170	90.6m-136
4-Point Group	280m-616	75m-100	165m-330	75m-60

TABLE 6.3.5 Problem 2

(*) *using double precision*

Method	G.S.		N.L.O.R.	
	M	A	M	A
Picard	-	-	126m	105m
2-Point Group	357m-306	127.5m-204	160m-120	90m-80
3-Point Group	288m-360	120m-288	144m-160	85.3m-128
4-Point Group	290m-638	87m-116	154m-308	70m-56

TABLE 6.3.6 Problem 3

6.3.4 CONCLUSIONS

The problems discussed and solved using the point Picard and the new 2,3 and 4 point groups G.S. and N.L.O.R. methods vary in non-linearity from mildly non-linear (Problem 1) to a highly non-linear (Problem 3). For Problem 1 and from Table (6.3.4), it can be seen that the Picard G.S. and N.L.O.R. methods require about $\frac{1}{30}$ and $\frac{1}{8}$ of the work required by the 2-point group G.S. and N.L.O.R. methods respectively, where the 2-point group is more efficient than the other two groups.

The results shown in Table (6.3.5) indicate that the Picard G.S. method requires more arithmetic operations than any of the groups considered to solve Problem 2, also a double precision program was used to achieve convergence for the Picard G.S. method. It is well known that calculations in double precision usually doubles the storage requirements and quadruples the running time as compared with single precision arithmetic. The same table shows that the Picard N.L.O.R. method requires less arithmetic operations than the 2,3 and 4-point group N.L.O.R. methods. Finally, for the highly non-linear Problem 3 convergence was hardly achieved by using the Picard G.S. method to solve this problem, but by applying the N.L.O.R. method, the 3-point group approach required about 1.5 times the arithmetic operations needed by Picard's approach.

Hence, from the above discussion we can conclude that the point methods involve less work than the explicit group methods especially for the mildly non-linear problems. The results also indicate that the more non-linear the problem becomes then the 3-point group method could become competitive both in computational work and convergence.

6.4 ALTERNATING GROUP EXPLICIT (A.G.E.) METHOD

6.4.1 DEVELOPMENT OF THE METHOD

Consider equation (6.1.1), where the finite difference replacement of this equation has been obtained and is given by equation (6.1.7), which in matrix notation has the form,

$$A\underline{u} = \underline{b} \; . \tag{6.4.1a}$$

where the (m×m) matrix A and the two vectors $\underline{u}$ (the discrete approximation to the solution U(x)) and $\underline{b}$ are given by:

$$A = \begin{bmatrix} 2+q_1h^2 & -1 & & & & \\ -1 & 2+q_2h^2 & -1 & & O & \\ & \ddots & \ddots & \ddots & & \\ & & \ddots & \ddots & \ddots & \\ & O & & -1 & 2+q_{m-1}h^2 & -1 \\ & & & & -1 & 2+q_mh^2 \end{bmatrix} \tag{6.4.1b}$$

$$\underline{u} = (u_1,u_2,\ldots,u_m)^T \; , \tag{6.4.1c}$$

and

$$\underline{b} = h^2(f_1+\frac{\alpha}{h^2},f_2,\ldots,f_{m-1},f_m+\frac{\beta}{h^2})^T \; . \tag{6.4.1d}$$

We now consider a class of methods for solving the system (6.4.1), which is based on the "splitting" of the matrix A into the sum of three matrices,

$$A = G_1 + G_2 + \Sigma \; , \tag{6.4.2}$$

where Σ is a non-negative diagonal matrix and where G_1,G_2 and Σ satisfy the conditions:

(i) $G_1+\theta\Sigma+rI$ and $G_2+\theta\Sigma+rI$ are non-singular for any $\theta\geq 0$, $r>0$.

(ii) for any vectors $\underline{v}_1$ and $\underline{v}_2$ and for any constants $\theta\geq 0$ and $r>0$ it is "convenient" to solve the systems explicitly, i.e.,

$$\underline{y}_1 = G_1^{-1}\underline{v}_1 \quad \text{and} \quad \underline{y}_2 = G_2^{-1}\underline{v}_2$$

for $\underline{y}_1$ and $\underline{y}_2$ respectively.

We shall be concerned here with the situation where G_1 and G_2 are either small 2×2 block systems or can be made so by a suitable permutation of their rows and corresponding columns. This procedure is "convenient" in the sense that the work required is much less than would be required to solve the original system (6.4.1a) directly.

From the above discussion, we have G_1, G_2 and Σ given by,

$$G_1 = \left(\begin{array}{cc|cc|c|cc} 1 & -1 & & & & & \\ -1 & 1 & & & & 0 & \\ \hline & & 1 & -1 & & & \\ & & -1 & 1 & & & \\ \hline & & & & \ddots & & \\ \hline & 0 & & & & 1 & -1 \\ & & & & & -1 & 1 \end{array}\right), \quad G_2 = \left(\begin{array}{c|cc|cc|c|c} 1 & & & & & & \\ \hline & 1 & -1 & & & & 0 \\ & -1 & 1 & & & & \\ \hline & & & 1 & -1 & & \\ & & & -1 & 1 & & \\ \hline & & & & & \ddots & \\ \hline & 0 & & & & & 1 \end{array}\right)$$

and
$$\Sigma = h^2 \begin{pmatrix} q_1 & & & & \\ & q_2 & & 0 & \\ & & \ddots & & \\ & 0 & & q_{m-1} & \\ & & & & q_m \end{pmatrix} \qquad (6.4.3)$$

By using (6.4.2) we can write the matrix equation (6.4.1a) in the form

$$(G_1+G_2+\Sigma)\underline{u} = \underline{b} \ . \qquad (6.4.4)$$

Let us consider two equivalent forms,

$$(G_1+\theta\Sigma+rI)\underline{u} = \underline{b}-(G_2+(1-\theta)\Sigma-rI)\underline{u} \ , \qquad (6.4.5a)$$

and
$$(G_2+\theta\Sigma+r'I)\underline{u} = \underline{b}-(G_1+(1-\theta)\Sigma-r'I)\underline{u} \ . \qquad (6.4.5b)$$

In the Peaceman-Rachford method (1955), discussed in Chapter 3, one selects positive *iteration parameters* r and r' and determines $\underline{u}^{(k+\frac{1}{2})}$ and $\underline{u}^{(k+1)}$ respectively, by,

$$(G_1+\Sigma\theta+rI)\underline{u}^{(k+\frac{1}{2})} = \underline{b}-(G_2+(1-\theta)\Sigma-rI)\underline{u}^{(k)} \ , \tag{6.4.6a}$$

and

$$(G_2+\Sigma\hat{\theta}+r'I)\underline{u}^{(k+1)} = \underline{b}-(G_1+(1-\hat{\theta})\Sigma-r'I)\underline{u}^{(k+\frac{1}{2})} \ , \tag{6.4.6b}$$

where $\underline{u}^{(0)}$ is an arbitrary initial vector approximation of the unique solution of (6.4.1a).

For simplicity, we shall consider here the special case where,

$$\theta = \hat{\theta} = 1/2 \ , \quad r=r' \ , \tag{6.4.7}$$

and we let, $$\bar{G}_1 = G_1 + \tfrac{1}{2}\Sigma \ , \quad \bar{G}_2 = G_2 + \tfrac{1}{2}\Sigma \ . \tag{6.4.8}$$

Evidently, $\bar{G}_1$ and $\bar{G}_2$ satisfy the following conditions:

(i) $\bar{G}_1+rI$ and $\bar{G}_2+rI$ are non-singular for any $r>0$,

(ii) for any vectors $\underline{v}_1$ and $\underline{v}_2$ and for any $r>0$ it is convenient to solve the systems explicitly,

$$(\bar{G}_1+rI)\underline{y}_1 = \underline{v}_1, \quad (\bar{G}_2+rI)\underline{y}_2 = \underline{v}_2 \ . \tag{6.4.9}$$

Thus (6.4.4) becomes,

$$(\bar{G}_1+\bar{G}_2)\underline{u} = \underline{b} \ , \tag{6.4.10}$$

and (6.4.5a)-(6.4.5b) become, respectively,

$$(\bar{G}_1+rI)\underline{u}^{(k+\frac{1}{2})} = \underline{b}-(\bar{G}_2-rI)\underline{u}^{(k)} \ , \tag{6.4.11a}$$

$$(\bar{G}_2+rI)\underline{u}^{(k+1)} = \underline{b}-(\bar{G}_1-rI)\underline{u}^{(k+\frac{1}{2})} \ . \tag{6.4.11b}$$

If we combine the above two equations into the form,

$$\underline{u}^{(k+1)} = T_r\underline{u}^{(k)} + \underline{b}_r \ , \tag{6.4.12}$$

where,

$$\underline{b}_r = (\bar{G}_2+rI)^{-1}\{I-(\bar{G}_1-rI)(\bar{G}_1+rI)^{-1}\}\underline{b} \ . \tag{6.4.13a}$$

and $$T_r = (\bar{G}_2+rI)^{-1}(\bar{G}_1-rI)(\bar{G}_1+rI)^{-1}(G_2-rI) \ . \tag{6.4.13b}$$

The matrix T_r is called the *A.G.E. matrix*.

Now, to analyze the convergence of the A.G.E. method, we assume that $\underline{U}$ is the true solution of (6.4.1a) then,

$$(\bar{G}_1+\bar{G}_2)\underline{U} = \underline{b} \ , \tag{6.4.14}$$

and
$$(\bar{G}_1+rI)\underline{U} = \underline{b}-(\bar{G}_2-rI)\underline{U} \ . \tag{6.4.15}$$

If $\underline{e}^{(k)} = \underline{u}^{(k)}-\underline{U}$ is the error vector associated with the vector iterate $\underline{u}^{(k)}$. Therefore from (6.4.11a) and (6.4.15) we have,

$$(\bar{G}_1+rI)\underline{e}^{(k+\frac{1}{2})} = -(\bar{G}_2-rI)\underline{e}^{(k)} \ . \tag{6.4.16a}$$

Similarly,
$$(\bar{G}_2+rI)\underline{e}^{(k+1)} = -(\bar{G}_1-rI)\underline{e}^{(k+\frac{1}{2})} \tag{6.4.16b}$$

and hence,
$$\underline{e}^{(k+1)} = T_r\underline{e}^{(k)} \ , \tag{6.4.17}$$

where T_r is given in (6.4.13b).

To indicate the convergence properties of T_r, we prove the following theorem.

<u>Theorem 6.4.1</u>

If $\bar{G}_1$ and $\bar{G}_2$ are real positive definite matrices and if $r>0$ then, $\rho(T_r)<1$.

<u>Proof</u>: If we define the matrix $\tilde{T}_r$ as

$$\begin{aligned}\tilde{T}_r &= (\bar{G}_2+rI)T_r(\bar{G}_2+rI)^{-1}\\ &= (\bar{G}_1-rI)(\bar{G}_1+rI)^{-1}(\bar{G}_2-rI)(\bar{G}_2+rI)^{-1} \ ,\end{aligned} \tag{6.4.18}$$

then it is evident that T_r is similar to $\tilde{T}_r$, and hence from the properties of the matrix norms we have,

$$||\tilde{T}_r|| \leq ||(\bar{G}_1-rI)(\bar{G}_1+rI)||.||(\bar{G}_2-rI)(\bar{G}_2+rI)|| . \tag{6.4.19}$$

But since $\bar{G}_1$ and $\bar{G}_2$ are symmetric and since $(\bar{G}_1-rI)$ commutes with $(\bar{G}_1+rI)^{-1}$ we have,

$$\begin{aligned}||(\bar{G}_1-rI)(\bar{G}_1+rI)^{-1}|| &= \rho((\bar{G}_1-rI)(\bar{G}_1+rI)^{-1})\\ &= \max_{\lambda}\left|\frac{\lambda-r}{\lambda+r}\right| \ ,\end{aligned} \tag{6.4.20}$$

where λ ranges over all eigenvalues of $\bar{G}_1$. But since $\bar{G}_1$ is positive definite, its eigenvalues are positive. Therefore,

$$||(\bar{G}_1-rI)(\bar{G}_1+rI)^{-1}|| < 1 \ . \tag{6.4.21}$$

The same argument applied to the corresponding matrix product with $\bar{G}_2$ shows that $||(\bar{G}_2-rI)(\bar{G}_2+rI)^{-1}||<1$, and we therefore conclude that,

$$\rho(T_r) = \rho(\tilde{T}_r) \leq ||\tilde{T}_r|| < 1 \ , \tag{6.4.22}$$

hence, the convergence follows.

Let us now assume that $\bar{G}_1$ and $\bar{G}_2$ are real positive definite matrices and that the eigenvalues λ of $\bar{G}_1$ and μ of $\bar{G}_2$ lie in the ranges,

$$0 < a \leq \lambda \leq b \quad , \quad 0 < a \leq \mu \leq b \ . \tag{6.4.23}$$

Evidently, if $r>0$ we have,

$$\rho(T_r) \leq \rho((\bar{G}_1-rI)(\bar{G}_1+rI)^{-1})\rho((\bar{G}_2-rI)(\bar{G}_2+rI)^{-1})$$

$$= (\max_{a\leq\lambda\leq b}\left|\frac{\lambda-r}{\lambda+r}\right|)(\max_{a\leq\mu\leq b}\left|\frac{\mu-r}{\mu+r}\right|) = [\max_{a\leq\gamma\leq b}\left|\frac{\gamma-r}{\gamma+r}\right|]^2 = \phi(a,b;r) \tag{6.4.24}$$

Since $(\gamma-r)/(\gamma+r)$ is an increasing function of γ we have

$$\max_{a\leq\gamma\leq b}\left|\frac{\gamma-r}{\gamma+r}\right| = \max(\left|\frac{a-r}{a+r}\right|,\left|\frac{b-r}{b+r}\right|) \ . \tag{6.4.25}$$

When $r=\sqrt{ab}$, then,

$$\left|\frac{a-r}{a+r}\right| = \left|\frac{b-r}{b+r}\right| = \frac{\sqrt{b}-\sqrt{a}}{\sqrt{b}+\sqrt{a}} \ . \tag{6.4.26}$$

Moreover, if $0<r<\sqrt{ab}$ we have,

$$\left|\frac{b-r}{b+r}\right| - \frac{\sqrt{b}-\sqrt{a}}{\sqrt{b}+\sqrt{a}} = \frac{2\sqrt{b}(\sqrt{ab}-r)}{(b+r)(\sqrt{b}+\sqrt{a})} > 0 \ , \tag{6.4.27}$$

and if $\sqrt{ab}<r$, then,

$$\left|\frac{a-r}{a+r}\right| - \frac{\sqrt{b}-\sqrt{a}}{\sqrt{b}+\sqrt{a}} = \frac{2\sqrt{b}(r-\sqrt{ab})}{(r+a)(\sqrt{b}+\sqrt{a})} > 0 \ . \tag{6.4.28}$$

Therefore $\phi(a,b;r)$ is minimized when $r=\sqrt{ab}$ and

$$\rho(T_{\sqrt{ab}}) \leq \phi(a,b;\sqrt{ab}) = (\frac{\sqrt{b}-\sqrt{a}}{\sqrt{b}+\sqrt{a}})^2 \ . \tag{6.4.29}$$

Thus, $r=\sqrt{ab}$ is optimum in the sense that the bound $\phi(a,b;r)$ for $\rho(T_r)$ is minimized.

The convergence of the A.G.E. method is frequently very rapid if one allows r to vary cyclically from iteration to iteration. This rapid convergence can be proved to hold for an appropriate choice of the iteration parameters in the commutative case. Following Birkhoff, Varga and Young (1962), we say that the commutative case holds if the matrices G_1, G_2 and Σ of (6.4.4) satisfy the following conditions:

(i) $G_1G_2 = G_2G_1$, (6.4.30a)

(ii) $\Sigma=\sigma I$, where σ is a non-negative constant, (6.4.30b)

(iii) G_1 and G_2 are similar to non-negative diagonal matrices. (6.4.30c)

If these assumptions hold then the matrices $\bar{G}_1=G_1+\frac{1}{2}\Sigma$, $\bar{G}_2=G_2+\frac{1}{2}\Sigma$ satisfy the conditions,

(i') $\bar{G}_1\bar{G}_2 = \bar{G}_2\bar{G}_1$ (6.4.31a)

(ii') $\bar{G}_1$ and $\bar{G}_2$ are similar to non-negative diagonal matrices. (6.4.31b)

The importance of the above conditions depends on the following theorem of Frobenius, which we state without proof.

Theorem 6.4.2

If $\bar{G}_1$ and $\bar{G}_2$ are similar to diagonal matrices and if $\bar{G}_1\bar{G}_2=\bar{G}_2\bar{G}_1$ then there exists a non-singular matrix W such that,

$$W^{-1}\bar{G}_1W = D_1 \quad , \qquad W^{-1}\bar{G}_2W = D_2 \; , \tag{6.4.32}$$

where, D_1 and D_2 are diagonal matrices.

It follows from (6.4.31a) and (6.4.31b) that there exists a set of linearly independent vectors $\underline{v}_1,\underline{v}_2,\ldots,\underline{v}_k$, which corresponds to the columns of W in Theorem (6.4.2), such that each $\underline{v}_i$ is an eigenvector both of $\bar{G}_1$ and of $\bar{G}_2$.

Now, for any column $\underline{v}$ of W we have,

$$\bar{G}_1\underline{v} = \lambda\underline{v}\ ,\quad \bar{G}_2\underline{v} = \mu\underline{v}\ , \tag{6.4.33}$$

for some eigenvalues λ and μ of $\bar{G}_1$ and $\bar{G}_2$ respectively. Evidently,

$$T_r\underline{v} = (\bar{G}_2+rI)^{-1}(\bar{G}_1-rI)(\bar{G}_1+rI)^{-1}(\bar{G}_2-rI)\underline{v}\ . \tag{6.4.34}$$

Since $(\bar{G}_1+rI)^{-1}\underline{v} = (\lambda+r)^{-1}\underline{v}$, $(\bar{G}_2+rI)^{-1}\underline{v}=(\mu+r)^{-1}\underline{v}$ it follows that,

$$T_r\underline{v} = \frac{(\lambda-r)(\mu-r)}{(\lambda+r)(\mu+r)}\,\underline{v}\ . \tag{6.4.35}$$

Thus $\underline{v}$ is an eigenvector of T_r for any r.

6.4.2 EXPERIMENTAL RESULTS

We now describe some numerical experiments in order to compare the new group strategy with the N.L.O.R. method. All the results were obtained using the Prime Computer at Loughborough University.

Problem 1

We consider again Problem 1 given in subsection (6.3.1) and by following the same steps we obtain the simplified form,

$$-(1-\frac{h}{16}u_i)u_{i-1}+2u_i-(1+\frac{h}{16}u_i)u_{i+1} = -4h^2(1+\frac{1}{16}x_i^3),\ i=1,2,\ldots,m \quad (6.4.36)$$

which in matrix form is

$$A\underline{u} = \underline{b}\ , \quad (6.4.37)$$

where,

$$A = \begin{bmatrix} 2 & -g_1 & & & & \\ -f_2 & 2 & -g_2 & & & \\ & -f_3 & 2 & -g_3 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -f_{m-1} & 2 & -g_{m-1} \\ & & & & -f_m & 2 \end{bmatrix}, \quad (6.4.38a)$$

$$\underline{b} = (c_1+17f_1, c_2, \ldots, c_{m-1}, c_m + \frac{43}{3}g_m)^T\ , \quad (6.4.38b)$$

and $$\underline{u} = (u_1,u_2,\ldots,u_{m-1},u_m)^T\ , \quad (6.4.38c)$$

$$g_i = g(u_i) = 1 + \frac{16}{h}u_i\ ,\quad f_i = f(u_i) = 1 - \frac{h}{16}u_i\ ,$$

and $$c_i = -4h^2(1+\frac{1}{16}x_i^3),\quad i=1,2,\ldots,m. \quad (6.4.38d)$$

We now split the matrix A into the form (6.4.2), hence from (6.4.8) $\overline{G_1}$ and $\overline{G_2}$ have the form,

$$\bar{G}_1 = \left[\begin{array}{cc|cc|c|cc} 1 & -g_1 & & & & & \\ -f_2 & 1 & & & & 0 & \\ \hline & & 1 & -g_3 & & & \\ & & -f_4 & 1 & & & \\ \hline & & & & \ddots & & \\ \hline & 0 & & & & 1 & -g_{m-1} \\ & & & & & -f_m & 1 \end{array}\right] = \begin{bmatrix} G_1^{(1)} & & & & 0 \\ & G_1^{(2)} & & & \\ & & \ddots & & \\ & & & G_1^{(\frac{m}{2}-1)} & \\ 0 & & & & G_1^{(m/2)} \end{bmatrix} \qquad (6.4.39)$$

$$\bar{G}_2 = \left[\begin{array}{c|cc|c|cc|c} 1 & & & & & & \\ \hline & 1 & -g_2 & & 0 & & \\ & -f_3 & 1 & & & & \\ \hline & & & \ddots & & & \\ \hline & & & & 1 & -g_{m-2} & \\ & 0 & & & -f_{m-1} & 1 & \\ \hline & & & & & & 1 \end{array}\right] = \begin{bmatrix} 1 & & & & & 0 \\ & G_2^{(1)} & & & & \\ & & G_2^{(2)} & & & \\ & & & \ddots & & \\ & & & & G_2^{(\frac{m}{2}-1)} & \\ 0 & & & & & 1 \end{bmatrix} \qquad (6.4.40)$$

with,

$$G_1^{(i)} = \begin{bmatrix} 1 & -g_{2i-1} \\ -f_{2i} & 1 \end{bmatrix}, \quad i=1,2,\ldots,\frac{m}{2},$$

$$G_2^{(j)} = \begin{bmatrix} 1 & -g_{2j} \\ -f_{2j+1} & 1 \end{bmatrix}, \quad j=1,2,\ldots,\frac{m}{2}-1 .$$

By applying the A.G.E. method, $\underline{u}^{(k+1/2)}$ and $\underline{u}^{(k+1)}$ can be determined successively by

$$\begin{aligned} \underline{u}^{(k+1/2)} &= (\bar{G}_1+rI)^{-1}[\underline{b}-(\bar{G}_2-rI)\underline{u}^{(k)}] , \\ \underline{u}^{(k+1)} &= (\bar{G}_2+rI)^{-1}[\underline{b}-(\bar{G}_1-rI)\underline{u}^{(k+1/2)}] , \end{aligned} \qquad (6.4.41)$$

where r is the iteration parameter.

Evidently, $(\bar{G}_1+rI)$, $(\bar{G}_2+rI)$, $(\bar{G}_1-rI)$ and $(\bar{G}_2-rI)$ can be determined by inspection with $(\bar{G}_1+rI)$ and $(\bar{G}_2+rI)$ easily invertible and the (2×2) submatrices of $(\bar{G}_1+rI)$, $(\bar{G}_2+rI)$, $(\bar{G}_1-rI)$ and $(\bar{G}_2-rI)$ verified to have the forms,

$$G_1^{(i)}+rJ = \hat{G}_1^{(i)} = \begin{pmatrix} \alpha & -g_{2i-1} \\ -f_{2i} & \alpha \end{pmatrix} , \quad G_1^{(i)}-rJ = \check{G}_1^{(i)} = \begin{pmatrix} \beta & -g_{2i-1} \\ -f_{2i} & \beta \end{pmatrix}$$

$$i=1,2,\dots,m/2$$

$$G_2^{(j)}+rJ = \hat{G}_2^{(j)} = \begin{pmatrix} \alpha & -g_{2j} \\ -f_{2j+1} & \alpha \end{pmatrix} , \quad G_2^{(j)}-rJ = \check{G}_2^{(j)} = \begin{pmatrix} \beta & -g_{2j} \\ -f_{2j+1} & \beta \end{pmatrix}$$

$$j=1,2,\dots,\frac{m}{2}-1 \qquad (6.4.42)$$

where $\alpha=1+r$, $\beta=1-r$ and J is the (2×2) identity matrix.

Hence,

$$(\bar{G}_1+rI)^{-1}=\hat{G}_1^{-1}=\begin{pmatrix} \hat{G}_1^{(1)} & & & & \\ & \hat{G}_1^{(2)} & & O & \\ & & \ddots & & \\ & & & \hat{G}_1^{(\frac{m}{2}-1)} & \\ & O & & & \hat{G}_1^{(m/2)} \end{pmatrix}^{-1} = \begin{pmatrix} (\hat{G}_1^{(1)})^{-1} & & & \\ & (\hat{G}_1^{(2)})^{-1} & & O \\ & & \ddots & \\ & O & & (\hat{G}_1^{(m/2)})^{-1} \end{pmatrix} \qquad (6.4.43a)$$

$$(\bar{G}_2+rI)^{-1}=\hat{G}_2^{-1}\begin{pmatrix} \alpha & & & & & \\ & \hat{G}_2^{(1)} & & & & \\ & & \hat{G}_2^{(2)} & & O & \\ & & & \ddots & & \\ & & & & \hat{G}_2^{(\frac{m}{2}-1)} & \\ & O & & & & \alpha \end{pmatrix}^{-1} = \begin{pmatrix} 1/\alpha & & & & & \\ & (\hat{G}_2^{(1)})^{-1} & & & & \\ & & (\hat{G}_2^{(2)})^{-1} & & O & \\ & & & \ddots & & \\ & O & & & (\hat{G}_2^{(\frac{m}{2}-1)})^{-1} & \\ & & & & & 1/\alpha \end{pmatrix} \qquad (6.4.43b)$$

with

$$(\hat{G}_1^{(i)})^{-1} = d_i \begin{pmatrix} \alpha & g_{2i-1} \\ f_{2i} & \alpha \end{pmatrix} , \text{ where } d_i = \frac{1}{\alpha^2-f_{2i}g_{2i-1}} ,$$

$$i=1,2,\dots,m/2 , \qquad (6.4.44a)$$

and
$$(\hat{G}_2^{(j)})^{-1} = d'_j \begin{pmatrix} \alpha & g_{2j} \\ f_{2j+1} & \alpha \end{pmatrix} \text{, where } d'_j = \frac{1}{\alpha^2 - f_{2j+1}g_{2j}} \text{,}$$
$$j=1,2,\ldots,\frac{m}{2}-1 \text{ .} \tag{6.4.44b}$$

From (6.4.41) $\underline{u}^{(k+\frac{1}{2})}$ and $\underline{u}^{(k+1)}$ are given by,

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{m-1} \\ u_m \end{pmatrix}^{(k+\frac{1}{2})} = \left(\begin{array}{cc|cc|c|cc} \alpha d_1 & g_1 d_1 & & & & & \\ f_2 d_1 & \alpha d_1 & & & & \multicolumn{2}{c}{0} \\ \hline & & \alpha d_2 & g_3 d_2 & & & \\ & & f_4 d_2 & \alpha d_2 & & & \\ \hline & & & & \ddots & & \\ \hline & 0 & & & & \alpha d_{\frac{m}{2}} & g_{m-1} d_{\frac{m}{2}} \\ & & & & & f_m d_{\frac{m}{2}} & \alpha d_{\frac{m}{2}} \end{array}\right) \begin{pmatrix} c_1+17f_1-\beta u_1 \\ c_2-\beta u_2+g_2u_3 \\ c_3+f_3u_2-\beta u_3 \\ c_4-\beta u_4+g_4u_5 \\ \vdots \\ c_{m-1}+f_{m-1}u_{m-2}-\beta u_{m-1} \\ c_m+\frac{43}{3}g_m-\beta u_m \end{pmatrix}^{(k)}$$

(6.4.45a)

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{m-2} \\ u_{m-1} \\ u_m \end{pmatrix}^{(k+1)} = \left(\begin{array}{c|cc|c|cc|c} 1/\alpha & & & & & & \\ \hline & \alpha d'_1 & g_2 d'_1 & & & 0 & \\ & f_3 d'_1 & \alpha d'_1 & & & & \\ \hline & & & \ddots & & & \\ \hline & & & & \alpha d'_{\frac{m}{2}-1} & g_{m-2} d'_{\frac{m}{2}-1} & \\ & & 0 & & f_{m-1} d'_{\frac{m}{2}-1} & \alpha d'_{\frac{m}{2}-1} & \\ \hline & & & & & & 1/\alpha \end{array}\right) \begin{pmatrix} c_1+17f_1-\beta u_1+g_1u_2 \\ c_2+f_2u_1-\beta u_2 \\ c_3-\beta u_3+g_3u_4 \\ \vdots \\ c_{m-2}+f_{m-1}u_{m-3}-\beta u_{m-2} \\ c_{m-1}-\beta u_{m-1}+g_{m-1}u_m \\ c_m+\frac{43}{3}g_m+f_mu_{m-1}-\beta u_m \end{pmatrix}^{(k+\frac{1}{2})}$$

(6.4.45b)

Problem 2

We consider Problem 2 given in subsection (6.3.1) and by following the same process again, we obtain the non-linear equation,

$$-u_{i-1}+(2+h^2u_i^2)u_i-u_{i+1} = h^2\sin x_i(1+\sin^2 x_i) \text{ , } i=1,2,\ldots,m \tag{6.4.46}$$

which has the matrix form (6.4.37), where,

$$A = \begin{bmatrix} 2g_1 & -1 & & & & \\ -1 & 2g_2 & -1 & & 0 & \\ & -1 & 2g_3 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & 0 & & -1 & 2g_{m-1} & -1 \\ & & & & -1 & 2g_m \end{bmatrix}, \quad \underline{b} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_{m-1} \\ c_m \end{bmatrix} \qquad (6.4.47)$$

and $\underline{u} = (u_1, u_2, \ldots, u_{m-1}, u_m)^T$,

with

$$g_i = 1 + \frac{h^2}{2} u_i^2 \qquad \text{and} \qquad c_i = h^2 \sin x_i (1 + \sin^2 x_i), \; i = 1, 2, \ldots, m. \qquad (6.4.48)$$

Hence $\overline{G_1}$ and $\overline{G_2}$ of (6.4.8) are given by,

$$\overline{G_1} = \left[\begin{array}{cc|cc|c|cc} g_1 & -1 & & & & & \\ -1 & g_2 & & & & 0 & \\ \hline & & g_3 & -1 & & & \\ & & -1 & g_4 & & & \\ \hline & & & & \ddots & & \\ \hline & 0 & & & & g_{m-1} & -1 \\ & & & & & -1 & g_m \end{array}\right] = \begin{bmatrix} G_1^{(1)} & & & \\ & G_1^{(2)} & & 0 \\ & & \ddots & \\ 0 & & & G_1^{(\frac{m}{2})} \end{bmatrix} \qquad (6.4.49)$$

$$\overline{G_2} = \left[\begin{array}{c|cc|c|cc|c} g_1 & & & & & & \\ \hline & g_2 & -1 & & & 0 & \\ & -1 & g_3 & & & & \\ \hline & & & \ddots & & & \\ \hline & & & & g_{m-2} & -1 & \\ & 0 & & & -1 & g_{m-1} & \\ \hline & & & & & & g_m \end{array}\right] = \begin{bmatrix} g_1 & & & & \\ & G_2^{(1)} & & & 0 \\ & & G_2^{(2)} & & \\ & & & \ddots & \\ & 0 & & G_2^{(\frac{m}{2}-1)} & \\ & & & & g_m \end{bmatrix}$$

(6.4.50)

where,

$$G_1^{(i)} = \begin{bmatrix} g_{2i-1} & -1 \\ -1 & g_{2i} \end{bmatrix}, \quad i=1,2,\ldots,\frac{m}{2},$$

$$G_2^{(j)} = \begin{bmatrix} g_{2j} & -1 \\ -1 & g_{2j+1} \end{bmatrix}, \quad j=1,2,\ldots,\frac{m}{2}-1.$$

The A.G.E. method can now be applied to obtain $\underline{u}^{(k+\frac{1}{2})}$ and hence $\underline{u}^{(k+1)}$ (see (6.4.41)). The (2×2) submatrices of $(\bar{G}_1+rI)$, $(\bar{G}_2+rI)$, $(\bar{G}_1-rI)$ and $(\bar{G}_2-rI)$ have the forms,

$$G_1^{(i)}+rJ = \hat{G}_1^{(i)} = \begin{bmatrix} g_{2i-1}+r & -1 \\ -1 & g_{2i}+r \end{bmatrix}, \quad G_1^{(i)}-rJ = \check{G}_1^{(i)} = \begin{bmatrix} g_{2i-1}-r & -1 \\ -1 & g_{2i}-r \end{bmatrix}$$

$$i=1,2,\ldots,\frac{m}{2},$$

$$G_2^{(j)}+rJ = \hat{G}_2^{(j)} = \begin{bmatrix} g_{2j}+r & -1 \\ -1 & g_{2j+1}+r \end{bmatrix}, \quad G_2^{(j)}-rJ = \check{G}_2^{(j)} = \begin{bmatrix} g_{2j}-r & -1 \\ -1 & g_{2j+1}-r \end{bmatrix}$$

$$j=1,2,\ldots,\frac{m}{2}-1, \tag{6.4.51}$$

and J is the (2×2) identity matrix.

Hence,

$$(\bar{G}_1+rI)^{-1} = \hat{G}_1^{-1} = \begin{bmatrix} (\hat{G}_1^{(1)})^{-1} & & & & \\ & (\hat{G}_1^{(2)})^{-1} & & 0 & \\ & & \ddots & & \\ & 0 & & (\hat{G}_1^{(\frac{m}{2}-1)})^{-1} & \\ & & & & (\hat{G}_1^{(m/2)})^{-1} \end{bmatrix} \tag{6.4.52}$$

$$(\bar{G}_2+rI)^{-1} = \hat{G}_2^{-1} = \begin{bmatrix} 1/g_1+r & & & & & \\ & (\hat{G}_2^{(1)})^{-1} & & & 0 & \\ & & (\hat{G}_2^{(2)})^{-1} & & & \\ & & & \ddots & & \\ & 0 & & & (\hat{G}_2^{(\frac{m}{2}-1)})^{-1} & \\ & & & & & 1/g_m+r \end{bmatrix} \tag{6.4.53}$$

with

$$(\hat{G}_1^{(i)})^{-1} = d_i \begin{pmatrix} g_{2i}+r & 1 \\ 1 & g_{2i-1}+r \end{pmatrix}, \text{ where } d_i = \frac{1}{(g_{2i-1}+r)(g_{2i}+r)-1}$$

$$i=1,2,\ldots,m/2$$

and

$$(\hat{G}_2^{(j)})^{-1} = d'_j \begin{pmatrix} g_{2j+1}+r & 1 \\ 1 & g_{2j}+r \end{pmatrix}, \text{ where } d'_j = \frac{1}{(g_{2j}+r)(g_{2j+1}+r)-1}$$

$$j=1,2,\ldots,\frac{m}{2}-1.$$

Hence, the explicit equations for $\underline{u}^{(k+\frac{1}{2})}$ and $\underline{u}^{(k+1)}$ have the form,

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{m-1} \\ u_m \end{pmatrix}^{(k+\frac{1}{2})} = \left(\begin{array}{cc|cc|c|cc} (g_2+r)d_1 & d_1 & & & & & \\ d_1 & (g_1+r)d_1 & & & & \multicolumn{2}{c}{0} \\ \hline & & (g_4+r)d_2 & d_2 & & & \\ & & d_2 & (g_3+r)d_2 & & & \\ \hline & & & & \ddots & & \\ \hline & 0 & & & & (g_m+r)d_{\frac{m}{2}} & d_{\frac{m}{2}} \\ & & & & & d_{\frac{m}{2}} & (g_{m-1}+r)d_{\frac{m}{2}} \end{array}\right) \begin{pmatrix} c_1-(g_1-r)u_1 \\ c_2-(g_2-r)u_2+u_3 \\ \vdots \\ c_{m-1}+u_{m-2}-(g_{m-1}-r)u_{m-1} \\ c_m-(g_m-r)u_m \end{pmatrix}^{(k)} \qquad (6.4.54a)$$

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{m-2} \\ u_{m-1} \\ u_m \end{pmatrix}^{(k+1)} = \left(\begin{array}{c|cc|c|cc|c} 1/g_1+r & & & & & & \\ \hline & (g_3+r)d'_1 & d'_1 & & & 0 & \\ & d'_1 & (g_2+r)d'_1 & & & & \\ \hline & & & \ddots & & & \\ \hline & & & & (g_{m-1}+r)d'_{\frac{m}{2}-1} & d'_{\frac{m}{2}-1} & \\ & & 0 & & d'_{\frac{m}{2}-1} & (g_{m-2}+r)d'_{\frac{m}{2}-1} & \\ \hline & & & & & & 1/g_m+r \end{array}\right)$$

$$\begin{bmatrix} c_1-(g_1-r)u_1+u_2 \\ c_2+u_1-(g_2-r)u_2 \\ c_3-(g_3-r)u_3+u_4 \\ \vdots \\ c_{m-2}+u_{m-3}-(g_{m-2}-r)u_{m-2} \\ c_{m-1}-(g_{m-1}-r)u_{m-1}+u_m \\ c_m+u_{m-1}-(g_m-r)u_m \end{bmatrix}^{(k+\frac{1}{2})} \qquad (6.4.54b)$$

Problem 3

Again we consider Problem 3 as given in subsection (6.3.1), where the following non-linear equations can be obtained,

$$-u_{i-1}+(2+h^2u_i^9)u_i-u_{i+1} = h^2 \sin x_i(1+\sin^9 x_i), \quad i=1,2,\ldots,m \qquad (6.4.55)$$

which can be solved by the A.G.E. method similar to Problem 2 with g_i and c_i replaced by,

$$\left.\begin{aligned} g_i &= 1+\frac{h^2}{2}u_i^9 \\ c_i &= h^2 \sin x_i(1+\sin^9 x_i) \end{aligned}\right\} i=1,2,\ldots,m \qquad (6.4.56)$$

and

Problem 4

We now consider the linear problem,

$$U_1' = U_2 \,, \qquad (6.4.57a)$$

$$U_2' = 400(U_1+\cos^2(\pi x)) + 2\pi^2\cos(2\pi x) \,, \qquad (6.4.57b)$$

subject to the boundary conditions,

$$U_1(0) = U_1(1) = 0 \,. \qquad (6.4.57c)$$

The exact solution for this problem is given by,

$$\left.\begin{aligned} U_1(x) &= \frac{e^{-20}}{1+e^{-20}}\cdot e^{20x} + \frac{1}{1+e^{-20}}\cdot e^{-20x} - \cos^2(\pi x) \\ U_2(x) &= \frac{20\,e^{-20}e^{20x}}{1+e^{-20}} - \frac{20}{1+e^{-20}}\,e^{-20x} + \pi\sin(2\pi x) \end{aligned}\right\} \qquad (6.4.58)$$

From (6.4.57a) and (6.4.57b), we have,

$$U_1'' = 400(U_1+\cos^2(\pi x)) + 2\pi^2 \cos(2\pi x) \quad . \tag{6.4.59}$$

By following the finite difference procedure of Section (6.1), equation (6.4.59) can be approximated to obtain the linear difference equation (assuming that $u=u_1$),

$$\frac{u_{i-1}-2u_i+u_{i+1}}{h^2} = 400[u_i+\cos^2(\pi x_i)] + 2\pi^2 \cos(2\pi x_i),\ i=1,2,\ldots,m\,.$$

This equation can be simplified to the form,

$$-u_{i-1}+(2+400h^2)u_i-u_{i+1} = -2h^2[200\cos^2(\pi x_i) + \pi^2\cos(2\pi x_i)] \qquad i=1,2,\ldots,m\,. \tag{6.4.60}$$

The boundary conditions are replaced by,

$$u_0 = 0 \quad \text{and} \quad u_{m+1} = 0 \tag{6.4.61}$$

where $h = \frac{1}{m+1}$.

The linear system (6.4.60) can be represented in matrix notation as

$$A\underline{u} = (\bar{G}_1+\bar{G}_2)\underline{u} = \underline{b}\ . \tag{6.4.62}$$

The vector $\underline{u}$ is defined in the usual way, and $\underline{b}$ is given by,

$$\underline{b} = (c_1,c_2,\ldots,c_{m-1},c_m)^T \quad , \tag{6.4.63a}$$

where

$$c_i = -2h^2[200\cos^2(\pi x_i) + \pi^2\cos(2\pi x_i)] \ ,\ i=1,2,\ldots,m \tag{6.4.63b}$$

$\bar{G}_1$ and $\bar{G}_2$ are given by,

$$\bar{G}_1 = \left[\begin{array}{cc|cc|c|cc} g & -1 & & & & & \\ -1 & g & & & & & \\ \hline & & g & -1 & & & \\ & & -1 & g & & & \\ \hline & & & & \ddots & & \\ \hline & & & & & g & -1 \\ & & & & & -1 & g \end{array}\right] \quad \text{and} \quad \bar{G}_2 = \left[\begin{array}{c|cc|c|cc|c} g & & & & & & \\ \hline & g & -1 & & & & \\ & -1 & g & & & & \\ \hline & & & \ddots & & & \\ \hline & & & & g & -1 & \\ & & & & -1 & g & \\ \hline & & & & & & g \end{array}\right] \tag{6.4.64}$$

with $g = 1+200h^2$.

Hence, by applying the A.G.E. method of subsection (6.4.1), we can determine $\underline{u}^{(k+\frac{1}{2})}$ and $\underline{u}^{(k+1)}$ successively from equation (6.4.41). It is obvious that the (2×2) submatrices of $(\bar{G}_1+rI)$ and $(\bar{G}_2+rI)$ are of the form

$$\hat{G} = \begin{pmatrix} \alpha & -1 \\ -1 & \alpha \end{pmatrix} ,$$

where $\alpha = g+r$, and the inverse of $\hat{G}$ is given by,

$$\hat{G}^{-1} = d\begin{pmatrix} \alpha & 1 \\ 1 & \alpha \end{pmatrix} , \text{ where } \quad d = \frac{1}{\alpha^2-1} \quad . \tag{6.4.65}$$

Hence the vector $\underline{u}^{(k+1)}$ can be determined from $\underline{u}^{(k)}$ in two steps, we first determine $\underline{u}^{(k+\frac{1}{2})}$ as follows,

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ \vdots \\ u_{m-1} \\ u_m \end{pmatrix}^{(k+\frac{1}{2})} = d \left(\begin{array}{cc|cc|c|cc} \alpha & 1 & & & & & \\ 1 & \alpha & & & & & \\ \hline & & \alpha & 1 & & & \\ & & 1 & \alpha & & & \\ \hline & & & & \ddots & & \\ \hline & & & & & \alpha & 1 \\ & & & & & 1 & \alpha \end{array}\right) \begin{pmatrix} c_1-\beta u_1 \\ c_2-\beta u_2+u_3 \\ c_3+u_2-\beta u_3 \\ c_4-\beta u_4+u_5 \\ \vdots \\ c_{m-1}+u_{m-2}-\beta u_{m-1} \\ c_m-\beta u_m \end{pmatrix}^{(k)} \tag{6.4.66a}$$

and by using the values of $u^{(k+\frac{1}{2})}$ we determine $\underline{u}^{(k+1)}$

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{m-2} \\ u_{m-1} \\ u_m \end{pmatrix}^{(k+1)} = \left(\begin{array}{c|cc|c|cc|c} 1/\alpha & & & & & & \\ \hline & \alpha d & d & & & & \\ & d & \alpha d & & & & \\ \hline & & & \ddots & & & \\ \hline & & & & \alpha d & d & \\ & & & & d & \alpha d & \\ \hline & & & & & & 1/\alpha \end{array}\right) \begin{pmatrix} c_1-\beta u_1+u_2 \\ c_2+u_1-\beta u_2 \\ c_3-\beta u_3+u_4 \\ \vdots \\ c_{m-2}+u_{m-3}-\beta u_{m-2} \\ c_{m-1}-\beta u_{m-1}+u_m \\ c_m+u_{m-1}-\beta u_m \end{pmatrix}^{(k+\frac{1}{2})} \tag{6.4.66b}$$

where $\beta = g-r$.

The following tables show the results obtained by solving the four problems by the A.G.E. method, also in Tables (6.4.1), (6.4.2) and (6.4.3) we recorded the results obtained by solving the three non-linear problems by the 2-point group N.L.O.R. method (see subsection (6.3.1)), and in Table (6.4.4), the results for solving problem 4 by the S.O.R. method are also included. The convergence test used was the average test (6.2.27) with $\varepsilon=10^{-5}$. For each method, the logarithm of the minimum number of iterations was plotted against $\log(h^{-1})$, the graphs which support the theory are shown in Figures (6.4.1), (6.4.2), (6.4.3) and (6.4.4).

h^{-1}	2-Point Group N.L.O.R.		A.G.E.	
	Opt.Relax.Fac.(ω_b)	No.of Iter.(k)	Opt.Iteration Parameter (r)	k
13	1.474	15	0.444-0.456	19
25	1.68-1.681	29	0.251-0.253	37
37	1.772	41	0.168-0.174	54
49	1.821-1.824	55	0.133-0.135	72

Problem 1

TABLE (6.4.1)

h^{-1}	2-Point Group N.L.O.R.		A.G.E.	
	ω_b	k	r	k
13	1.08-1.096	12	0.772-0.822	11
25	1.386-1.406	21	0.436-0.476	22
37	1.557-1.565	29	0.32-0.33	31
49	1.636-1.672	39	0.246-0.249	42

Problem 2

TABLE (6.4.2)

h^{-1}	2-Point Group N.L.O.R.		A.G.E.	
	ω_b	k	r	k
13	1.084-1.1	12	0.9-0.925	9
25	1.43-1.444	20	0.513-0.522	18
37	1.572-1.592	29	0.362-0.366	27
49	1.679-1.681	37	0.281-0.282	36

Problem 3

TABLE (6.4.3)

h^{-1}	S.O.R.		A.G.E.	
	ω_b	k	r	k
13	1.075-1.077	9	1.75-1.9	5
25	1.206-1.238	16	0.58-0.98	9
37	1.325-1.374	23	0.53-0.59	12
49	1.42-1.47	30	0.25-0.465	16

Problem 4

TABLE (6.4.4)

Finally, the problems were solved using Picard's method and the results are given in Table (6.4.5).

h^{-1}	Problem 1		Problem 2		Problem 3		Problem 4	
	ω_b	k	ω_b	k	ω_b	k	ω_b	k
13	0.953-1.02	5	0.534-0.582	13	0.387-0.41	21	1	2
25	0.954-1.03	5	0.565-0.582	13	0.396-0.412	21	1	2
37	0.94-1.034	5	0.575-0.578	13	0.399-0.412	21	1	2
49	0.942-1.03	5	0.578-0.586	13	0.399-0.414	21	1	2

TABLE (6.4.5)

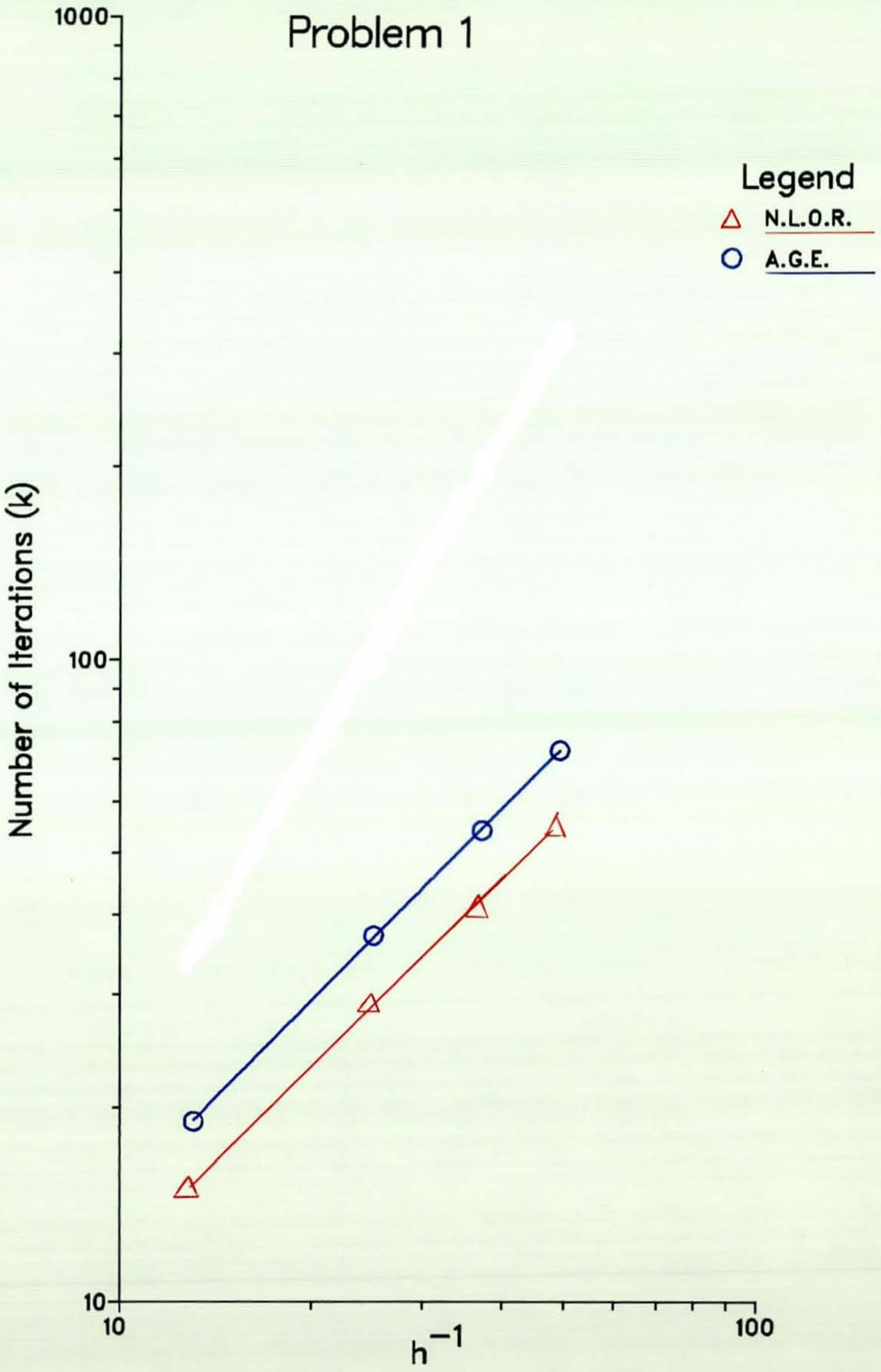

Problem 1
Legend
N.L.O.R.
A.G.E.
1000
100
10
Number of Iterations (k)
10
100
h^{-1}

Figure (6.4.1)

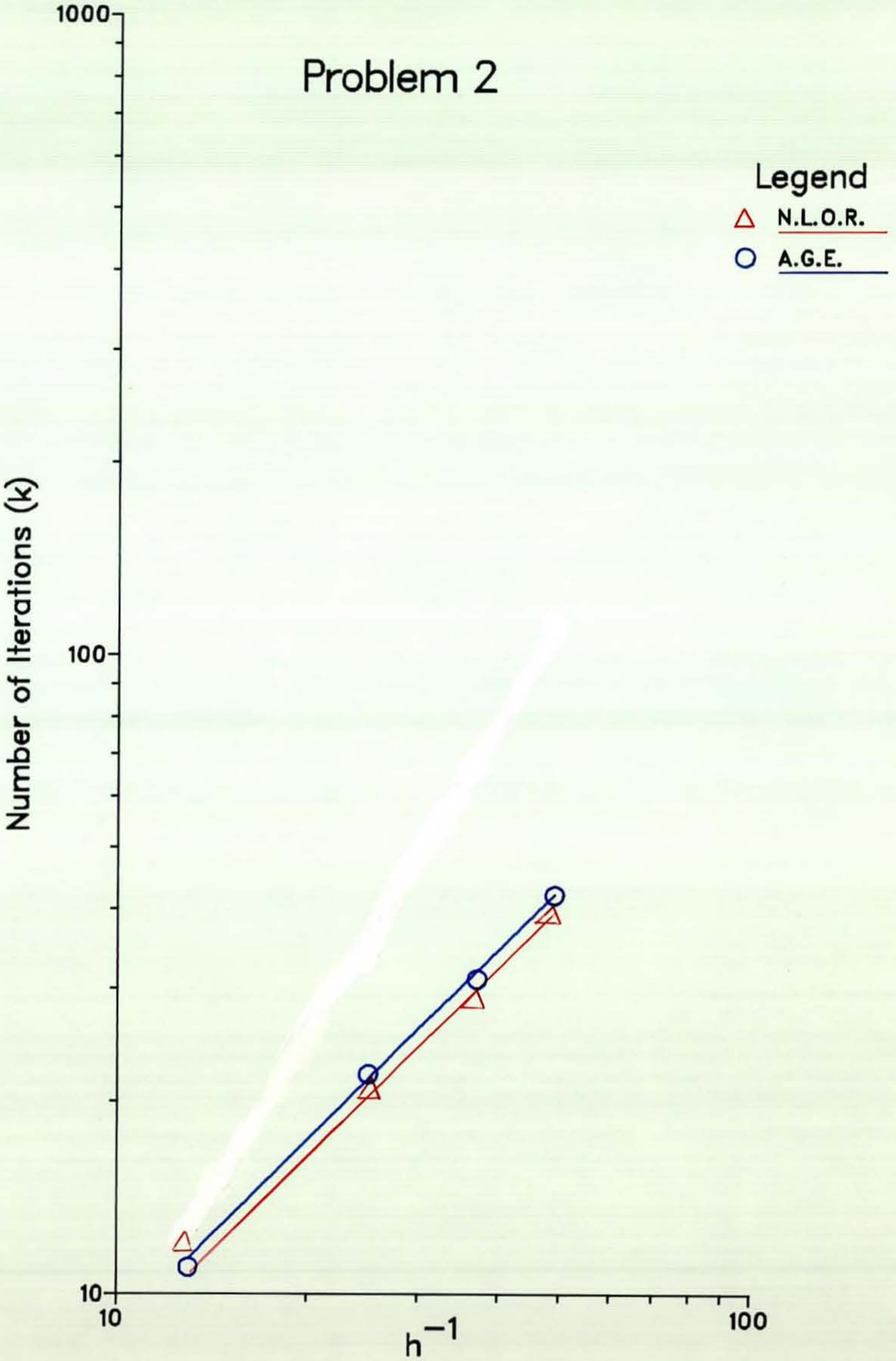

Problem 2
Legend
N.L.O.R.
A.G.E.
1000
100
10
10
100
Number of Iterations (k)
h^{-1}

Figure (6.4.2)

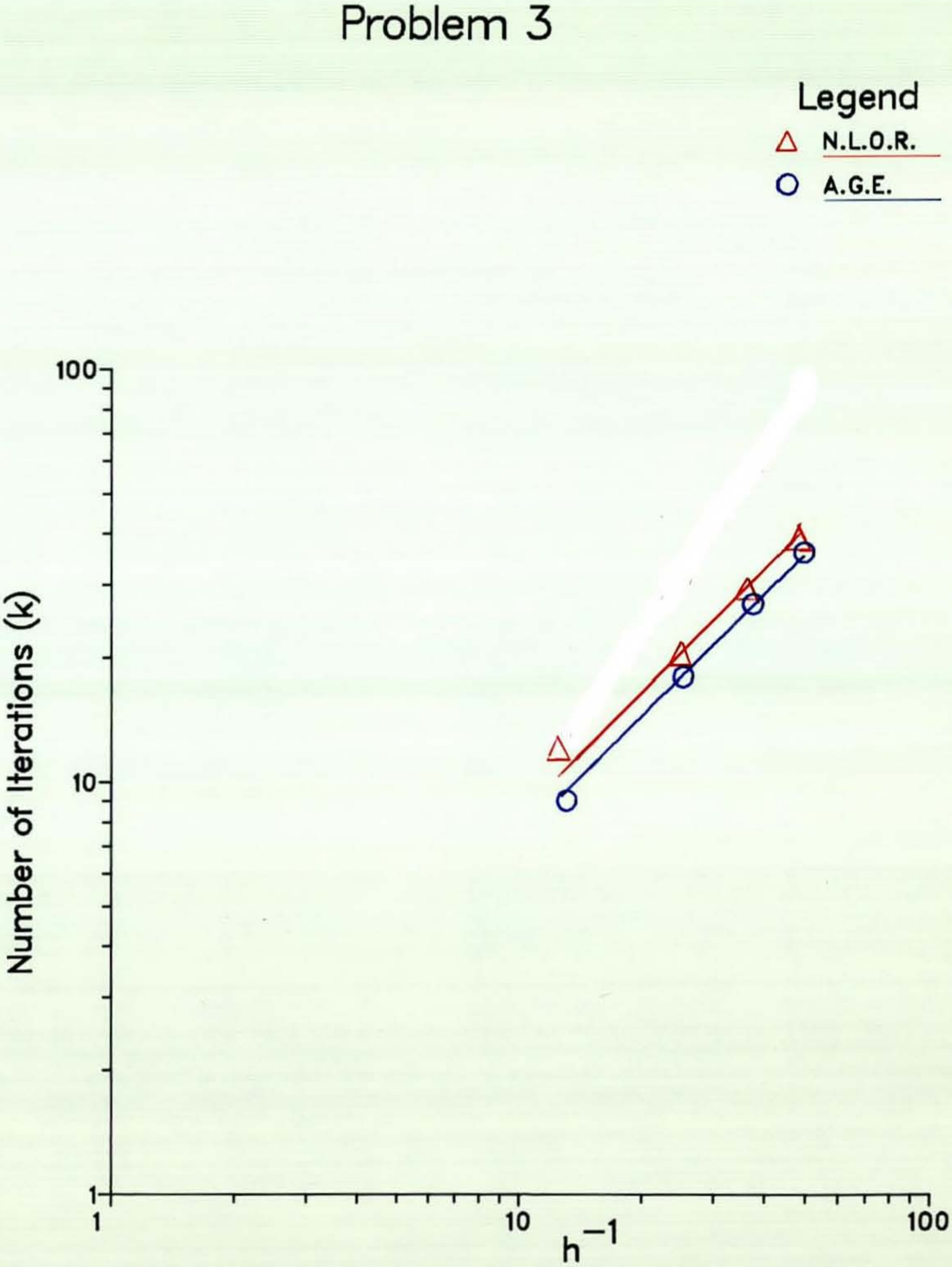

Problem 3
Legend
N.L.O.R.
A.G.E.
100
10
1
1
10
100
Number of Iterations (k)
h^{-1}

Figure (6.4.3)

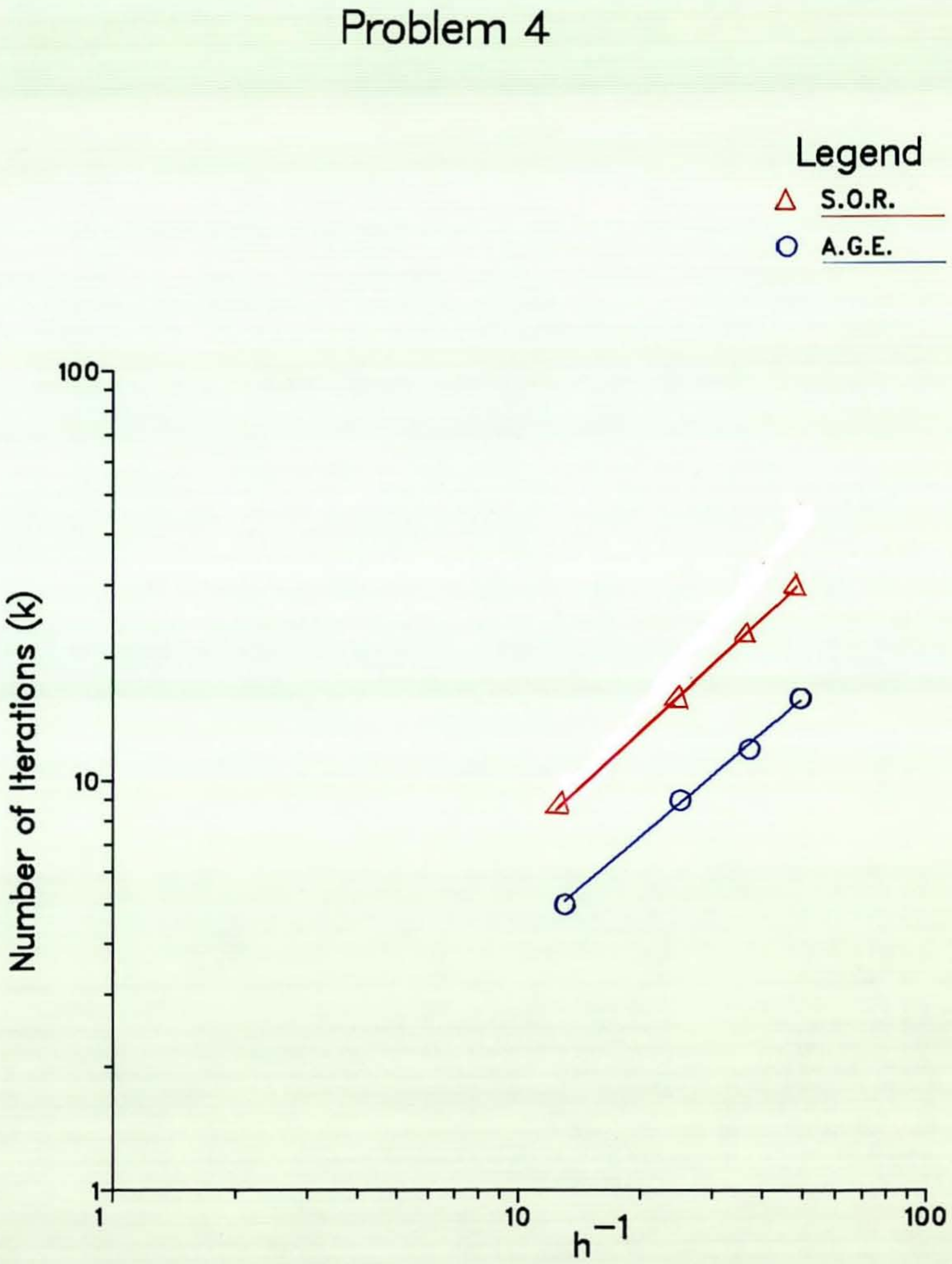

Problem 4
Legend
S.O.R.
A.G.E.
100
10
1
1
10
100
Number of Iterations (k)
h^{-1}

Figure (6.4.4)

6.4.3 THE COMPUTATIONAL COMPLEXITY

Now, in terms of arithmetic calculations, we will estimate the amount of operations required to solve each problem by applying the A.G.E. method.

Problem 1

We first estimate the number of operations required to determine the $(k+\frac{1}{2})$th iterate of the A.G.E. method. From system (6.4.45a), if we assume that,

$$r_1 = c_1+17f_1-\beta u_1^{(k)} \quad \text{and} \quad r_2 = c_2-\beta u_2^{(k)}+g_2u_3^{(k)} ,$$

then,

$$u_1^{(k+\frac{1}{2})} = (\alpha r_1+g_1r_2)d_1 \quad \text{and} \quad u_2^{(k+\frac{1}{2})} = (f_2r_1+\alpha r_2)d_1 , \qquad (6.4.67a)$$

and for $i=3,(2),m-3$

$$u_i^{(k+\frac{1}{2})} = (\alpha r_1' + g_ir_2')d_{\frac{i+1}{2}} ,$$

$$u_{i+1}^{(k+\frac{1}{2})} = (f_{i+1}r_1' + \alpha r_2')d_{\frac{i+1}{2}} , \qquad (6.4.67b)$$

where $r_1' = c_i+f_iu_{i-1}^{(k)}-\beta u_i^{(k)}$ and $r_2' = c_{i+1}-\beta u_{i+1}^{(k)}+g_{i+1}u_{i+2}^{(k)}$,

and finally,

$$u_{m-1}^{(k+\frac{1}{2})} = (\alpha r_1''+g_{m-1}r_2'')d_{\frac{m}{2}} , \quad u_m^{(k+\frac{1}{2})} = (f_mr_1''+\alpha r_2'')d_{\frac{m}{2}} , \qquad (6.4.67c)$$

where $r_1'' = c_{m-1}+f_{m-1}u_{m-2}^{(k)}-\beta u_{m-1}^{(k)}$ and $r_2'' = c_m + \frac{43}{3} g_m-\beta u_m^{(k)}$.

g_i, f_i and c_i for $i=1,2,\ldots,m$ are given in (6.4.38d), d_i $(i=1,2,\ldots,\frac{m}{2})$ is given in (6.4.44a), α and β are given in (6.4.42) and m is divisible by 2.

Hence, the number of operations required for the m mesh points are

$(8m-11)$ multiplications + $(5m-4)$ additions + R.H.S. unit.

In a similar manner, the $(k+1)$th iterate can be determined, and we can show that the number of operations required are

$(8m-4)$ multiplications + $5m$ additions + R.H.S. unit

Hence, the A.G.E. iterative method requires

$(16m-15)$ multiplications + $(10m-4)$ additions + R.H.S. unit

(6.4.68)

for the m mesh points per iteration.

Problems 2 and 3

For these problems and in a similar manner to problem 1 the $(k+\frac{1}{2})^{th}$ and $(k+1)^{th}$ iterates of the A.G.E. method can be obtained, and the number of operations required to solve this problem by the A.G.E. iterative method is

$(12m-7)$ multiplications and $(13m-9)$ additions + R.H.S. unit

(6.4.69)

for the m mesh points per iteration.

Problem 4

For this linear problem, $\underline{u}^{(k+\frac{1}{2})}$ and $\underline{u}^{(k+1)}$ can be obtained from (6.4.66a) and (6.4.66b) respectively. Hence, we can show that the number of operations required to solve this problem by the A.G.E. iterative method is

$(6m-7)$ multiplications + $(4m - \frac{11}{2})$ additions + R.H.S. unit (6.4.70)

per iteration.

On the other hand, to solve this problem by the S.O.R. iterative method we require per iteration

$2m$ multiplications + $(4m-2)$ additions + R.H.S. unit (6.4.71)

Hence, by combining the results shown in Tables (6.4.1), (6.4.2), (6.4.3) and (6.4.4) with the corresponding number of operations per iteration required to solve Problems 1,2 and 3 by the 2-Point Group N.L.O.R. and the A.G.E. methods and Problem 4 by the S.O.R. and A.G.E. methods, we can obtain the total number of arithmetic operations required. These are given in Tables (6.4.5), (6.4.6), (6.4.7) and (6.4.8) for Problems 1,2,3 and 4 respectively.

Method / h^{-1}	2-Point Group N.L.O.R.		A.G.E.	
	Multiplications	Additions	M	A
13	120m-90	90m-60	304m-285	190m-76
25	232m-174	174m-116	592m-555	370m-148
37	328m-246	246m-164	864m-810	540m-216
49	440m-330	330m-220	1152m-1080	720m-288

Problem 1

TABLE (6.4.5)

Method / h^{-1}	2-Point Group N.L.O.R.		A.G.E.	
	M	A	M	A
13	104m-78	58.5m-52	132m-77	143m-99
25	168m-126	94.5m-84	264m-154	286m-198
37	232m-174	130.5m-116	372m-217	403m-279
49	312m-324	175.5m-156	504m-294	546m-378

Problem 2

TABLE (6.4.6)

Method / h^{-1}	2-Point Group N.L.O.R.		A.G.E.	
	M	A	M	A
13	96m-72	54m-48	108m-63	117m-81
25	160m-120	90m-80	216m-126	234m-162
37	232m-174	130m-116	324m-189	351m-243
49	296m-222	166.5m-148	432m-252	468m-324

Problem 3

TABLE (6.4.7)

Method / h^{-1}	S.O.R.		A.G.E.	
	M	A	M	A
13	18m	36m-18	30m-35	20m-27.5
25	32m	64m-32	54m-63	36m-49.5
37	46m	92m-46	72m-84	48m-66
49	60m	120m-60	96m-112	64m-88

Problem 4

TABLE (6.4.8)

The total number of arithmetic operations required to solve the four problems using Picard's method can be obtained and are given in Table (6.4.9).

Problem 1		Problem 2		Problem 3		Problem 4	
M	A	M	A	M	A	M	A
30m	25m	78m	65m	126m	105m	12m	10m

TABLE (6.4.9)

By comparing the total number of arithmetic operations required by the 2-point group N.L.O.R. method and the A.G.E. method to solve the three non-linear problems, Tables (6.4.5), (6.4.6) and (6.4.7) shows that the new A.G.E. method requires more computer time than the N.L.O.R. method. Also, it can be seen from Table (6.4.8) that the A.G.E. algorithm is more efficient than the S.O.R. iterative approach to solve the linear Problem 4.

For the Picard's method, it can be seen from Table (6.4.5) that the rate of convergence appears to be independent of h^{-1} for these class of problems and Table (6.4.9) shows that it requires less arithmetic operations than the other methods to solve the linear and the non-linear problems.

CHAPTER 7

FURTHER APPLICATIONS OF THE 4 AND 9 POINT EXPLICIT GROUP OF MESH POINTS TO:

A. THE 9 POINT FINITE DIFFERENCE EQUATION OF THE LAPLACE OPERATOR.

B. THE 13 POINT FINITE DIFFERENCE EQUATION OF THE BIHARMONIC OPERATOR.

7.1 INTRODUCTION

In this Chapter and in a similar manner to the four and nine point groups of Chapter 4, we now present explicit four and nine point groups to the 9-point finite difference equation. *The well known 9-point approximation* to Laplace's equation is given by,

$$20u_{i,j}-4(u_{i-1,j}+u_{i,j-1}+u_{i+1,j}+u_{i,j+1})-(u_{i-1,j-1}+u_{i+1,j-1}+u_{i+1,j+1}+u_{i-1,j+1}) = 0 , \qquad (7.1.1)$$

and is represented by the computational molecule given in Figure (3.2.4). For this equation, the local discretisation error is $O(h^6)$ (see Section (3.2)), this is, therefore, an extremely accurate approximation to use when solving Laplace's equation.

Further, for completeness in Section (7.4), we present the theoretical results for the explicit four and nine point groups to the 13-point finite difference equation of the biharmonic operator (see Figure (3.2.6)).

7.2 THE 4 AND 9 POINT GROUPS OF THE 9 POINT F.D.E. FOR THE LAPLACE OPERATOR

7.2.1 THE 4-POINT GROUP

Consider the linear matrix equation

$$A\underline{u} = \underline{b}\ , \tag{7.2.1}$$

defined in the unit square $0 \leq x,y \leq 1$, with m^2 internal mesh points (see Figure (4.2.1)) and assume that the mesh points are ordered in groups of four as shown in Figure (7.2.1).

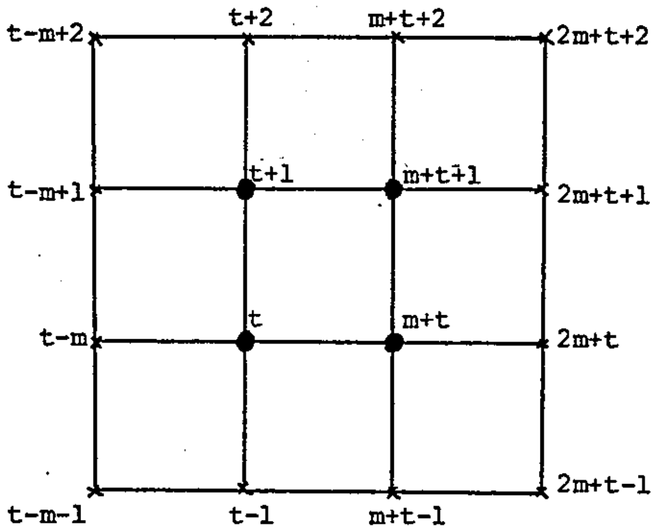


FIGURE (7.2.1)

where $t=rm+1,(2),(r+1)m-1$ and $r=0(2),m-2$ with m taken to be an even number. Hence to use the 9-point approximation scheme, the finite difference equation at the point P of Figure (7.2.2) has the form,

$$u_P+\alpha(u_{R,P}+u_{T,P}+u_{L,P}+u_{B,P})+\beta(u_{TR,P}+u_{TL,P}+u_{BL,P}+u_{BR,P}) = b_P\ . \tag{7.2.2}$$

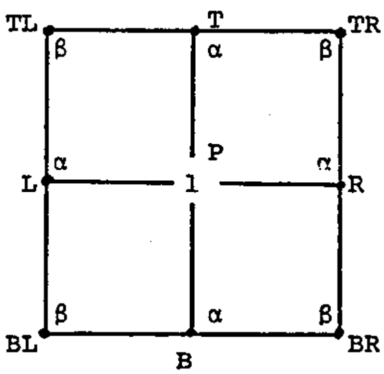


FIGURE (7.2.2)

Now, if we consider equation (7.2.1) defined in a unit square with $\Delta x=\Delta y=h=1/7$ and assume that the groups are ordered in red-black ordering, then by applying equation (7.2.2) to each of the mesh points in turn, the resulting coefficient matrix A has the block form,

$$A = \left[\begin{array}{ccccc|cccc}
R_0 & 0 & R_1 & & & R_5 & R_8 & & \\
0 & R_0 & R_2 & \bigcirc & & R_6 & 0 & R_8 & \bigcirc \\
R_3 & R_4 & R_0 & R_2 & R_1 & R_7 & R_6 & R_5 & R_8 \\
 & & R_4 & R_0 & 0 & & R_7 & 0 & R_5 \\
\bigcirc & & R_3 & 0 & R_0 & \bigcirc & & R_7 & R_6 \\
\hline
R_6 & R_5 & R_8 & & & R_0 & R_2 & R_1 & 0 \\
R_7 & 0 & R_5 & R_8 & \bigcirc & R_4 & R_0 & 0 & R_1 \\
 & R_7 & R_6 & 0 & R_8 & R_3 & 0 & R_0 & R_2 \\
\bigcirc & & R_7 & R_6 & R_5 & 0 & R_3 & R_4 & R_0
\end{array}\right] \tag{7.2.3a}$$

where,

$$R_0 = \begin{bmatrix} 1 & \alpha & \alpha & \beta \\ \alpha & 1 & \beta & \alpha \\ \alpha & \beta & 1 & \alpha \\ \beta & \alpha & \alpha & 1 \end{bmatrix}, \quad R_1 = \left[\begin{array}{cc|cc} \multicolumn{2}{c|}{\bigcirc} & \multicolumn{2}{c}{\bigcirc} \\ \hline 0 & 0 & \multicolumn{2}{c}{\multirow{2}{*}{$\bigcirc$}} \\ \beta & 0 & & \end{array}\right], \quad R_2 = \left[\begin{array}{cc|cc} \multicolumn{2}{c|}{\multirow{2}{*}{$\bigcirc$}} & 0 & 0 \\ & & \beta & 0 \\ \hline \multicolumn{2}{c|}{\bigcirc} & \multicolumn{2}{c}{\bigcirc} \end{array}\right],$$

$$R_3 = \left[\begin{array}{cc|cc} \multicolumn{2}{c|}{\multirow{2}{*}{$\bigcirc$}} & 0 & \beta \\ & & 0 & 0 \\ \hline \multicolumn{2}{c|}{\bigcirc} & \multicolumn{2}{c}{\bigcirc} \end{array}\right], \quad R_4 = \left[\begin{array}{cc|cc} \multicolumn{2}{c|}{\bigcirc} & \multicolumn{2}{c}{\bigcirc} \\ \hline 0 & \beta & \multicolumn{2}{c}{\multirow{2}{*}{$\bigcirc$}} \\ 0 & 0 & & \end{array}\right], \quad R_5 = \left[\begin{array}{cc|cc} \multicolumn{2}{c|}{\bigcirc} & \multicolumn{2}{c}{\bigcirc} \\ \hline \alpha & \beta & \multicolumn{2}{c}{\multirow{2}{*}{$\bigcirc$}} \\ \beta & \alpha & & \end{array}\right],$$

$$R_6 = \left[\begin{array}{cc|cc} 0 & 0 & 0 & 0 \\ \alpha & 0 & \beta & 0 \\ \hline 0 & 0 & 0 & 0 \\ \beta & 0 & \alpha & 0 \end{array}\right], \quad R_7 = \left[\begin{array}{cc|cc} \multicolumn{2}{c|}{\multirow{2}{*}{$\bigcirc$}} & \alpha & \beta \\ & & \beta & \alpha \\ \hline \multicolumn{2}{c|}{\bigcirc} & \multicolumn{2}{c}{\bigcirc} \end{array}\right] \quad \text{and} \quad R_8 = \left[\begin{array}{cc|cc} 0 & \alpha & 0 & \beta \\ 0 & 0 & 0 & 0 \\ \hline 0 & \beta & 0 & \alpha \\ 0 & 0 & 0 & 0 \end{array}\right] \tag{7.2.3b}$$

Notice that the (4×4) submatrices R_1, R_2, R_3 and R_4 have only one non-zero element which makes the two square submatrices of A to be sparse submatrices. Now the explicit group S.O.R. method can be considered as a point S.O.R. method applied to a transformed matrix $A^E=[\text{diag}\{R_0\}]^{-1}A$ and modified vector $\underline{b}^E=[\text{diag}\{R_0\}]^{-1}\underline{b}$. For the linear system $A^E\underline{u}=\underline{b}^E$, with

$A^E=I-L^E-U^E$, the method can be written compactly as,

$$\underline{u}^{(k+1)} = \underline{u}^{(k)} + \omega[L^E\underline{u}^{(k+1)}+U^E\underline{u}^{(k)} + \underline{b}^E - \underline{u}^{(k)}] \;. \tag{7.2.4}$$

The matrix $[\text{diag}\{R_O\}]^{-1}$ is simply $\text{diag}\{R_O^{-1}\}$ and the inverse of R_O is given by,

$$R_O^{-1} = \frac{1}{d} \begin{bmatrix} \gamma_1 & \gamma_2 & \gamma_2 & \gamma_3 \\ \gamma_2 & \gamma_1 & \gamma_3 & \gamma_2 \\ \gamma_2 & \gamma_3 & \gamma_1 & \gamma_2 \\ \gamma_3 & \gamma_2 & \gamma_2 & \gamma_1 \end{bmatrix}, \tag{7.2.5a}$$

where $\quad d=1-2\beta^2-4\alpha^2\beta(\beta^3+\beta-2)$, $\gamma_1 = 1+2\alpha^2\beta(1-\beta)$, $\gamma_2=\gamma(2\beta-\beta^2-1)$

and $\quad \gamma_3 = 2\alpha^2(1-\beta)-\beta(1-\beta^2)$. (7.2.5b)

The block structure of the matrix A^E is the same as that of A with the submatrices R_O, replaced by the identity matrices I, and the submatrices R_i, replaced by $R_O^{-1}R_i$, i=1,2,...,8, which can be easily determined, for example,

$$R_O^{-1}R_1 = \frac{1}{d}\begin{bmatrix} \beta\gamma_3 & 0 & 0 & 0 \\ \beta\gamma_2 & 0 & 0 & 0 \\ \beta\gamma_2 & 0 & 0 & 0 \\ \beta\gamma_1 & 0 & 0 & 0 \end{bmatrix}, \quad R^{-1}R_5 = \frac{1}{d}\begin{bmatrix} \alpha\gamma_2+\beta\gamma_3 & \beta\gamma_2+\alpha\gamma_3 & 0 & 0 \\ \alpha\gamma_3+\beta\gamma_2 & \beta\gamma_3+\alpha\gamma_2 & 0 & 0 \\ \alpha\gamma_1+\beta\gamma_2 & \beta\gamma_1+\alpha\gamma_2 & 0 & 0 \\ \beta\gamma_2+\alpha\gamma_1 & \alpha\gamma_2+\beta\gamma_1 & 0 & 0 \end{bmatrix},$$

whilst $R_O^{-1}R_2$, $R_O^{-1}R_3$, $R_O^{-1}R_4$, $R_O^{-1}R_6$, $R_O^{-1}R_7$ and $R_O^{-1}R_8$ can also be obtained in a similar manner. For the model problem (Laplace's equation) $\alpha = -1/5$ and $\beta = -1/20$, it follows that R_O^{-1} is a symmetric matrix given by,

$$R_O^{-1} = \frac{1}{2079}\begin{bmatrix} 2320 & 560 & 560 & 340 \\ 560 & 2320 & 340 & 560 \\ 560 & 340 & 2320 & 560 \\ 340 & 560 & 560 & 2320 \end{bmatrix},$$

and hence,

$$R_O^{-1}R_1 = -\frac{1}{2079}\begin{bmatrix} 17 & 0 & 0 & 0 \\ 28 & 0 & 0 & 0 \\ 28 & 0 & 0 & 0 \\ 116 & 0 & 0 & 0 \end{bmatrix}, \quad R_O^{-1}R_5 = -\frac{1}{693}\begin{bmatrix} 43 & 32 & 0 & 0 \\ 32 & 43 & 0 & 0 \\ 164 & 76 & 0 & 0 \\ 76 & 164 & 0 & 0 \end{bmatrix}.$$

Similarly, $R_O^{-1}R_i$, i=2,3,4,6,7 and 8 can be obtained and hence the computational molecule of the explicit 4 point group can now be established at the point P to be

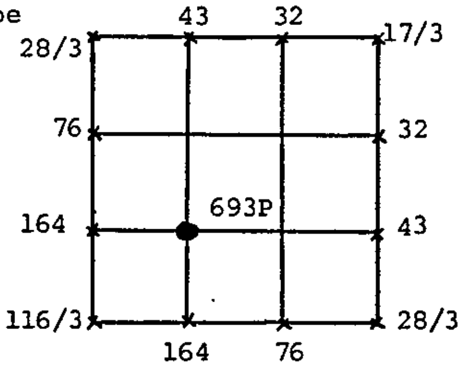


FIGURE 7.2.3

From the explicit equations derived from the new molecule, the explicit group Jacobi method can be obtained and is given by,

$$u_t^{(k+1)} = \frac{1}{693}[164(u_{t-m}^{(k)}+u_{t-1}^{(k)})+76(u_{t-m+1}^{(k)}+u_{m+t-1}^{(k)})+43(u_{t+2}^{(k)}+u_{2m+t}^{(k)})+$$
$$32(u_{m+t+2}^{(k)}+u_{2m+t+1}^{(k)})+\frac{1}{3}\{116u_{t-m-1}^{(k)}+28(u_{t-m+2}^{(k)}+u_{2m+t-1}^{(k)})+17u_{2m+t+2}^{(k)}\}],$$

$$u_{t+1}^{(k+1)} = \frac{1}{693}[164(u_{t-m+1}^{(k)}+u_{t+2}^{(k)})+76(u_{t-m}^{(k)}+u_{m+t+2}^{(k)})+43(u_{t-1}^{(k)}+u_{2m+t+1}^{(k)})+$$
$$32(u_{m+t-1}^{(k)}+u_{2m+t}^{(k)})+\frac{1}{3}\{116u_{t-m+2}^{(k)}+28(u_{t-m-1}^{(k)}+u_{2m+t+2}^{(k)})+17u_{2m+t-1}^{(k)}\}],$$

$$u_{m+t}^{(k+1)} = \frac{1}{693}[164(u_{m+t-1}^{(k)}+u_{2m+t}^{(k)})+76(u_{t-1}^{(k)}+u_{2m+t+1}^{(k)})+43(u_{t-m}^{(k)}+u_{m+t+2}^{(k)})+$$
$$32(u_{t-m+1}^{(k)}+u_{t+2}^{(k)})+\frac{1}{3}\{116u_{2m+t-1}^{(k)}+28(u_{t-m-1}^{(k)}+u_{2m+t+2}^{(k)})+17u_{t-m+2}^{(k)}\}],$$

and

$$u_{m+t+1}^{(k+1)} = \frac{1}{693}[164(u_{m+t+2}^{(k)}+u_{2m+t+1}^{(k)})+76(u_{t+2}^{(k)}+u_{2m+t}^{(k)})+43(u_{t-m+1}^{(k)}+u_{m+t-1}^{(k)})+$$
$$32(u_{t-m}^{(k)}+u_{t-1}^{(k)})+\frac{1}{3}\{116u_{2m+t+2}^{(k)}+28(u_{t-m+2}^{(k)}+u_{2m+t-1}^{(k)})+17u_{t-m-1}^{(k)}\}],$$

(7.2.6)

where t=rm+1,(2),(r+1)m-1 and r=0,(2),m-2. It can be noticed that $u_{t+1}^{(k+1)}$, $u_{m+t+1}^{(k+1)}$ and $u_{m+t}^{(k+1)}$ can be obtained by rotating the molecule in a clockwise manner.

A similar set of equations can be obtained for the 4-point group G.S., J.O.R. and S.O.R. iterative methods.

7.2.2 THE 9-POINT GROUP

In this case, the mesh points of Figure (4.2.1) are ordered in groups of nine as shown in Figure (7.2.4).

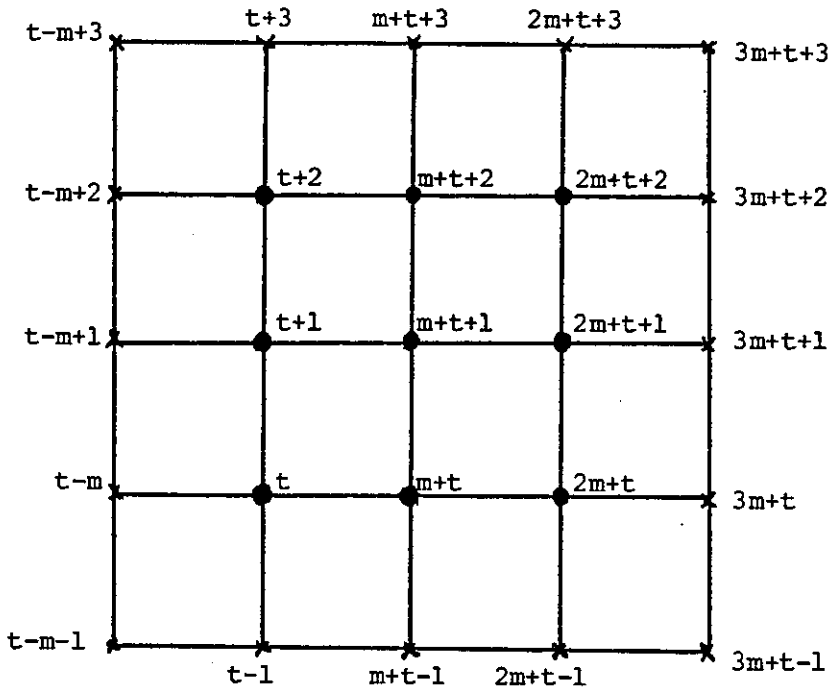


FIGURE 7.2.4

where t=rm+1,(3),(r+1)m-2 and r=0,(3),m-3. For this scheme of the grouping of the mesh points m obviously must be divisible by 3.

For the simple case of the unit square and $\Delta x=\Delta y=h=1/7$, then with a red-black ordering of the 9-point groups and by applying equation (7.2.2) to each of the mesh points in turn, the resulting coefficient matrix can be shown to have the block structure

$$A = \left[\begin{array}{cc|cc} R_0 & R_1 & R_5 & R_6 \\ R_3 & R_0 & R_8 & R_7 \\ \hline R_7 & R_6 & R_0 & R_2 \\ R_8 & R_5 & R_4 & R_0 \end{array}\right] \qquad (7.2.7a)$$

where

$$R_0 = \left(\begin{array}{ccc|ccc|ccc} 1 & \alpha & 0 & \alpha & \beta & 0 & & & \\ \alpha & 1 & \alpha & \beta & \alpha & \beta & & O & \\ 0 & \alpha & 1 & 0 & \beta & \alpha & & & \\ \hline \alpha & \beta & 0 & 1 & \alpha & 0 & \alpha & \beta & 0 \\ \beta & \alpha & \beta & \alpha & 1 & \alpha & \beta & \alpha & \beta \\ 0 & \beta & \alpha & 0 & \alpha & 1 & 0 & \beta & \alpha \\ \hline & & & \alpha & \beta & 0 & 1 & \alpha & 0 \\ & O & & \beta & \alpha & \beta & \alpha & 1 & \alpha \\ & & & 0 & \beta & \alpha & 0 & \alpha & 1 \end{array}\right), \quad R_1 = \left(\begin{array}{c|c} O & O \\ \hline \\ \beta & O \end{array}\right)$$

$$R_2 = \left(\begin{array}{c|c} O & \\ & \beta \\ \hline O & O \end{array}\right), \quad R_3 = \left(\begin{array}{c|c} & \beta \\ O & \\ \hline O & O \end{array}\right)$$

$$R_4 = \left(\begin{array}{c|c} O & O \\ \hline \beta & \\ & O \end{array}\right), \quad R_5 = \left(\begin{array}{ccc|c} & O & & O \\ \hline \alpha & \beta & 0 & \\ \beta & \alpha & \beta & O \\ 0 & \beta & \alpha & \end{array}\right)$$

$$R_6 = \left(\begin{array}{c|c|c} & & O \\ \alpha & \beta & \\ \hline & & \\ \beta & \alpha & \beta \\ \hline O & & \\ & \beta & \alpha \end{array}\right), \quad R_7 = \left(\begin{array}{c|ccc} & \alpha & \beta & 0 \\ O & \beta & \alpha & \beta \\ & 0 & \beta & \alpha \\ \hline O & & O & \end{array}\right)$$

and
$$R_8 = \left(\begin{array}{c:c:c} \alpha & \beta & O \\ \hdashline \beta & \alpha & \beta \\ \hdashline O & \beta & \alpha \end{array}\right) \qquad (7.2.7b)$$

To derive the explicit block S.O.R. method we follow the same steps as for the 4-point group, hence we have to determine the inverse of R_0 which is obtained by using REDUCE and is given by,

$$R_0^{-1} = \frac{1}{d} \begin{pmatrix} \gamma_1 & \gamma_2 & \gamma_3 & \gamma_2 & \gamma_4 & \gamma_5 & \gamma_3 & \gamma_5 & \gamma_6 \\ \gamma_2 & \gamma_7 & \gamma_2 & \gamma_4 & \gamma_8 & \gamma_4 & \gamma_5 & \gamma_9 & \gamma_5 \\ \gamma_3 & \gamma_2 & \gamma_1 & \gamma_5 & \gamma_4 & \gamma_2 & \gamma_6 & \gamma_5 & \gamma_3 \\ \gamma_2 & \gamma_4 & \gamma_5 & \gamma_7 & \gamma_8 & \gamma_9 & \gamma_2 & \gamma_4 & \gamma_5 \\ \gamma_4 & \gamma_8 & \gamma_4 & \gamma_8 & \gamma_{10} & \gamma_8 & \gamma_4 & \gamma_8 & \gamma_4 \\ \gamma_5 & \gamma_4 & \gamma_2 & \gamma_9 & \gamma_8 & \gamma_7 & \gamma_5 & \gamma_4 & \gamma_2 \\ \gamma_3 & \gamma_5 & \gamma_6 & \gamma_2 & \gamma_4 & \gamma_5 & \gamma_1 & \gamma_2 & \gamma_3 \\ \gamma_5 & \gamma_9 & \gamma_5 & \gamma_4 & \gamma_8 & \gamma_4 & \gamma_2 & \gamma_7 & \gamma_2 \\ \gamma_6 & \gamma_5 & \gamma_3 & \gamma_5 & \gamma_4 & \gamma_2 & \gamma_3 & \gamma_2 & \gamma_1 \end{pmatrix} \qquad (7.2.8a)$$

where,

$$d = 1-4[8\alpha^6(4\beta^2-4\beta+1)-\alpha^4(16\beta^4+24\beta^2-32\beta+9)+\alpha^2(16\beta^4-8\beta+3)-2\beta^2(2\beta^2-1)]$$

$$\gamma_1 = 1-4\alpha^6(8\beta^2-10\beta+3)+2\alpha^4(8\beta^4+26\beta^2-36\beta+11)-2\alpha^2(16\beta^4+2\beta^2-13\beta+5)+\beta^2(12\beta^2-7),$$

$$\gamma_2 = -\alpha[10\alpha^4(\beta^2-4\beta+1)-\alpha^2(16\beta^4+16\beta^2-24\beta+7)+8\beta^4-2\beta^2-2\beta+1,$$

$$\gamma_3 = 4\alpha^6(8\beta^2-6\beta+1)-4\alpha^4(4\beta^4+7\beta^2-6\beta+1)+\alpha^2(16\beta^4+4\beta^2-6\beta+1)-\beta^2(4\beta^2-1),$$

$$\gamma_4 = -8\alpha^6(2\beta-1)+4\alpha^4(4\beta^3+3\beta-2)-2\alpha^2(8\beta^3-1)+\beta(4\beta^2-1),$$

$$\gamma_5 = -\alpha[-6\alpha^4(4\beta^2-4\beta+1)+\alpha^2(16\beta^4-8\beta+3)-2\beta(4\beta^3-3\beta+1)] ,$$

$$\gamma_6 = -4\alpha^6(8\beta^2-10\beta+3)+2\alpha^4(8\beta^4-6\beta^2-4\beta+3)+6\alpha^2\beta(2\beta-1)-\beta^2(4\beta^2-1),$$

$$\gamma_7 = 8\alpha^6(2\beta-1)+6\alpha^4(4\beta^2-8\beta+3)-\alpha^2(16\beta^4-20\beta+9)+8\beta^4-6\beta^6+1,$$

$$\gamma_8 = -\alpha[4\alpha^4(4\beta^2-4\beta+1)-4\alpha^2(4\beta^2-4\beta+1)+4\beta^2-4\beta+1]$$

$$\gamma_9 = 8\alpha^6(2\beta-1)-2\alpha^4(20\beta^2-8\beta-1)+\alpha^2(16\beta^4+16\beta^2-12\beta+1)-2\beta^2(4\beta^2-1),$$

and

$$\gamma_{10} = 16\alpha^6(2\beta-1)-4\alpha^4(4\beta^2+8\beta-5)+8\alpha^2(2\beta^2+\beta-1)-4\beta^2+1 \; . \qquad (7.2.8b)$$

Again, it is clear that the block structure of the matrix A^E, $A^E=[\text{diag}\{R_0^{-1}\}]A$, is the same as that of A with the submatrices R_0, replaced by identity matrices, and the submatrices R_i, replaced by $R_0^{-1}R_i$, $i=1,2,\ldots,8$.

For the Laplace equation, $\alpha=-1/5$ and $\beta=-1/20$, it follows that R_0^{-1} is a symmetric matrix and is given by,

$$R_0^{-1} = \frac{1}{49588}\begin{bmatrix} 56367 & & & & & & & & \\ 15510 & 60990 & & & & & & & \\ 4623 & 15510 & 56367 & & & & \text{symmetric} & & \\ 15510 & 11500 & 4730 & 60990 & & & & & \\ 11500 & 20240 & 11500 & 20240 & 68080 & & & & \\ 4730 & 11500 & 15510 & 7090 & 20240 & 60990 & & & \\ 4623 & 4730 & 2467 & 15510 & 11500 & 4730 & 56367 & & \\ 4730 & 7090 & 4730 & 11500 & 20240 & 11500 & 15510 & 60990 & \\ 2467 & 4730 & 4623 & 4730 & 11500 & 15510 & 4623 & 15510 & 56367 \end{bmatrix}$$

Hence,

$$R_0^{-1}R_3 = -\frac{1}{991760}\begin{bmatrix} 0 & & 0 & 56367 \\ 0 & & 0 & 15510 \\ 0 & & 0 & 4623 \\ 0 & \bigcirc & 0 & 15510 \\ 0 & & 0 & 11500 \\ 0 & & 0 & 4730 \\ 0 & & 0 & 4623 \\ 0 & & 0 & 4730 \\ 0 & & 0 & 2467 \end{bmatrix}_{9\times 9}$$

$$R_0^{-1}R_5 = -\frac{1}{495880}\left[\begin{array}{ccc|c} 11611 & 13005 & 7299 & \\ 13005 & 18910 & 13005 & \\ 7299 & 13005 & 11611 & \\ 36770 & 33120 & 15210 & \\ 33120 & 51980 & 33120 & \\ 15210 & 33120 & 36770 & O \\ 120489 & 61515 & 17001 & \\ 61515 & 13749 & 61515 & \\ 17001 & 61515 & 120489 & \end{array}\right]_{9\times 9},$$

Also, $R_0^{-1}R_1$, $R_0^{-1}R_2$, $R_0^{-1}R_4$, $R_0^{-1}R_6$, $R_0^{-1}R_7$ and $R_0^{-1}R_8$ can be determined similarly. The computational molecule can now be established and is given representatively in Figure (7.2.5) for the points P_1, P_2 and P_3.

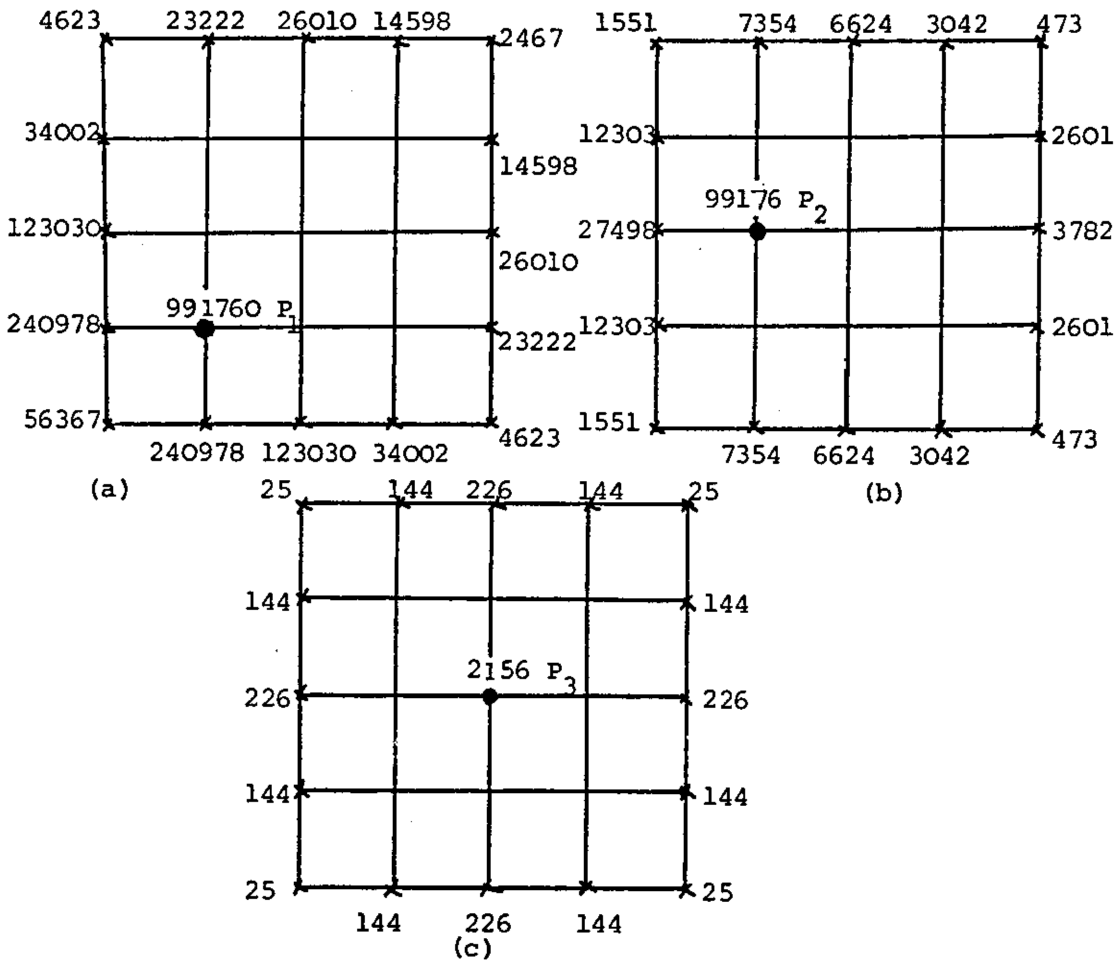


FIGURE 7.2.5

Hence, an explicit group Gauss-Seidel method can be derived from this molecule and is given by,

$$u_t^{(k+1)} = \frac{1}{991760}\,[240978(u_{t-m}^{(k+1)}+u_{t-1}^{(k+1)})+123030(u_{t-m+1}^{(k+1)}+u_{m+t-1}^{(k+1)})+56367u_{t-m-1}^{(k+1)}$$
$$+34002(u_{t-m+2}^{(k+1)}+u_{2m+t-1}^{(k+1)})+26010(u_{m+t+3}^{(k)}+u_{3m+t+1}^{(k)})+23222(u_{t+3}^{(k)}+u_{3m+t}^{(k)})$$
$$+14598(u_{2m+t+3}^{(k)}+u_{3m+t+2}^{(k)})+4623(u_{t-m+3}^{(k)}+u_{3m+t-1}^{(k+1)})+2467u_{3m+t+3}^{(k)}] .$$

By rotating the points of Figure (7.2.5a), a completely analogous expression can be obtained for the points $u_{t+2}^{(k+1)}$, $u_{2m+t+2}^{(k+1)}$ and $u_{2m+t}^{(k+1)}$.

In a similar manner, we have the equation,

$$u_{t+1}^{(k+1)} = \frac{1}{99176}\,[27498u_{t-m+1}^{(k+1)}+12303(u_{t-m}^{(k+1)}+u_{t-m+2}^{(k+1)})+7354(u_{t-1}^{(k+1)}+u_{t+3}^{(k)})$$
$$+6624(u_{m+t-1}^{(k+1)}+u_{m+t+3}^{(k)})+3782u_{3m+t+1}^{(k)}+3042(u_{2m+t-1}^{(k+1)}+u_{2m+t+3}^{(k)})+2601(u_{3m+t}^{(k)}$$
$$+u_{3m+t+2}^{(k)})+1551(u_{t-m-1}^{(k+1)}+u_{t-m+3}^{(k)})+473(u_{3m+t-1}^{(k+1)}+u_{3m+t+3}^{(k)})] ,$$

while mesh points $u_{m+t+2}^{(k+1)}$, $u_{2m+t+1}^{(k+1)}$ and $u_{m+t}^{(k+1)}$ are found by formulae similar to the one above.

Finally, we have the equation,

$$u_{m+t+1}^{(k+1)} = \frac{1}{2156}\,[226(u_{t-m+1}^{(k+1)}+u_{m+t-1}^{(k+1)}+u_{m+t+3}^{(k)}+u_{3m+t+1}^{(k)})+144(u_{t-m}^{(k+1)}+u_{t-1}^{(k+1)}+$$
$$u_{2m+t-1}^{(k+1)}+u_{3m+t}^{(k)}+u_{3m+t+2}^{(k)}+u_{2m+t+3}^{(k)}+u_{t-m+2}^{(k+1)}+u_{t+3}^{(k)})+25(u_{t-m-1}^{(k+1)}+u_{t-m+3}^{(k)}+$$
$$u_{3m+t-1}^{(k+1)}+u_{3m+t+3}^{(k)})] . \qquad (7.2.9)$$

Clearly, similar sets of equations can be written down for the 9-point Jacobi, J.O.R. and S.O.R. iterative methods.

7.3 EXPERIMENTAL RESULTS FOR THE GROUP EXPLICIT S.O.R. METHODS

We now describe some numerical experiments which were carried out in order to compare the new 4 and 9 point group iterative S.O.R. methods with the basic point S.O.R. method. All the results were obtained using the Prime Computer at Loughborough University.

The problem is that of solving Laplace's equation

$$\left.\begin{aligned} &\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} = 0, \quad && \text{in } R, \\ \text{and} \quad &U = 0 && \text{on } \partial R. \end{aligned}\right\} \qquad (7.3.1)$$

where R is a unit square with boundary ∂R.

Although the matrix of the nine-point formula does not possess Property (A), the optimum relaxation factor (ω_b) for the point S.O.R. method for a rectangular region can be obtained from a separation of variable technique. van de Vooren and Vliegenthart (1967) have examined the formula for the Laplace equation in a square of side a, with equal mesh sizes and they obtain the result,

$$\omega_b = 2-2.116\,\frac{\pi h}{a} + 2.24\left(\frac{\pi h}{a}\right)^2 + O(h^3)\ . \qquad (7.3.2)$$

To determine the optimum relaxation factor for the 4 and 9 point explicit group S.O.R. methods, we first evaluate the spectral radius of the Jacobi scheme which can be estimated by using (4.3.4), then substitute the result in (4.3.3) for ω_b. The theoretical number of iterations is obtained by (4.3.6), whilst the convergence test used was the average test (4.3.5) where $\varepsilon=5\times10^{-6}$. The experimental optimum values of ω are determined to within ±0.001. The results obtained from these experiments are recorded in Tables (7.3.1), (7.3.2) and (7.3.3), and for each method the logarithm of the minimum number of iterations was plotted against $\log(h^{-1})$, where h is the mesh size, the graphs shown in Figure (7.3.1) reveal that the plots for the methods were all straight lines with slope of unity thus supporting the S.O.R. theory.

h^{-1}	ω_b		k	
	Theory	Experiment	Th.	Exp.
13	1.619	1.595-1.604	26	33
25	1.769	1.761-1.765	47	62
37	1.836	1.829-1.835	68	92
49	1.873	1.868	90	121

TABLE 7.3.1: Point S.O.R.

h^{-1}	ω_b		k	
	Th.	Exp.	Th.	Exp.
13	1.496	1.509-1.51	17	19
25	1.699	1.702-1.709	34	37
37	1.786	1.788	51	53
49	1.834	1.834-1.837	67	71

TABLE 7.3.2: 4-Point Group S.O.R.

h^{-1}	ω_b		k	
	Th.	Exp.	Th.	Exp.
13	1.419	1.435-1.447	14	17
25	1.644	1.649-1.657	28	31
37	1.744	1.748	41	44
49	1.80	1.802-1.805	55	59

TABLE 7.3.3: 9-Point Group S.O.R.

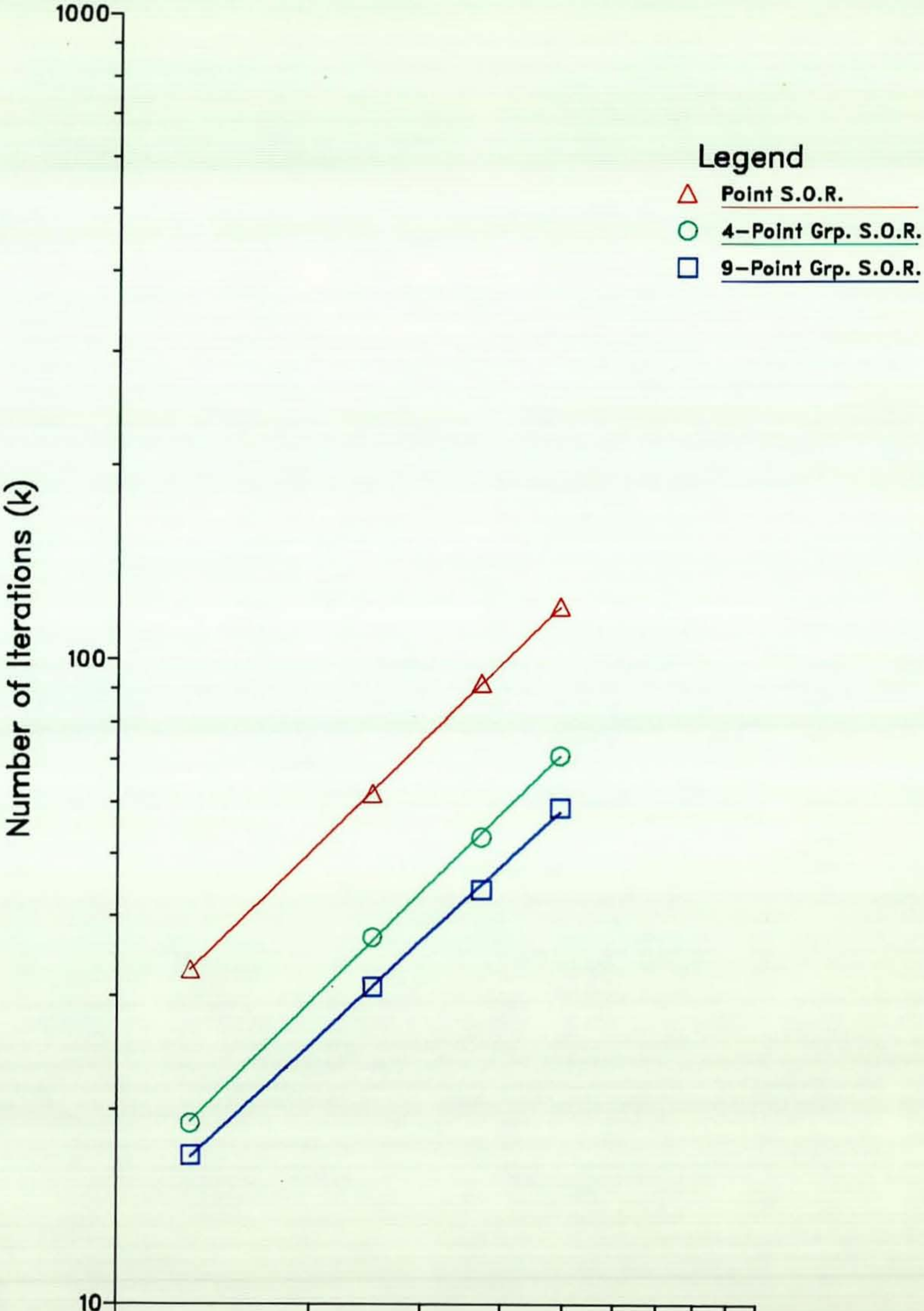

1000
100
10
Number of Iterations (k)
10
100
h^{-1}
Legend
Point S.O.R.
4-Point Grp. S.O.R.
9-Point Grp. S.O.R.

Figure (7.3.1)

We now consider the relative efficiencies of the three methods, and as stated on page (95), we must compare the amount of work required per iteration, as well as the number of iterations required for convergence.

The Point S.O.R. Method

In implementing the point S.O.R. method, the 9-point finite difference solution of the model problem (7.3.1) is given by

$$u_{i,j}^{(k+1)} = \omega(\tilde{u}_{i,j}^{(k+1)} - u_{i,j}^{(k)}) + u_{i,j}^{(k)} \quad , \tag{7.3.3}$$

where the $\tilde{u}_{i,j}^{(k+1)}$ are the components of the $(k+1)^{th}$ G.S. iteration and is defined by,

$$\tilde{u}_{i,j}^{(k+1)} = 0.2(u_{i-1,j}^{(k+1)}+u_{i,j-1}^{(k+1)}+u_{i+1,j}^{(k)}+u_{i,j+1}^{(k)})+0.05(u_{i-1,j-1}^{(k+1)}+u_{i+1,j-1}^{(k+1)}$$
$$+(u_{i+1,j+1}^{(k)}+u_{i-1,j+1}^{(k)}) \ . \tag{7.3.4}$$

Hence, the number of operations required is

$$3m^2 \text{ multiplications} + 9m^2 \text{ additions,} \tag{7.3.5}$$

for the m^2 internal mesh points per iteration.

The 4-Point Group S.O.R. Method

For this grouping, a set of G.S. iteration equations similar to the equations given in (7.2.6) can be calculated. To determine the amount of work, first we calculate $\tilde{u}_{i,j}^{(k+1)}$, $\tilde{u}_{i+1,j}^{(k+1)}$ and $\tilde{u}_{i+1,j+1}^{(k+1)}$ then we apply the 9-point finite difference formula to $\tilde{u}_{i,j+1}^{(k+1)}$. Hence, if we set,

$$\left.\begin{aligned}
s_1 &= u_{i-1,j}^{(k+1)}+u_{i,j-1}^{(k+1)} \ , & s_2 &= u_{i-1,j+1}^{(k+1)}+u_{i+1,j-1}^{(k+1)} \ ,\\
s_3 &= u_{i,j+2}^{(k)}+u_{i+2,j}^{(k)} \ , & s_4 &= u_{i+1,j+2}^{(k)}+u_{i+2,j+1}^{(k)} \ ,\\
s_5 &= u_{i-1,j+2}^{(k)}+u_{i+2,j-1}^{(k+1)} \ , & s_6 &= u_{i+1,j-1}^{(k+1)}+u_{i+2,j}^{(k)} \ ,
\end{aligned}\right|$$

$$\left.\begin{aligned} s_7 &= u_{i-1,j+1}^{(k+1)}+u_{i,j+2}^{(k)} \;, & s_8 &= u_{i,j-1}^{(k+1)}+u_{i+2,j+1}^{(k)} \;, \\ s_9 &= u_{i-1,j}^{(k+1)}+u_{i+1,j+2}^{(k)} \;, & s_{10} &= u_{i-1,j-1}^{(k+1)}+u_{i+2,j+2}^{(k)} \;, \end{aligned}\right\}$$

and $s_{11} = 28s_5$. (7.3.6)

Then we have,

$$\left.\begin{aligned} \tilde{u}_{i,j}^{(k+1)} &= \frac{1}{2079}(492s_1+228s_2+129s_3+116u_{i-1,j-1}^{(k+1)}+96s_4+s_{11}+17u_{i+2,j+2}^{(k)}), \\ \tilde{u}_{i+1,j}^{(k+1)} &= \frac{1}{2079}(492s_6+228s_8+129s_9+116u_{i+2,j-1}^{(k+1)}+96s_7+28s_{10}+17u_{i-1,j+2}^{(k)}), \\ \tilde{u}_{i+1,j+1}^{(k+1)} &= \frac{1}{2079}(492s_4+228s_3+129s_2+116u_{i+2,j+2}^{(k)}+96s_1+s_{11}+17u_{i-1,j-1}^{(k+1)}), \\ \text{and} \\ \tilde{u}_{i,j+1}^{(k+1)} &= 0.2(\tilde{u}_{i,j}^{(k+1)}+\tilde{u}_{i+1,j+1}^{(k+1)}+s_7)+0.05(\tilde{u}_{i+1,j}^{(k+1)}+s_9+u_{i-1,j+2}^{(k+1)}) \end{aligned}\right\} \tag{7.3.7}$$

Thus, the $(k+1)^{th}$ iterates of the S.O.R. iterative scheme are given by,

$$u_{i+\ell_1,j+\ell_2}^{(k+1)} = \omega(\tilde{u}_{i+\ell_1,j+\ell_2}^{(k+1)}-u_{i+\ell_1,j+\ell_2}^{(k)})+u_{i+\ell_1,j+\ell_2}^{(k)} \tag{7.3.8}$$

for $i,j=1,(2),m-1$ and $\ell_1,\ell_2=0,1$.

So, by assuming that the constant $\frac{1}{2079}$ is stored beforehand, this scheme requires per iteration,

$$\frac{13}{2}m^2 \text{ multiplications } + \frac{41}{4}m^2 \text{ additions}, \tag{7.3.9}$$

for the m^2 internal mesh points.

The 9-Point Group S.O.R. Method

For this method, we first calculate $\tilde{u}_{i+1,j}^{(k+1)}, \tilde{u}_{i,j+1}^{(k+1)}, \tilde{u}_{i+2,j+1}^{(k+1)}, \tilde{u}_{i+1,j+2}^{(k+1)}$ and $\tilde{u}_{i+1,j+1}^{(k+1)}$ by using equations similar to those given in (7.2.9), then apply the 9-point finite difference formula to the remaining points. If we set

$$s_1 = u^{(k+1)}_{i,j-1} + u^{(k+1)}_{i+2,j-1}\ , \qquad s_2 = u^{(k+1)}_{i-1,j}+u^{(k)}_{i+3,j}\ ,$$

$$s_3 = u^{(k+1)}_{i-1,j-1}+u^{(k+1)}_{i+3,j-1}\ , \qquad s_4 = u^{(k+1)}_{i-1,j+1}+u^{(k)}_{i+3,j+1}\ ,$$

$$s_5 = u^{(k+1)}_{i-1,j+2}+u^{(k)}_{i+3,j+2}\ , \qquad s_6 = u^{(k)}_{i,j+3}+u^{(k)}_{i+2,j+3}\ ,$$

$$s_7 = u^{(k)}_{i-1,j+3}+u^{(k)}_{i+3,j+3}\ , \qquad s_8 = u^{(k+1)}_{i-1,j}+u^{(k+1)}_{i-1,j+2}\ ,$$

$$s_9 = u^{(k+1)}_{i,j-1}+u^{(k)}_{i,j+3}\ , \qquad s_{10} = u^{(k+1)}_{i-1,j-1}+u^{(k)}_{i-1,j+3}\ ,$$

$$s_{11} = u^{(k+1)}_{i+1,j-1}+u^{(k)}_{i+1,j+3}\ , \qquad s_{12} = u^{(k+1)}_{i+2,j-1}+u^{(k)}_{i+2,j+3}\ ,$$

$$s_{13} = u^{(k)}_{i+3,j}+u^{(k)}_{i+3,j+2}\ , \qquad s_{14} = u^{(k+1)}_{i+3,j-1}+u^{(k)}_{i+3,j+3}\ ,$$

$$s_{15} = s_4+s_{11},\quad s_{16} = s_1+s_2+s_5+s_6\ ,\quad s_{17} = s_{10}+s_{14}\ ,$$

$$s_{18} = u^{(k+1)}_{i-1,j}+u^{(k+1)}_{i,j-1}\ , \qquad s_{19} = u^{(k+1)}_{i+2,j-1}+u^{(k)}_{i+3,j}\ ,$$

$$s_{20} = u^{(k+1)}_{i-1,j+2}+u^{(k+1)}_{i,j+3}\ , \qquad s_{21} = u^{(k)}_{i+2,j+3}+u^{(k)}_{i+3,j+2}\ ,$$

$$s_{22} = u^{(k+1)}_{i-1,j-1}+u^{(k+1)}_{i-1,j+1}+u^{(k+1)}_{i+1,j-1},\quad s_{23} = u^{(k+1)}_{i+1,j-1}+u^{(k+1)}_{i+3,j-1}+u^{(k)}_{i+3,j+1}\ ,$$

$$s_{24} = u^{(k+1)}_{i-1,j+1}+u^{(k)}_{i-1,j+3}+u^{(k)}_{i+1,j+3},\quad s_{25} = u^{(k)}_{i+1,j+3}+u^{(k)}_{i+3,j+1}+u^{(k)}_{i+3,j+3}\ ,$$

$$s_{26} = 6624s_4 \quad \text{and} \quad s_{27} = 6624s_{11}\ . \tag{7.3.10}$$

Then, we have,

$$\tilde{u}^{(k+1)}_{i+1,j} = \frac{1}{99176}(27498u^{(k+1)}_{i+1,j-1}+12303s_1+7354s_2+s_{26}+3782u^{(k)}_{i+1,j+3}+3042s_5$$
$$+2601s_6+1551s_3+473s_7)\ ,$$

$$\tilde{u}^{(k+1)}_{i,j+1} = \frac{1}{99176}(27498u^{(k+1)}_{i-1,j+1}+12303s_8+7354s_9+s_{27}+3782u^{(k)}_{i+3,j+1}+3042s_{12}+$$
$$+2601s_{13}+1551s_{10}+473s_{14})\ ,$$

$$\tilde{u}^{(k+1)}_{i+2,j+1} = \frac{1}{99176}(27498u^{(k)}_{i+3,j+1}+12303s_{13}+7354s_{12}+s_{27}+3782u^{(k+1)}_{i-1,j+1}+$$
$$3042s_9+2601s_8+1551s_{14}+473s_{10})\ ,$$

$$\tilde{u}^{(k+1)}_{i+1,j+2} = \frac{1}{99176}(27498u^{(k)}_{i+1,j+3}+12303s_6+7354s_5+s_{26}+3782u^{(k+1)}_{i+1,j-1}+$$

$$3042\ s_2+2601s_1+1551s_7+473s_3)\ ,$$

$$\tilde{u}^{(k+1)}_{i+1,j+1} = \frac{1}{2156}(226s_{15}+144s_{16}+25s_{17})\ ,$$

$$\tilde{u}^{(k+1)}_{i,j} = 0.2(\tilde{u}^{(k+1)}_{i+1,j}+\tilde{u}^{(k+1)}_{i,j+1}+s_{18})+0.05(\tilde{u}^{(k+1)}_{i+1,j+1}+s_{22})\ ,$$

$$\tilde{u}^{(k+1)}_{i+2,j} = 0.2(\tilde{u}^{(k+1)}_{i+1,j}+\tilde{u}^{(k+1)}_{i+2,j+1}+s_{19})+0.05(\tilde{u}^{(k+1)}_{i+1,j+1}+s_{23})\ ,$$

$$\tilde{u}^{(k+1)}_{i,j+2} = 0.2(\tilde{u}^{(k+1)}_{i,j+1}+\tilde{u}^{(k+1)}_{i+1,j+2}+s_{20})+0.05(\tilde{u}^{(k+1)}_{i+1,j+1}+s_{24})\ ,$$

and

$$\tilde{u}^{(k+1)}_{i+2,j+2} = 0.2(\tilde{u}^{(k+1)}_{i+2,j+1}+\tilde{u}^{(k+1)}_{i+1,j+2}+s_{21})+0.05(\tilde{u}^{(k+1)}_{i+1,j+1}+s_{25})\ . \tag{7.3.11}$$

Hence, the S.O.R. iterative method is given by,

$$u^{(k+1)}_{i+\ell_1,j+\ell_2} = \omega(\tilde{u}^{(k+1)}_{i+\ell_1,j+\ell_2}-u^{(k)}_{i+\ell_1,j+\ell_2})+ u^{(k)}_{i+\ell_1,j+\ell_2}\ , \tag{7.3.12}$$

for $i,j=1,(3),m-2$ and $\ell_1,\ell_2=0,1,2$.

Assuming that the constants $\frac{1}{99176}$ and $\frac{1}{2156}$ are stored beforehand it can be seen that this method requires,

$$6m^2 \text{ multiplications} + 11m^2 \text{ additions}\ , \tag{7.3.13}$$

for the m^2 internal mesh points per iteration.

Now, if we combine the results for the experimental number of iterations given in Tables (7.3.1), (7.3.2) and (7.3.3) for the point, 4-point group and 9-point group S.O.R. respectively with the number of operations in each iteration required by each of the methods and given in (7.3.5), (7.3.9) and (7.3.13), we can obtain the total number of arithmetic operations or computational work required to obtain a solution and these are recorded in Table (7.3.4).

Method / h^{-1}	Point S.O.R.		4-Point Group S.O.R.		9-Point Group S.O.R.	
	Multiplications	Additions	M	A	M	A
13	$99m^2$	$297m^2$	$123.5m^2$	$194.75m^2$	$102m^2$	$187m^2$
25	$186m^2$	$558m^2$	$240.5m^2$	$379.25m^2$	$186m^2$	$341m^2$
37	$276m^2$	$828m^2$	$344.5m^2$	$543.25m^2$	$264m^2$	$484m^2$
49	$363m^2$	$1089m^2$	$461.5m^2$	$727.75m^2$	$354m^2$	$649m^2$

TABLE 7.3.4

The following table shows the results for the execution time of the three methods, which were obtained on the Prime Computer.

Method / h^{-1}	Point S.O.R.	4-Point Group S.O.R.	9-Point Group S.O.R.
13	109	128	115
25	816	978	818
37	2692	3146	2612
49	6292	7520	6259

TABLE 7.3.5: The execution time in centisec. for the optimum value of ω

This table and Table (7.3.4) which shows the total number of arithmetic operations required to solve Laplace's equation in the unit square by the point, 4-point group and 9-point group S.O.R. iterative methods, indicates that the 9-point group requires less computer time than the other two methods.

7.4 THE 4 AND 9 POINT GROUPS OF THE BIHARMONIC OPERATOR

As mentioned in Section (3.2), the biharmonic equation is a more complicated p.d.e. of elliptic type which when expressed as a model problem has the form,

$$\frac{\partial^4 U}{\partial x^4} + 2\frac{\partial^4 U}{\partial x^2 \partial y^2} + \frac{\partial^4 U}{\partial y^4} = 0 , \quad (x,y) \in R , \tag{7.4.1}$$

where R is a unit square $0 \leq x,y \leq 1$ and the boundary conditions usually take one of the two forms given in (3.2.17) and (3.2.18). Consider the region shown in Figure (4.2.1) with m^2 internal mesh points, the well known 13-point finite difference equation involving mesh values of u (u is an approximation to the exact solution U) at the nodes given in Figure (7.4.1), can be written explicitly as,

$$\begin{aligned}20u_t-8(u_{t-m}+u_{t-1}+u_{t+1}+u_{m+t})+2(u_{t-m-1}+u_{t-m+1}+u_{m+t-1}+u_{m+t+1})\\ +u_{t-2m}+u_{t-2}+u_{t+2}+u_{2m+t} = 0 ,\end{aligned} \tag{7.4.2}$$

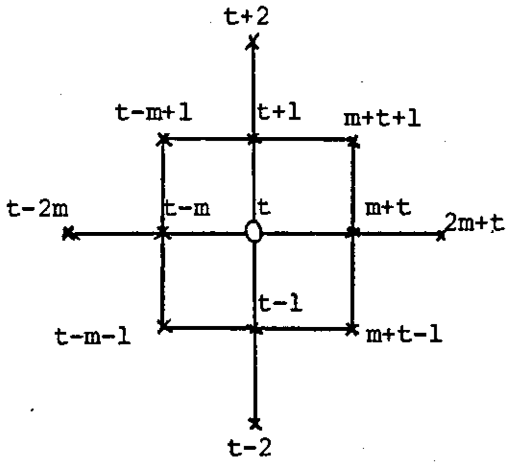


FIGURE 7.4.1

where $t=1,m^2$

In this section and in a similar manner to the 4 and 9 point groups of subsections (7.2.1) and (7.2.2) respectively we construct the 4 and 9 point groups for the biharmonic operator.

7.4.1 THE 4-POINT GROUP

We order the mesh points in groups of 4 as shown in Figure (7.4.2)

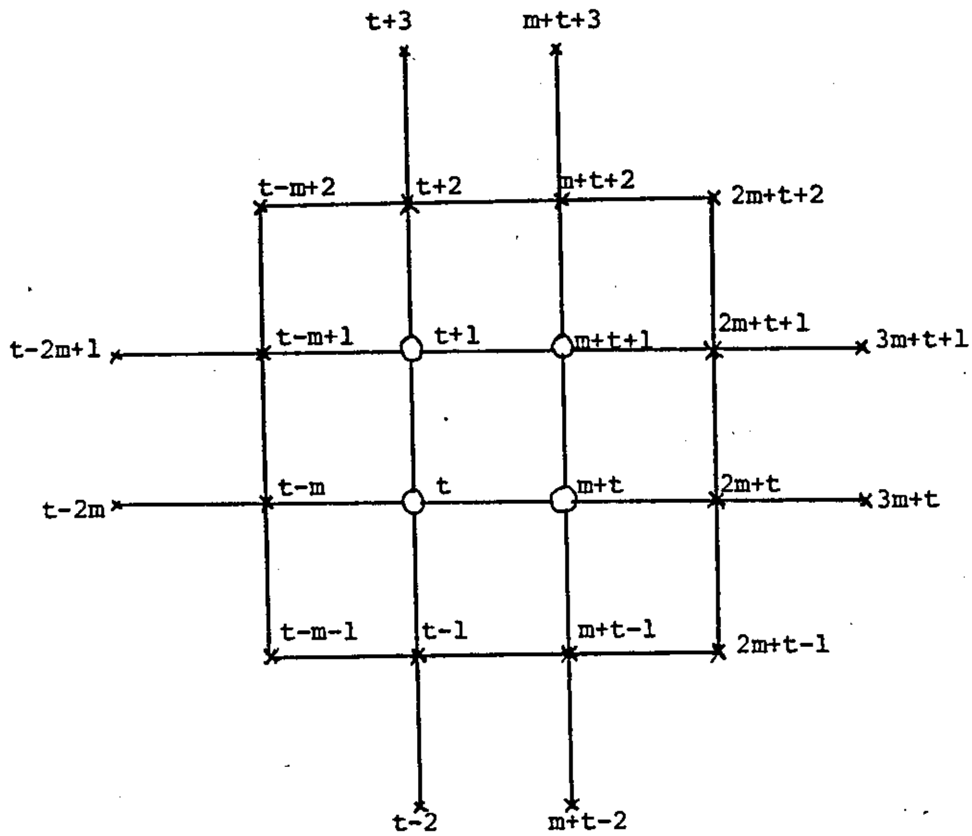


FIGURE 7.4.2

where $t=rm+1,(2),(r+1)m-1$ and $r=0,(2),m-2$ with m taken to be an even number. If equation (7.4.2) is applied to the nodes $t,t+1,m+t$ and $m+t+1$, we obtain the linear matrix equation $A_t\underline{u}_t=\underline{b}_t$, where the (4×4), matrix A_t and the two vectors $\underline{u}_t$ and $\underline{b}_t$ are given by,

$$A_t = \begin{bmatrix} 20 & -8 & -8 & 2 \\ -8 & 20 & 2 & -8 \\ -8 & 2 & 20 & -8 \\ 2 & -8 & -8 & 20 \end{bmatrix} , \quad \underline{u}_t = \begin{bmatrix} u_t \\ u_{t+1} \\ u_{m+t} \\ u_{m+t+1} \end{bmatrix} , \tag{7.4.3}$$

$$\underline{b}_t = \begin{bmatrix} 8(u_{t-m}+u_{t-1})-2(u_{t-m+1}+u_{t-m-1}+u_{m+t-1})-(u_{t-2m}+u_{t-2}+u_{2m+t}+u_{t-2}) \\ 8(u_{t-m+1}+u_{t+2})-2(u_{t-m}+u_{t-m+2}+u_{m+t+2})-(u_{t-2m+1}+u_{t+3}+u_{t-1}+u_{2m+t+1}) \\ 8(u_{m+t-1}+u_{2m+t})-2(u_{t-1}+u_{2m+t-1}+u_{2m+t+1})-(u_{m+t-2}+u_{3m+t}+u_{t-m}+u_{m+t+2}) \\ 8(u_{m+t+2}+u_{2m+t+1})-2(u_{2m+t}+u_{2m+t+2}+u_{t+2})-(u_{3m+t+1}+u_{m+t+3}+u_{t-m+1}+u_{m+t-1}) \end{bmatrix} .$$

Evidently, it can be seen that the matrix A_t can be easily inverted, hence, the group G.S. iterative method to solve the model problem (7.4.1) may be written explicitly, as,

$$\begin{aligned} u_t^{(k+1)} = \frac{1}{342} [& 172(u_{t-m}^{(k+1)}+u_{t-1}^{(k+1)})+56(u_{2m+t}^{(k)}+u_{t+2}^{(k)})-52u_{t-m-1}^{(k+1)}+37(u_{t-m+1}^{(k+1)} \\ & +u_{m+t-1}^{(k+1)})-26(u_{t-2m}^{(k+1)}+u_{t-2}^{(k+1)})-24(u_{t-m+2}^{(k)}+u_{2m+t-1}^{(k+1)})+20(u_{m+t+2}^{(k)} \\ & +u_{2m+t+1}^{(k)})-14u_{2m+t+2}^{(k)}-12(u_{t-2m+1}^{(k+1)}+u_{m+t-2}^{(k+1)}+u_{t+3}^{(k)}+u_{3m+t}^{(k)})-7(u_{m+t+3}^{(k)}+u_{3m+t+1}^{(k)}) . \end{aligned} \tag{7.4.4}$$

The other three points, i.e. $u_{t+1}^{(k+1)}$, $u_{m+t+1}^{(k+1)}$ and $u_{m+t}^{(k+1)}$ are found from completely analogous expressions obtaintable by rotating the computational molecule of Figure (7.4.3) in a clockwise manner.

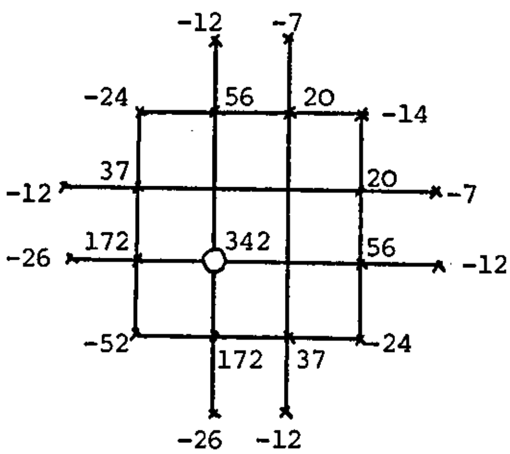


FIGURE 7.4.3

Obviously similar sets of equations can be written down for the 4 point explicit Jacobi, J.O.R. and S.O.R. methods.

7.4.2 THE 9-POINT GROUP

The mesh points of Figure (4.2.1) are now ordered in groups of nine as shown in Figure (7.4.4)

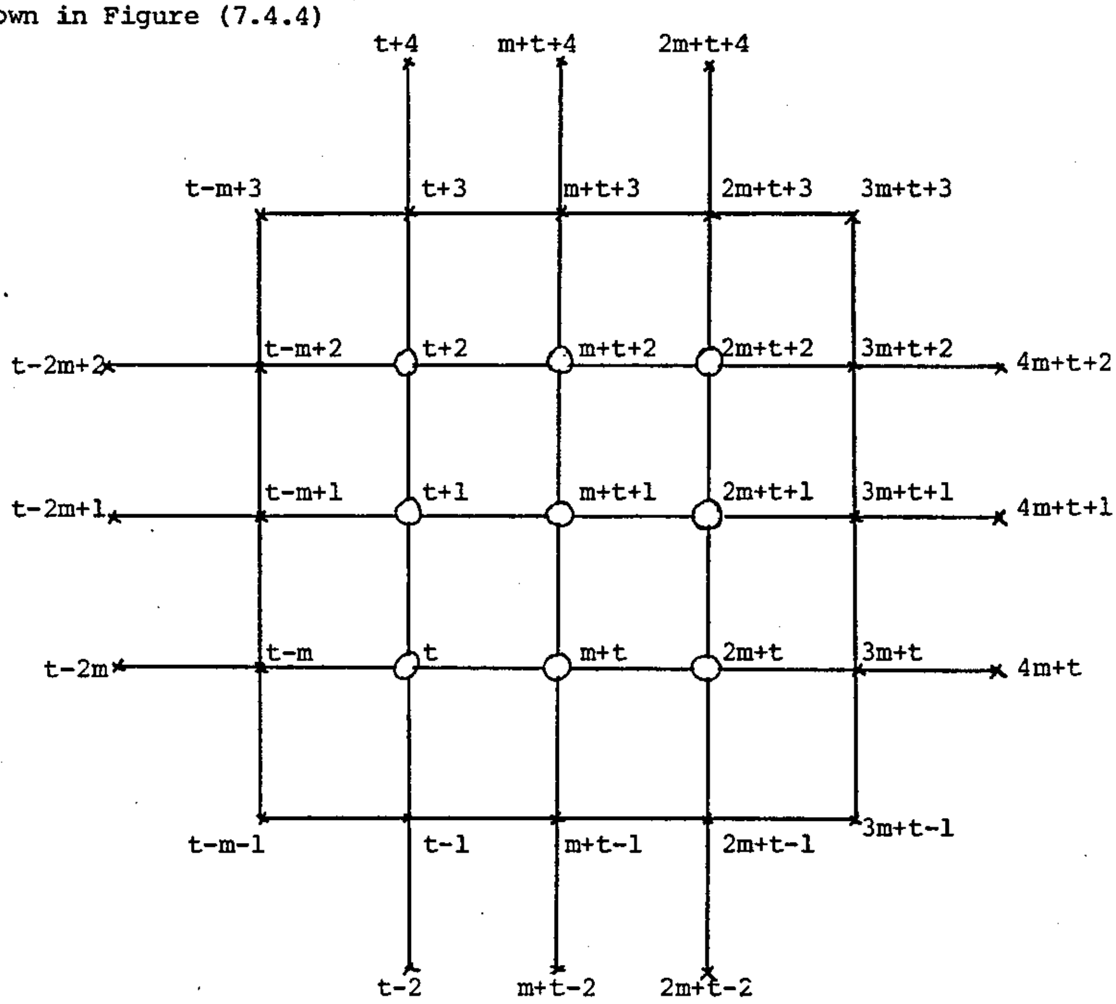


FIGURE 7.4.4

where $t=rm+1,(3),(r+1)m-2$ and $r=0,(3),m-3$, obviously m must be divisible by 3. Hence by applying equation (7.4.2) to the mesh points t, $t+1$, $t+2$, $m+t$, $m+t+1$, $m+t+2$, $2m+t$, $2m+t+1$ and $2m+t+2$, we obtain the linear equation $A_t\underline{u}_t=\underline{b}_t$ where the matrix A_t and the two vectors $\underline{u}_t$ and $\underline{b}_t$ are given by,

$$A_t = \begin{bmatrix} 20 & -8 & 1 & -8 & 2 & 0 & 1 & 0 & 0 \\ -8 & 20 & -8 & 2 & -8 & 2 & 0 & 1 & 0 \\ 1 & -8 & 20 & 0 & 2 & -8 & 0 & 0 & 1 \\ -8 & 2 & 0 & 20 & -8 & 1 & -8 & 2 & 0 \\ 2 & -8 & 2 & -8 & 20 & -8 & 2 & -8 & 2 \\ 0 & 2 & -8 & 1 & -8 & 20 & 0 & 2 & -8 \\ 1 & 0 & 0 & -8 & 2 & 0 & 20 & -8 & 1 \\ 0 & 1 & 0 & 2 & -8 & 2 & -8 & 20 & -8 \\ 0 & 0 & 1 & 0 & 2 & -8 & 1 & -8 & 20 \end{bmatrix}_{(9\times 9)}, \quad \underline{u}_t = \begin{bmatrix} u_t \\ u_{t+1} \\ u_{t+2} \\ u_{m+t} \\ u_{m+t+1} \\ u_{m+t+2} \\ u_{2m+t} \\ u_{2m+t+1} \\ u_{2m+t+2} \end{bmatrix} \tag{7.4.5}$$

$$\underline{b}_t = \begin{bmatrix} 8(u_{t-m}+u_{t-1})-2(u_{t-m-1}+u_{t-m+1}+u_{m+t-1})-(u_{t-2m}+u_{t-2}) \\ 8u_{t-m+1}-2(u_{t-m}+u_{t-m+2})-(u_{t-1}+u_{t+3}+u_{t-2m+1}) \\ 8(u_{t-m+2}+u_{t+3})-2(u_{t-m+1}+u_{t-m+3}+u_{m+t+3})-(u_{t-2m+2}+u_{t+4}) \\ 8u_{m+t-1}-2(u_{t-1}+u_{2m+t-1})-(u_{t-m}+u_{m+t-2}+u_{3m+t}) \\ -(u_{t-m+1}+u_{m+t-1}+u_{3m+t+1}+u_{m+t+3}) \\ 8u_{m+t+3}-2(u_{t+3}+u_{2m+t+3})-(u_{t-m+2}+u_{m+t+4}+u_{3m+t+2}) \\ 8(u_{2m+t-1}+u_{3m+t})-2(u_{m+t-1}+u_{3m+t-1}+u_{3m+t+1})-(u_{2m+t-2}+u_{4m+t}) \\ 8u_{3m+t+1}-2(u_{3m+t}+u_{3m+t+2})-(u_{2m+t-1}+u_{2m+t+3}+u_{4m+t+1}) \\ 8(u_{2m+t+3}+u_{3m+t+2})-2(u_{3m+t+1}+u_{3m+t+3}+u_{m+t+3})-(u_{4m+t+2}+u_{2m+t+4}) \end{bmatrix}.$$

By using REDUCE, the matrix A_t can be inverted and the explicit group Jacobi iterative method to solve the model problem (7.4.1) may be written as,

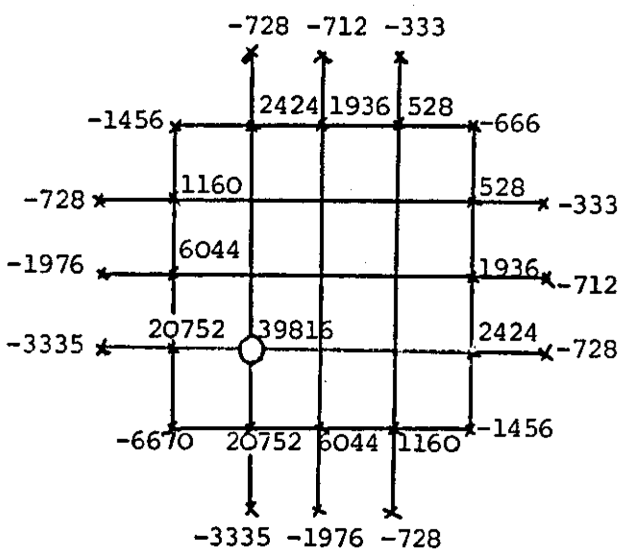
$$\begin{aligned} u_t^{(k+1)} = \frac{1}{39816} [& 20752(u_{t-m}^{(k)}+u_{t-1}^{(k)})-6670u_{t-m-1}^{(k)}+6044(u_{t-m+1}^{(k)}+u_{m+t-1}^{(k)}) \\ & -3335(u_{t-2m}^{(k)}+u_{t-2}^{(k)})+2424(u_{t+3}^{(k)}+u_{3m+t}^{(k)})-1976(u_{t-2m+1}^{(k)}+u_{m+t-2}^{(k)})+ \\ & +1936(u_{m+t+3}^{(k)}+u_{3m+t+1}^{(k)})-1456(u_{t-m+3}^{(k)}+u_{3m+t-1}^{(k)})+1160(u_{t-m+2}^{(k)}+u_{2m+t-1}^{(k)}) \\ & -728(u_{t-2m+2}^{(k)}+u_{2m+t-2}^{(k)}+u_{t+4}^{(k)}+u_{4m+t}^{(k)})-712(u_{m+t+4}^{(k)}+u_{4m+t+1}^{(k)})-666u_{3m+t+3}^{(k)} \\ & +528(u_{2m+t+3}^{(k)}+u_{3m+t+2}^{(k)})-333(u_{2m+t+4}^{(k)}+u_{4m+t+2}^{(k)}) , \end{aligned} \tag{7.4.6a}$$

$$u_{t+1}^{(k+1)} = \frac{1}{84609}\,[51434u_{t-m+1}^{(k)}+17288(u_{t-1}^{(k)}+u_{t+3}^{(k)})+11442(u_{t-m}^{(k)}+u_{t-m+2}^{(k)})$$
$$+10038(u_{m+t-1}^{(k)}+u_{m+t+3}^{(k)})-9332u_{t-2m+1}^{(k)}+8458u_{3m+t+1}^{(k)}-8398(u_{t-m-1}^{(k)}+$$
$$u_{t-m+3}^{(k)})-4199(u_{t-2m}^{(k)}+u_{t-2m+2}^{(k)}+u_{t-2}^{(k)}+u_{t+4}^{(k)})-3486(u_{m+t-2}^{(k)}+u_{m+t+4}^{(k)})$$
$$+3384(u_{3m+t}^{(k)}+u_{3m+t+2}^{(k)})-3026(u_{3m+t-1}^{(k)}+u_{3m+t+3}^{(k)})-2617u_{4m+t+1}^{(k)}+$$
$$+2515(u_{2m+t-1}^{(k)}+u_{2m+t+3}^{(k)})-1513(u_{2m+t-2}^{(k)}+u_{2m+t+4}^{(k)}+u_{4m+t}^{(k)}+u_{4m+t+2}^{(k)}). \tag{7.4.6b}$$

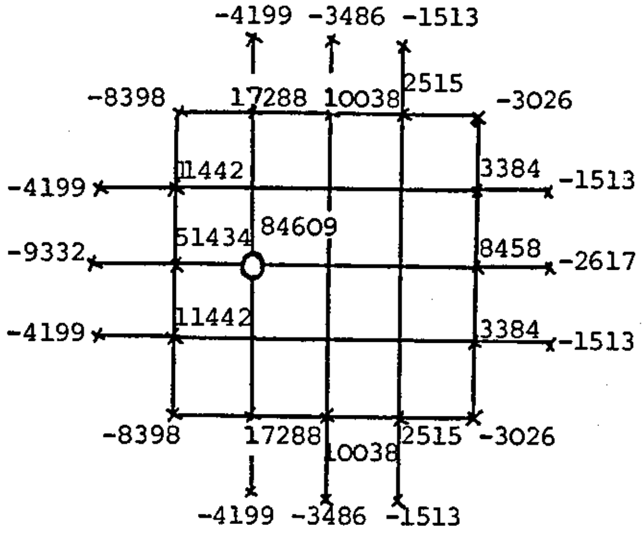
The points $u_{t+2},u_{2m+t+2},u_{2m+t}$ are found by a similar formula to (7.4.6a) and the points $u_{m+t+2},u_{2m+t+1},u_{m+t}$ are found by a similar formula to (7.4.6b) and the two sets of points can be obtained by rotating the computational molecule of Figures (7.4.5a) and (7.4.5b) in a clockwise direction respectively. Finally,

$$u_{m+t+1}^{(k+1)} = \frac{1}{316}\,[91(u_{t-m+1}^{(k)}+u_{m+t-1}^{(k)}+u_{3m+t+1}^{(k)}+u_{m+t+3}^{(k)})+32(u_{t-m}+u_{t-m+2}+u_{t-1}$$
$$+u_{t+3}^{(k)}+u_{2m+t-1}^{(k)}+u_{2m+t+3}^{(k)}+u_{3m+t}^{(k)}+u_{3m+t+2}^{(k)})-26(u_{t-m-1}^{(k)}+u_{t-m+3}^{(k)}+u_{3m+t-1}^{(k)}+$$
$$u_{3m+t+3}^{(k)})-24(u_{t-2m+1}^{(k)}+u_{m+t-2}^{(k)}+u_{4m+t+1}^{(k)}+u_{m+t+4}^{(k)})-13(u_{t-2m}^{(k)}+u_{t-2m+2}^{(k)}+$$
$$u_{t-2}^{(k)}+u_{t+4}^{(k)}+u_{2m+t-2}^{(k)}+u_{2m+t+4}^{(k)}+u_{4m+t}^{(k)}+u_{4m+t+2}^{(k)})\;, \tag{7.4.6c}$$

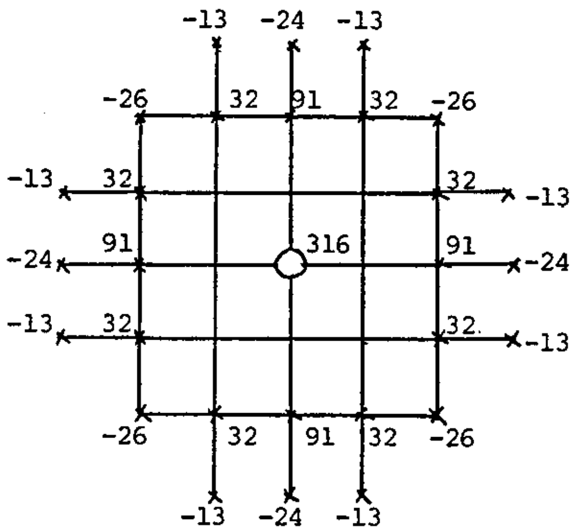
with the computational molecule given in Figure (7.4.5c).



(a)



(b)



(c)

FIGURE 7.4.5

It should be noted that these computational molecules are for the standard points in the main part of the region. Within the vicinity of the boundary then the technique mentioned in Chapter 3 for the boundary condition has to be applied which in turn will affect the matrix A_t and the vector $\underline{b}_t$ and a new computational molecule will need to be evaluated.

CHAPTER 8

CONCLUSIONS AND FINAL REMARKS

The main study in this thesis was the derivation of a new group technique for the solution by successive overrelaxation methods of boundary value problems in one, two and three space dimensions using finite difference approximations. A detailed analysis of the computational efficiency of both new and existing algorithms has also been made.

The study in Chapter 4 of the numerical solution of Laplace's equation by the novel approach of using a fixed size group strategy by the S.O.R. iterative method indicates the fact that these group methods require less storage and are simpler to program than the classic block methods (one and two-line iterative methods) and as indicated in Chapter 4, the 9 point group method appears to be the most efficient. In the 3-space dimensions, this group technique (the 8-point group S.O.R. iterative method) is also found to be superior to the point S.O.R. iterative method for solving Laplace's equation.

Further research is needed to implement the work on parallel computers, in which the new 7,9,11,13,15,17 or 21 point computational molecules (Figures(4.2.5),(4.2.7), (4.2.9),(4.2.11),(4.2.13),(4.2.15) and (4.2.17)) are used instead of the equivalent 5 point computational molecule (Figure (3.2.1)). Also, for the 3-space dimensions case, instead of using the 7-point computational molecule (Figure (3.2.7)), the equivalent 25 point computational molecule (Figure (4.5.4)) can be used to implement the work in parallel.

In Chapter 5, the numerical results obtained for the 4 point-2 line iterative method are encouraging and confirm its superior convergence properties, hence further research is required to extend this technique (i.e., the implicit block explicit overrelaxation scheme) to the 9 point group in which the 9 point explicit equation (4.2.31) can be grouped together again in an implicit iteration method to give the new 9 point-3 line iterative method. Also, further theoretical investigation is required to evaluate a theoretical value of the spectral radius of the iteration matrix and hence determine the rate of convergence theoretically. Also, further work is needed to prove the two hypotheses given in Section (5.5).

In Chapter 6, some explicit group iterative methods were applied to solve the two-point boundary value problem (6.2.1)-(6.2.2) in which the 3-point group iterative approach is found to be more attractive than the other groups since it is easier to program and simpler to invert when compared with the 8 and 12 point group methods. It also offers some gains in computational efficiency over the 2,4 and 6 point groups. In Section (6.3) the group technique is used to solve non-linear boundary value problems iteratively and in Section (6.4) the A.G.E. method is used to solve linear and non-linear problems. From the numerical results obtained it shows that the A.G.E. method requires more computer time than the equivalent 2-point group N.L.O.R. method to solve three different non-linear problems. But, it is more efficient than the S.O.R. iterative approach to solve the linear boundary value problem.

Further work is required to include more than one iteration parameter (r) in the A.G.E. method for which an improvement in the rate of convergence is predicted.

The study in Chapter 7 of the new 4 and 9 point group methods to the

9 point finite difference equation indicates that the 9 point group S.O.R. method is more efficient and requires less computational time than the point and 4 point group S.O.R. methods. Again, further work is needed to implement the work on parallel computers.

Finally more extensive work is required on the 4 and 9-point group methods of the biharmonic operator to solve the elliptic boundary-value problem numerically, where instead of using the usual 13 point computational molecule (Figure (3.2.6)) we use the equivalent new 21 point (or 29) point computational molecules (Figures (7.4.3) and (7.4.5)).

REFERENCES

AMES, W.F. (1965), *Non-Linear Partial Differential Equations in Engineering*, Academic Press, New York.

AMES, W.F. (1977), *Numerical Methods for Partial Differential Equations*, 2nd Ed., Academic Press, New York.

ARMS, R.J., GATES, L.D. and ZONDEK, B. (1956), *A Method of Block Iteration*, J.Soc.Indust.Appl.Math., Vol. 4, pp.220-229.

BATHE, K.J. and WILSON, E.L. (1976), *Numerical Methods in Finite Element Analysis*, Prentice-Hall, Englewood Cliffs, N.J.

BELL, W.W. (1975), *Matrices for Scientists and Engineers*, Van Nostrand Reinhold Company, New York.

BENSON, A. (1969), *The Numerical Solution of Partial Differential Equations by Finite Difference Methods*, Ph.D. Thesis, University of Sheffield.

BENSON, A. and EVANS, D.J. (1972), *The Successive Peripheral Block Over-relaxation*, J. Inst. Maths. Applics., Vol. 9, pp.68-79.

BIRKHOFF, G., VARGA, R.S. and YOUNG, D.M. (1962), *Alternating Direction Implicit Methods*, Advances in Computers, Vol. 3, Academic Press, New York.

BROWN, W.S. and HEARN, A.C. (1978), *Application of Symbolic Algebraic Computation*, Proceedings, Computational Atomic and Molecular Physics Conference, Nottingham, England.

BROYDEN, C.G. (1975), *Basic Matrices*, The Macmillan Press Ltd., London.

CARRE, B.A. (1961), *The Determination of the Optimum Acceleration Factor for Successive Overrelaxation*, The Computer Journal, Vol. 4, pp.73-78.

CLOUTMAN, L.D. and FULLERTON, L.W. (1977), *Some Applications of Automated Heuristic Stability Analysis*, Report LA-6885-MS, Los Alamos Scientific Laboratory, Los Alamos, N.M.

COHEN, A.M. (1973), *Numerical Analysis*, McGraw-Hill Book Company, London.

CONTE, S.D. and de BOOR, C., (1972), *Elementary Numerical Analysis, An Algorithmic Approach*, 2nd ed. McGraw-Hill Kogakusha, Ltd. Tokyo.

CUTHILL, E.G. and VARGA, R.S. (1959), *A Method of Normalized Block Iteration*, Jour.Assoc.Comp.Mach., Vol. 6, pp.236-244.

DOUGLAS, J. and RACHFORD, H.H. (1956), *On the Numerical Solution of Heat Conduction Problems in Two and Three Space Variables*, Trans. Amer. Math.Soc., Vol. 82, pp.421-439.

EVANS, D.J. (1972), *A New Iterative Procedure for the Solution of Sparse Systems of Linear Difference Equations*, in "Sparse Matrices and Their Applications", eds. Rose, D.J. and Willoughby, R.A., Plenum Press, New York, pp.89-100.

EVANS, D.J. and BENSON, A. (1976), *The Solution of Elliptic Partial Differential Equations by Peripheral Over-relaxation Techniques*, Comp.Maths. in Physics, pp.387-405.

EVANS, D.J. and BIGGINS, M.J., (1982), *The Solution of Elliptic Partial Differential Equations by a New Block Over-relaxation Technique*, Int.Jour.Comp.Math., Vol. 10, pp.269-282.

FADDEEV, D.K. and FADDEEVA, V.N., (1963), *Computational Methods of Linear Algebra*, W.H. Freeman and Company, San Francisco.

FENNER, R.T. (1975), *Finite Element Methods for Engineers*, The Macmillan Press Ltd., London.

FORSYTHE, G.E. and MOLER, C.B. (1967), *Computer Solution of Linear Algebraic Systems*, Prentice-Hall, New Jersey.

FORSYTHE, G.E. and WASOW, W.R. (1960), *Finite Difference Methods for Partial Differential Equations*, John Wiley and Sons, New York.

FOX, L. (1950), *The Numerical Solution of Elliptic Differential Equations When Boundary Conditions Involve a Derivative*, Phil.Trans.Roy.Soc. A 242, pp.345-378.

FOX, L. (1964), *An Introduction to Numerical Linear Algebra*, Clarendon Press, Oxford.

FOX, L. (1977), *Finite-Difference Methods for Elliptic Boundary-Value Problems*, in The State of the Art in Numerical Analysis, ed. Jacobs, D.A.H., Academic Press, London, pp.799-881.

FRANKEL, S.P. (1950), *Convergence Rates of Iterative Treatments of Partial Differential Equations*, Math. Tables Aids.Comp., Vol. 4, pp.65-75.

FROBERG, C.E. (1979), *Introduction to Numerical Analysis*, 2nd ed., Addison-Wesley Publishing Company, Reading, Mass.

GANTMAKHER, F.R. (1959), *Applications of the Theory of Matrices*, translated and revised by J.L. Brenner, Interscience Publishers, New York.

GERALD, C.F., (1978), *Applied Numerical Analysis*, 2nd ed., Addison-Wesley Publishing Company, Reading, Mass.

GLADWELL, I. and WAIT, R. (1979), *A Survey of Numerical Methods for Partial Differential Equations*, Clarendon Press, Oxford.

GOULT, R.J., HOSKINS, R.F., MILNER, J.A. and PRATT, M.J. (1974), *Computational Methods in Linear Algebra*, Stanley Thornes (Publishers) Ltd., London.

GOURLAY, A.R. and WATSON, G.A. (1973), *Computational Methods for Matrix Eigenproblems*, John Wiley, London.

GRAHAM, A. (1979), *Matrix Theory and Applications for Engineers and Mathematicians*, John Wiley, New York.

GREGORY, R.T. and KARNEY, D.L. (1969), *A Collection of Matrices for Testing Computational Algorithms*, Wiley Interscience, New York.

HABETLER, G.J. and WACHSPRESS, E.L. (1961), *Symmetric Successive Over-relaxation in Solving Diffusion Difference Equations*, Maths. of Comp. Vol. 15, pp.356-362.

HAGEMAN, L.A. and KELLOGG, R.B. (1968), *Estimating Optimum Over-relaxation Parameters*, Maths. of Comp., Vol. 22, pp.60-68.

HEARN, A.C. (1983), *REDUCE User's Manual*, Version 3.0, The Rand Corporation, Santa Monica.

HILDEBRAND, F.B. (1974), *Introduction to Numerical Analysis*, 2nd ed. McGraw-Hill, New Delhi.

JAIN, M.K. (1979), *Numerical Solution of Differential Equations*, Wiley Eastern Limited, New Delhi.

JENNINGS, A. (1977), *Matrix Computation for Engineers and Scientists*, John Wiley, Chichester.

JENSON, J. and NIORDSON, F. (1977), *Symbolic and Algebraic Manipulation Languages and Their Application in Mechanics*, in Structrual Mechanics Software Series, Vol. 1, pp.541-576, University Press of Virginia, Charlottesville.

JOHNSON, L.W. and RIESS, R.D. (1982), *Numerical Analysis*, 2nd ed., Addison-Wesley Publishing Company, Reading.

KAHAN, W. (1958), *Gauss-Seidel Methods of Solving Large Systems of Linear Equations*, Doctoral Thesis, University of Toronto, Toronto, Canada.

KERSTEN, R.D. (1969), *Engineering Differential Systems*, McGraw-Hill Book Company, New York.

KUNZ, K.S. (1957), *Numerical Analysis*, McGraw-Hill Book Company Inc., New York.

LIEBERSTEIN, H.M. (1959), *Over-relaxation for Non-linear Elliptic Partial Differential Equations*, University of Wisconsin Math.Res. Center (Madison, Wis.) Tech. Sum. Rept. No. MRC-TSR-80.

LEIBERSTEIN, H.M. (1968), *A Course in Numerical Analysis*, Harper and Row.

MARKOFF, W. (1916), *Über Polynome, die in einem gegebenan Intervalle möglichst wenig von Null abweichen*, Math. Ann., Vol. 77, pp.213-258 (Translated by J. Grossmann from Russian original of 1892).

MEYER, C.D. and PLEMMONS, R.J. (1977), *Convergent Powers of a Matrix with Applications to Iterative Methods for Singular Linear Systems*, SIAM J. Numer.Anal., Vol. 14, pp.699-705.

MILLER, K.S. (1959), *Partial Differential Equations in Engineering Problems*, Prentice-Hall, New Jersey.

MILNE, W.E. (1970), *Numerical Solution of Differential Equations*, 2nd ed., Peter Smith Publisher, Gloucester, Mass.

MITCHELL, A.R. (1976), *Computational Methods in Partial Differential Equations*, John Wiley, London.

MITCHELL, A.R. and GRIFFITHS, D.F. (1980), *The Finite Difference Method in Partial Differential Equations*, John-Wiley, New York.

MITCHELL, A.R. and WAIT, R. (1977), *The Finite Element Method in Partial Differential Equations*, John Wiley, New York.

ORTEGA, J.M. and RHEINBOLDT, W.C. (1970), *Iterative Solution of Non-Linear Equations of Several Variables*, Academic Press, New York.

PARTER, S.V. (1961), *Multi-line Iterative Methods for Elliptic Difference Equations and Fundamental Frequencies*, Numer.Math. Vol. 3, pp.305-319.

PARTER, S.V. (1981), *Block Iterative Methods*, in Elliptic Problem Solvers, ed. Schultz, M.H., Academic Press, pp.375-382.

PARTER, S.V. and STEUERWALT, M. (1982), *Block Iterative Methods for Elliptic and Parabolic Differential Equations*, SIAM.J.Numer. Anal., Vol. 19, pp.1173-1195.

PEACEMAN, D.W. and RACHFORD, H.H. (1955), *The Numerical Solution of Parabolic and Elliptic Differential Equations*, J.Soc.Indust.Appl. Math., Vol. 3, pp.28-41.

RALSTON, A. (1965), *A First Course in Numerical Analysis*, McGraw-Hill Book Company, New York.

RICHARDSON, L.F. (1910), *The Approximate Arithmetical Solution by Finite Differences of Physical Problems Involving Differential Equations with Application to the Stress in a Masonry Dam*, Philos.Trans.Roy. Soc. London, Ser. A210, pp.307-357.

SHORTLEY, G.H. and WELLER, R. (1938), *The Numerical Solution of Laplace's Equations*, J.Appl.Phys. Vol. 9, pp.334-348.

SMITH, G.D. (1978), *Numerical Solution of Partial Differential Equations: Finite Difference Methods*, 2nd ed., Oxford University Press, Oxford.

SOJOODI-HAGHIGHI, R. (1981), *The Numerical Solution of Elliptic Partial Differential Equations by Novel Block Iterative Methods*, Ph.D. Thesis, Loughborough University of Technology.

STEIN, P. and ROSENBERG, R. (1948), *On the Solution of Linear Simultaneous Equations by Iteration*, J. London Math.Soc., Vol. 23, pp.111-118.

STIEFEL, E.G. (1958), *Kernel Polynomials in Linear Algebra and Their Numerical Applications*, Nat.Bur. Standards Appl.Math. Ser. 49, pp.1-22.

STRANG, G. and FIX, G.J. (1973), *An Analysis of the Finite Element Method*, Prentice-Hall, Englewood Cliffs, N.J.

TODD, J. (1950), *The Condition of a Certain Matrix*, Proc. Cambridge Philos. Soc., Vol. 46, pp.116-118.

TODD, J. (1962), *Survey of Numerical Analysis*, McGraw-Hill Book Company, New York.

VAN DE VOOREN, A.I. and VLIEGENTHART, A.C. (1967), *On the 9-Point Difference Formula for Laplace's Equation*, J. Eng.Math., Vol. 1, pp.187-202.

VARGA, R.S. (1962), *Matrix Iterative Analysis*, Prentice-Hall, Englewood Cliffs, N.J.

VICHNEVETSKY, R. (1981), *Computer Methods for Partial Differential Equations*, Vol. 1, Prentice-Hall, Englewood Cliffs, N.J.

WACHSPRESS, E.L. (1957), *CURE: A Generalized Two-Space-Dimension Multi-group Coding for the IBM-704*, Report KAPL-1724, Knolls Atomic Power Lab., Schenectady, New York.

WACHSPRESS, E.L. (1966), *Iterative Solution of Elliptic Systems and Applications to the Neutron Diffusion Equations of Reactor Physics*, Prentice-Hall, Englewood Cliffs, N.J.

WILKINSON, J.H. (1965), *The Algebraic Eigenvalue Problem*, Oxford University Press, Oxford.

YOUNG, D.M. (1954), *Iterative Methods for Solving Partial Differential Equations of Elliptic Type*, Trans. Amer.Math.Soc., Vol. 76, pp.92-111.

YOUNG, D.M. (1971), *Iterative Solution of Large Linear Systems*, Academic Press, New York.

YOUNG, D.M. and FRANK, T.G. (1963), *A Survey of Computer Methods for Solving Elliptic and Parabolic Partial Differential Equations*, Rome Centre Int. de Calcul., Vol. 2, No. 1.

YOUNG, D.M. and WARLICK, C.H. (1953), *On the Use of Richardson's Method for the Numerical Solution of Laplace's Equation on the ORDVAC*, Ballistic Research Labs., Memorandum Report No. 707, Aberdeen Proving Ground, Maryland.

APPENDIX A

SYMBOLIC INVERSION OF THE (3 X 4) POINTS BLOCK MATRIX (4.2.32 b)

RI(i,j) , (i=1,12 & j=1,12) , are the elements of the matrix $(\text{RØ})^{-1}$,(see page 159) , and D = det(RØ) . $A1 = \alpha_1$, $A2 = \alpha_2$, $A3 = \alpha_3$,and $A4 = \alpha_4$.

$$
\begin{aligned}
D:= {} & A3^{6}*A1^{6} -12*A3^{5}*A2*A1^{5}*A4-9*A3^{5}*A1^{5} +44*A3^{4}*A2^{2}*A1^{4}*A4^{2} +24* \\
& A3^{4}*A2*A1^{4}*A4+30*A3^{4}*A1^{4} -48*A3^{3}*A2^{3}*A1^{3}*A4^{3} -108*A3^{3}*A2^{2}*A1^{3} \\
& *A4^{2} +36*A3^{3}*A2*A1^{3}*A4 - 45*A3^{3}*A1^{3} + 16*A3^{2}*A2^{4}*A1^{2}*A4^{4} +112* \\
& A3^{2}*A2^{3}*A1^{2}*A4^{3} -4*A3^{2}*A2^{2}*A1^{2}*A4^{2} -56*A3^{2}*A2*A1^{2}*A4 + 30*A3^{2}* \\
& A1^{2} -48*A3*A2^{4}*A1*A4^{4} +48*A3*A2^{3}*A1*A4^{3} - 48*A3*A2^{2}*A1*A4^{2} +36 \\
& *A3*A2*A1*A4-9*A3*A1+16*A2^{4}*A4^{4} - 32*A2^{3}*A4^{3} + 24*A2^{2}*A4^{2} -8 \\
& *A2*A4 + 1
\end{aligned}
$$

$$
\begin{aligned}
RI(1,1):=(& -2*A3^{5}*A1^{5} + 9*A3^{4}*A2*A1^{4}*A4 + 13*A3^{4}*A1^{4} - 38*A3^{3}*A2^{2}* \\
& A1^{3}*A4^{2} + 17*A3^{3}*A2*A1^{3}*A4 - 28*A3^{3}*A1^{3} + 44*A3^{2}*A2^{3}*A1^{2}* \\
& A4^{3} - 14*A3^{2}*A2^{2}*A1^{2}*A4^{2} - 30*A3^{2}*A2*A1^{2}*A4 + 23*A3^{2}*A1^{2} \\
& - 16*A3*A2^{4}*A1*A4^{4} + 20*A3*A2^{3}*A1*A4^{3} - 24*A3*A2^{2}*A1*A4^{2} \\
& + 25*A3*A2*A1*A4 - 8*A3*A1 + 8*A2^{4}*A4^{4} - 20*A2^{3}*A4^{3} + 18* \\
& A2^{2}*A4^{2} - 7*A2*A4 + 1)/ D
\end{aligned}
$$

$$
\begin{aligned}
RI(1,2):=(A3*(& A3^{5}*A1^{5} - 10*A3^{4}*A2*A1^{4}*A4 - 7*A3^{4}*A1^{4} +30*A3^{3}*A2^{2}* \\
& A1^{3}*A4^{2} + 12*A3^{3}*A2*A1^{3}*A4 + 17*A3^{3}*A1^{3} - 28*A3^{2}*A2^{3}* \\
& A1^{2}*A4^{3} - 32*A3^{2}*A2^{2}*A1^{2}*A4^{2} + 9*A3^{2}*A2*A1^{2}*A4 - 17* \\
& A3^{2}*A1^{2} + 8*A3*A2^{4}*A1*A4^{4} + 20*A3*A2^{3}*A1*A4^{3} + 2*A3* \\
& A2^{2}*A1*A4^{2} - 13*A3*A2*A1*A4 + 7*A3*A1 - 8*A2^{4}*A4^{4} +12 \\
& *A2^{3}*A4^{3} - 10*A2^{2}*A4^{2} + 5*A2*A4 - 1))/ D
\end{aligned}
$$

$$
\begin{aligned}
RI(1,3):=(A3^{2}*(& A3^{4}*A1^{4} -3*A3^{3}*A2*A1^{3}*A4 - 6*A3^{3}*A1^{3} + 16*A3^{2}*A2^{2}* \\
& A1^{2}*A4^{2} - 13*A3^{2}*A2*A1^{2}*A4 + 11*A3^{2}*A1^{2} - 24*A3*A2^{3}* \\
& A1*A4^{3} + 30*A3*A2^{2}*A1*A4^{2} + 3*A3*A2*A1*A4 - 6*A3*A1
\end{aligned}
$$

```
                4    4         3    3          2    2
          + 8*A2 *A4  - 16*A2 *A4  + 6*A2 *A4  - 2*A2*A4+1))
       / D

          3     4    4         3         3           3   3         2   2
RI(1,4):=(A3 *(-A3 *A1  + 9*A3 *A2*A1 *A4 + 6*A3 *A1 -24*A3 *A2
                2    2        2         2             2    2             3
           *A1 *A4  - 9*A3 *A2*A1 *A4 - 11*A3 *A1  + 24*A3*A2 *
              3          2          2
           A1*A4  - 6*A3*A2 *A1*A4  + 15*A3*A2*A1*A4 + 6*A3*A1
                4    4         3    3          2    2
            - 8*A2 *A4  + 16*A2 *A4  - 14*A2 *A4  - 2*A2*A4 -1)
         )/ D

                    5    5         4          4           4    4         3
RI(1,5):=(A2*(-2*A3 *A1  + 14*A3 *A2*A1 *A4 + 3*A3 *A1  - 20*A3 *
            2   3   2          3          3           3    3          2
          A2 *A1 *A4  - 38*A3 *A2*A1 *A4 + 10*A3 *A1  + 8*A3 *
            3   2   3          2   2   2   2          2      2
          A2 *A1 *A4  + 48*A3 *A2 *A1 *A4  + 8*A3 *A2*A1 *A4 -
                2    2          3       3          2       2
          13*A3 *A1  - 24*A3*A2 *A1*A4  + 16*A3*A2 *A1*A4 -14*
                                           3    3          2    2
          A3*A2*A1*A4 + 6*A3*A1 + 8*A2 *A4  - 12*A2 *A4  +6*A2*
          A4 - 1))/ D

                         4    4       3          3           3    3         2
RI(1,6):=(2*A3*A2*(2*A3 *A1 -4*A3 *A2*A1 *A4 - 8*A3 *A1  + 4*A3 *
           2   2   2         2          2          2    2             2
         A2 *A1 *A4  + 8*A3 *A2*A1 *A4 + 9*A3 *A1  - 12*A3*A2 *
              2                                 2    2
         A1*A4  + 8*A3*A2*A1*A4 - 5*A3*A1 + 4*A2 *A4  - 4*A2*A4
          + 1))/ D

           2        4    4      3         3          2    2    2    2
RI(1,7):=(A3 *A2*(A3 *A1 -6*A3 *A2*A1 *A4 + 4*A3 *A2 *A1 *A4 +22
             2       2              2    2             2       2
         *A3 *A2*A1 *A4 - 11*A3 *A1  - 12*A3*A2 *A1*A4  - 18*A3*
                                              2    2
         A2*A1*A4 + 12*A3*A1 + 4*A2 *A4  + 4*A2*A4 - 3))/ D

              3          3    3         2         2            2    2
RI(1,8):=(2*A3 *A2*(-3*A3 *A1  + 4*A3 *A2*A1 *A4 + 11*A3 *A1 -12
         *A3*A2*A1*A4 - 9*A3*A1 + 4*A2*A4 + 2))/ D

           2          4    4        3          3           3    3         2
RI(1,9):=(A2 *(-15*A3 *A1 +50*A3 *A2*A1 *A4 + 5*A3 *A1  - 52*A3
                2   2   2         2          2          2    2
           *A2 *A1 *A4  - 10*A3 *A2*A1 *A4 + 6*A3 *A1  + 16*A3*
             3          3          2          2                  3    3
           A2 *A1*A4  + 4*A3*A2 *A1*A4  - 3*A3*A1 - 8*A2 *A4 +
                  2    2
              12*A2 *A4  - 6*A2*A4 + 1))/ D

                    2       4    4         3          3           3    3
RI(1,10):=(A3*A2 *(2*A3 *A1  - 14*A3 *A2*A1 *A4 + 12*A3 *A1 +20*
```

```
  2   2   2   2         2       2            2   2
A3 *A2 *A1 *A4  - 12*A3 *A2*A1 *A4 - 15*A3 *A1  - 8*A3
   3    3          2      2
*A2 *A1*A4  + 4*A3*A2 *A1*A4  + 2*A3*A2*A1*A4 + 7*A3*
          3   3        2   2
A1 + 8*A2 *A4  - 2Ø*A2 *A4  + 14*A2*A4 - 3))/ D

               2   2       3   3         2       2        2   2
RI(1,11):=(A3 *A2 *(9*A3 *A1  - 28*A3 *A2*A1 *A4 - A3 *A1 +24*A3*
  2       2                                             3   3
A2 *A1*A4  + 6*A3*A2*A1*A4 - 9*A3*A1 - 8*A2 *A4  + 16*
  2   2
A2 *A4  - 18*A2*A4 + 6))/ D

               3   2        3   3         2       2         2   2
RI(1,12):=(A3 *A2 *( - 3*A3 *A1  + 2Ø*A3 *A2*A1 *A4 - 21*A3 *A1 -
          2      2                                       3
24*A3*A2 *A1*A4  + 18*A3*A2*A1*A4 + 27*A3*A1 + 8*A2 *
  3       2   2
A4  - 16*A2 *A4  + 1Ø*A2*A4 - 1Ø))/ D

                  5   5      4      4         4   4        3   2
RI(2,1):=(A1*(A3 *A1 -1Ø*A3 *A2*A1 *A4 - 7*A3 *A1  + 3Ø*A3 *A2 *
  3   2         3      3         3   3        2   3
A1 *A4  + 12*A3 *A2*A1 *A4 + 17*A3 *A1  - 28*A3 *A2 *
  2   3          2   2   2   2        2       2
A1 *A4  - 32*A3 *A2 *A1 *A4  + 9*A3 *A2*A1 *A4 - 17*
  2   2            4      4          3      3
A3 *A1  + 8*A3*A2 *A1*A4  + 2Ø*A3*A2 *A1*A4  + 2*A3*
  2       2                                    4   4
A2 *A1*A4  - 13*A3*A2*A1*A4 + 7*A3*A1 - 8*A2 *A4 +12
   3   3        2   2
*A2 *A4  - 1Ø*A2 *A4  + 5*A2*A4 - 1))/ D

              5   5      4      4         4   4         3   2   3
RI(2,2):=(-A3 *A1  + 6*A3 *A2*A1 *A4 + 7*A3 *A1  - 22*A3 *A2 *A1 *
     2         3      3         3   3         2   3   2   3
    A4  + 4*A3 *A2*A1 *A4 - 17*A3 *A1  + 2Ø*A3 *A2 *A1 *A4  +
           2   2   2   2         2       2         2   2
    16*A3 *A2 *A1 *A4  - 27*A3 *A2*A1 *A4 + 17*A3 *A1  - 8*A3*
      4      4            3      3          2      2
    A2 *A1*A4  + 4*A3*A2 *A1*A4  - 18*A3*A2 *A1*A4  + 23*A3*A2
                               4   4        3   3        2   2
    *A1*A4 - 7*A3*A1 + 8*A2 *A4  - 2Ø*A2 *A4  + 18*A2 *A4  - 7
    *A2*A4 + 1)/ D

                    4      4          4   4        3   2   3   2      3
RI(2,3):=(A3*(-A3 *A2*A1 *A4 - A3 *A1  + 6*A3 *A2 *A1 *A4  + 3*A3
             3         3   3        2   3   2   3          2
    *A2*A1 *A4 + 6*A3 *A1  - 4*A3 *A2 *A1 *A4  - 38*A3 *
      2   2   2          2       2          2   2
    A2 *A1 *A4  + 24*A3 *A2*A1 *A4 - 11*A3 *A1  + 36*A3*
      3       3          2      2
    A2 *A1*A4  - 12*A3*A2 *A1*A4  - 15*A3*A2*A1*A4 + 6*A3*
              4   4        3   3        2   2
    A1 - 8*A2 *A4  + 12*A2 *A4  - 1Ø*A2 *A4  + 5*A2*A4 - 1
    ))/ D
```

```
            2    4   4       3      3          3   3        2   2
RI(2,4):=(A3 *(A3 *A1  - 3*A3 *A2*A1 *A4 - 6*A3 *A1  + 16*A3 *A2 *
               2   2        2      2           2   2           3
             A1 *A4  - 13*A3 *A2*A1 *A4 + 11*A3 *A1  - 24*A3*A2 *
                  3           2      2
             A1*A4  + 3Ø*A3*A2 *A1*A4  + 3*A3*A2*A1*A4 - 6*A3*A1
                    4   4        3   3       2   2
              + 8*A2 *A4  - 16*A2 *A4  + 6*A2 *A4  - 2*A2*A4 + 1))
           / D

                       4   4     3      3          3   3       2
RI(2,5):=(2*A2*A1*(2*A3 *A1 -4*A3 *A2*A1 *A4 - 8*A3 *A1  + 4*A3 *
             2   2   2       2      2          2   2           2
           A2 *A1 *A4  + 8*A3 *A2*A1 *A4 + 9*A3 *A1  - 12*A3*A2 *
                2                                 2   2
           A1*A4  + 8*A3*A2*A1*A4 - 5*A3*A1 + 4*A2 *A4  - 4*A2*A4
            + 1))/ D

                 5   5       4      4          4   4        3   2
RI(2,6):=(A2*(-A3 *A1  + 8*A3 *A2*A1 *A4 + 3*A3 *A1  - 16*A3 *A2 *
              3   2        3      3        3   3       2   3   2
            A1 *A4  - 16*A3 *A2*A1 *A4 - A3 *A1  + 8*A3 *A2 *A1 *
              3        2   2   2   2        2      2        2   2
            A4  + 36*A3 *A2 *A1 *A4  - 1Ø*A3 *A2*A1 *A4 - A3 *A1
                       3      3           2      2
             - 24*A3*A2 *A1*A4  + 2Ø*A3*A2 *A1*A4  - 1Ø*A3*A2*A1*
                               3   3        2   2
            A4 + 3*A3*A1 + 8*A2 *A4  - 12*A2 *A4  + 6*A2*A4-1))/D

                      4   4       3   3       2   2   2   2
RI(2,7):=(2*A3*A2*(-A3 *A1  + 3*A3 *A1  + 4*A3 *A2 *A1 *A4  - 4*
             2      2              2      2
           A3 *A2*A1 *A4 - 12*A3*A2 *A1*A4  + 12*A3*A2*A1*A4 - 3*
                       2   2
           A3*A1 + 4*A2 *A4  - 4*A2*A4 + 1))/ D

            2       4   4       3      3          2   2   2   2
RI(2,8):=(A3 *A2*(A3 *A1  - 6*A3 *A2*A1 *A4 + 4*A3 *A2 *A1 *A4 +22
              2      2           2   2           2      2
           *A3 *A2*A1 *A4 - 11*A3 *A1  - 12*A3*A2 *A1*A4  - 18*A3*
                                     2   2
           A2*A1*A4 + 12*A3*A1 + 4*A2 *A4  + 4*A2*A4 - 3))/ D
            2         4   4      3      3           3   3        2
RI(2,9):=(A2 *A1*(2*A3 *A1 -14*A3 *A2*A1 *A4 + 12*A3 *A1  + 2Ø*A3
              2   2   2        2      2           2   2          3
           *A2 *A1 *A4  - 12*A3 *A2*A1 *A4 - 15*A3 *A1  - 8*A3*A2
                 3          2      2
           *A1*A4  + 4*A3*A2 *A1*A4  + 2*A3*A2*A1*A4 + 7*A3*A1 + 8
              3   3        2   2
           *A2 *A4  - 2Ø*A2 *A4  + 14*A2*A4 - 3))/ D

             2       4   4        3      3          3   3        2
RI(2,1Ø):=(A2 *(-6*A3 *A1  + 22*A3 *A2*A1 *A4 + 4*A3 *A1  - 28*A3
               2   2   2       2      2          2   2          3
            *A2 *A1 *A4  - 4*A3 *A2*A1 *A4 - 3*A3 *A1  + 8*A3*A2 *
```

```
                 3           2      2
            A1*A4  + 20*A3*A2 *A1*A4  - 18*A3*A2*A1*A4 + 3*A3*A1
                        3   3       2   2
             - 8*A2 *A4  + 12*A2 *A4  - 6*A2*A4 + 1))/ D

                      2     4   4       3      3        3   3      2
RI(2,11):=(A3*A2 *(-A3 *A1  + 6*A3 *A2*A1 *A4 - 9*A3 *A1  - 4*A3 *
              2   2   2        2      2          2   2                2
            A2 *A1 *A4  + 6*A3 *A2*A1 *A4 + 12*A3 *A1  - 12*A3*A2
                   2                                    3   3
            *A1*A4  + 12*A3*A2*A1*A4 - 3*A3*A1 + 8*A2 *A4  - 20*
              2   2
            A2 *A4  + 14*A2*A4 - 3))/ D

                 2   2      3   3        2      2        2   2
RI(2,12):=(A3 *A2 *(9*A3 *A1  - 28*A3 *A2*A1 *A4 - A3 *A1 +24*A3*
              2       2                                  3   3
            A2 *A1*A4  + 6*A3*A2*A1*A4 - 9*A3*A1 - 8*A2 *A4 +16*
              2   2
            A2 *A4  - 18*A2*A4 + 6))/ D

                2     4   4       3      3        3   3      2   2
RI(3,1):=(A1 *(A3 *A1  - 3*A3 *A2*A1 *A4 - 6*A3 *A1 +16*A3 *A2 *
               2   2        2      2           2   2         3
             A1 *A4  - 13*A3 *A2*A1 *A4 + 11*A3 *A1  - 24*A3*A2 *
                  3          2      2
             A1*A4  + 30*A3*A2 *A1*A4  + 3*A3*A2*A1*A4 - 6*A3*A1
                      4   4        3   3       2   2
              + 8*A2 *A4  - 16*A2 *A4  + 6*A2 *A4  - 2*A2*A4+1))
          / D

                    4       4        4   4        3   2   3   2        3
RI(3,2):=(A1*(-A3 *A2*A1 *A4 - A3 *A1  + 6*A3 *A2 *A1 *A4  + 3*A3
                 3          3   3         2   3   2   3           2
            *A2*A1 *A4 + 6*A3 *A1  - 4*A3 *A2 *A1 *A4  - 38*A3 *
              2   2   2         2      2           2   2
            A2 *A1 *A4  + 24*A3 *A2*A1 *A4 - 11*A3 *A1  + 36*A3*
              3      3           2      2
            A2 *A1*A4  - 12*A3*A2 *A1*A4  - 15*A3*A2*A1*A4 + 6*A3*
                     4   4        3   3        2   2
            A1 - 8*A2 *A4  + 12*A2 *A4  - 10*A2 *A4  + 5*A2*A4 - 1
            ))/ D

              5   5       4      4           4   4         3   2   3
RI(3,3):=(-A3 *A1  + 6*A3 *A2*A1 *A4 + 7*A3 *A1  - 22*A3 *A2 *A1 *
        2          3      3           3   3           2   3   2   3
        A4  + 4*A3 *A2*A1 *A4 - 17*A3 *A1  + 20*A3 *A2 *A1 *A4  +
             2   2   2   2          2      2          2   2
        16*A3 *A2 *A1 *A4  - 27*A3 *A2*A1 *A4 + 17*A3 *A1  - 8*A3*
          4      4          3      3           2      2
        A2 *A1*A4  + 4*A3*A2 *A1*A4  - 18*A3*A2 *A1*A4  + 23*A3*A2
                                   4   4        3   3        2   2
        *A1*A4 - 7*A3*A1 + 8*A2 *A4  - 20*A2 *A4  + 18*A2 *A4  - 7
        *A2*A4 + 1)/ D
```

$$\begin{aligned}RI(3,4) := (&A3*(A3^{5}*A1^{5} - 10*A3^{4}*A2*A1^{4}*A4 - 7*A3^{4}*A1^{4} + 30*A3^{3}*A2^{2}*\\
&A1^{3}*A4^{2} + 12*A3^{3}*A2*A1^{3}*A4 + 17*A3^{3}*A1^{3} - 28*A3^{2}*A2^{3}*\\
&A1^{2}*A4^{3} - 32*A3^{2}*A2^{2}*A1^{2}*A4^{2} + 9*A3^{2}*A2*A1^{2}*A4 - 17*\\
&A3^{2}*A1^{2} + 8*A3*A2^{4}*A1*A4^{4} + 20*A3*A2^{3}*A1*A4^{3} + 2*A3*\\
&A2^{2}*A1*A4^{2} - 13*A3*A2*A1*A4 + 7*A3*A1 - 8*A2^{4}*A4^{4} + 12\\
&*A2^{3}*A4^{3} - 10*A2^{2}*A4^{2} + 5*A2*A4 - 1))/ D\end{aligned}$$

$$\begin{aligned}RI(3,5) := (&A2*A1^{2}*(A3^{4}*A1^{4} - 6*A3^{3}*A2*A1^{3}*A4 + 4*A3^{2}*A2^{2}*A1^{2}*A4^{2} + 22\\
&*A3^{2}*A2*A1^{2}*A4 - 11*A3^{2}*A1^{2} - 12*A3*A2^{2}*A1*A4^{2} - 18*A3*\\
&A2*A1*A4 + 12*A3*A1 + 4*A2^{2}*A4^{2} + 4*A2*A4 - 3))/ D\end{aligned}$$

$$\begin{aligned}RI(3,6) := (&2*A2*A1*(-A3^{4}*A1^{4} + 3*A3^{3}*A1^{3} + 4*A3^{2}*A2^{2}*A1^{2}*A4^{2} - 4*\\
&A3^{2}*A2*A1^{2}*A4 - 12*A3*A2^{2}*A1*A4^{2} + 12*A3*A2*A1*A4 - 3*\\
&A3*A1 + 4*A2^{2}*A4^{2} - 4*A2*A4 + 1))/ D\end{aligned}$$

$$\begin{aligned}RI(3,7) := (&A2*(-A3^{5}*A1^{5} + 8*A3^{4}*A2*A1^{4}*A4 + 3*A3^{4}*A1^{4} - 16*A3^{3}*A2^{2}*\\
&A1^{3}*A4^{2} - 16*A3^{3}*A2*A1^{3}*A4 - A3^{3}*A1^{3} + 8*A3^{2}*A2^{3}*A1^{2}*\\
&A4^{3} + 36*A3^{2}*A2^{2}*A1^{2}*A4^{2} - 10*A3^{2}*A2*A1^{2}*A4 - A3^{2}*A1^{2}\\
&- 24*A3*A2^{3}*A1*A4^{3} + 20*A3*A2^{2}*A1*A4^{2} - 10*A3*A2*A1*\\
&A4 + 3*A3*A1 + 8*A2^{3}*A4^{3} - 12*A2^{2}*A4^{2} + 6*A2*A4-1))/D\end{aligned}$$

$$\begin{aligned}RI(3,8) := (&2*A3*A2*(2*A3^{4}*A1^{4} - 4*A3^{3}*A2*A1^{3}*A4 - 8*A3^{3}*A1^{3} + 4*A3^{2}*\\
&A2^{2}*A1^{2}*A4^{2} + 8*A3^{2}*A2*A1^{2}*A4 + 9*A3^{2}*A1^{2} - 12*A3*A2^{2}*\\
&A1*A4^{2} + 8*A3*A2*A1*A4 - 5*A3*A1 + 4*A2^{2}*A4^{2} - 4*A2*A4\\
&+ 1))/ D\end{aligned}$$

$$\begin{aligned}RI(3,9) := (&A2^{2}*A1^{2}*(9*A3^{3}*A1^{3} - 28*A3^{2}*A2*A1^{2}*A4 - A3^{2}*A1^{2} + 24*A3*\\
&A2^{2}*A1*A4^{2} + 6*A3*A2*A1*A4 - 9*A3*A1 - 8*A2^{3}*A4^{3} + 16*\\
&A2^{2}*A4^{2} - 18*A2*A4 + 6))/ D\end{aligned}$$

$$\begin{aligned}RI(3,10) := (&A2^{2}*A1*(-A3^{4}*A1^{4} + 6*A3^{3}*A2*A1^{3}*A4 - 9*A3^{3}*A1^{3} - 4*A3^{2}*\\
&A2^{2}*A1^{2}*A4^{2} + 6*A3^{2}*A2*A1^{2}*A4 + 12*A3^{2}*A1^{2} - 12*A3*A2^{2}\end{aligned}$$

```
                2                                      3   3
       *A1*A4  + 12*A3*A2*A1*A4 - 3*A3*A1 + 8*A2 *A4  - 20*
         2   2
       A2 *A4  + 14*A2*A4 - 3))/ D

             2         4   4          3      3          3   3       2
RI(3,11):=(A2 *(-6*A3 *A1  + 22*A3 *A2*A1 *A4 + 4*A3 *A1  - 28*A3
             2   2   2          2      2          2   2           3
       *A2 *A1 *A4  - 4*A3 *A2*A1 *A4 - 3*A3 *A1  + 8*A3*A2 *
            3           2     2
       A1*A4  + 20*A3*A2 *A1*A4  - 18*A3*A2*A1*A4 + 3*A3*A1
              3   3         2   2
        - 8*A2 *A4  + 12*A2 *A4  - 6*A2*A4 + 1))/ D

                 2        4   4          3      3          3   3
RI(3,12):=(A3*A2 *(2*A3 *A1  - 14*A3 *A2*A1 *A4 + 12*A3 *A1 +20*
         2   2   2   2          2      2           2   2
       A3 *A2 *A1 *A4  - 12*A3 *A2*A1 *A4 - 15*A3 *A1  - 8*A3
          3      3           2     2
       *A2 *A1*A4  + 4*A3*A2 *A1*A4  + 2*A3*A2*A1*A4 + 7*A3*
              3   3          2   2
       A1 + 8*A2 *A4  - 20*A2 *A4  + 14*A2*A4 - 3))/ D

            3      4   4         3      3          3   3         2   2
RI(4,1):=(A1 *(-A3 *A1  + 9*A3 *A2*A1 *A4 + 6*A3 *A1  - 24*A3 *A2
            2   2          2      2           2   2           3
        *A1 *A4  - 9*A3 *A2*A1 *A4 - 11*A3 *A1  + 24*A3*A2 *
             3           2     2
        A1*A4  - 6*A3*A2 *A1*A4  + 15*A3*A2*A1*A4 + 6*A3*A1
               4   4          3   3          2   2
         - 8*A2 *A4  + 16*A2 *A4  - 14*A2 *A4  - 2*A2*A4 - 1)
       )/ D

            2     4   4         3      3           3   3          2   2
RI(4,2):=(A1 *(A3 *A1 -3*A3 *A2*A1 *A4 - 6*A3 *A1  + 16*A3 *A2 *
           2   2           2      2           2   2           3
        A1 *A4  - 13*A3 *A2*A1 *A4 + 11*A3 *A1  - 24*A3*A2 *
             3            2     2
        A1*A4  + 30*A3*A2 *A1*A4  + 3*A3*A2*A1*A4 - 6*A3*A1
               4   4          3   3         2   2
         + 8*A2 *A4  - 16*A2 *A4  + 6*A2 *A4  - 2*A2*A4 + 1))
       / D

                 5   5          4      4           4   4          3   2
RI(4,3):=(A1*(A3 *A1 -10*A3 *A2*A1 *A4 - 7*A3 *A1  + 30*A3 *A2 *
          3   2           3      3           3   3          2   3
       A1 *A4  + 12*A3 *A2*A1 *A4 + 17*A3 *A1  - 28*A3 *A2 *
          2   3           2   2   2   2          2      2
       A1 *A4  - 32*A3 *A2 *A1 *A4  + 9*A3 *A2*A1 *A4 - 17*
          2   2           4     4           3     3
       A3 *A1  + 8*A3*A2 *A1*A4  + 20*A3*A2 *A1*A4  + 2*A3*
          2     2                                      4   4
       A2 *A1*A4  - 13*A3*A2*A1*A4 + 7*A3*A1 - 8*A2 *A4  +12
          3   3          2   2
       *A2 *A4  - 10*A2 *A4  + 5*A2*A4 - 1))/ D
```

$$RI(4,4):=(-2 \cdot A3^{5} \cdot A1^{5} + 9 \cdot A3^{4} \cdot A2 \cdot A1^{4} \cdot A4 + 13 \cdot A3^{4} \cdot A1^{4} - 38 \cdot A3^{3} \cdot A2^{2} \cdot A1^{3} \cdot A4^{2} + 17 \cdot A3^{3} \cdot A2 \cdot A1^{3} \cdot A4 - 28 \cdot A3^{3} \cdot A1^{3} + 44 \cdot A3^{2} \cdot A2^{3} \cdot A1^{2} \cdot A4^{3} - 14 \cdot A3^{2} \cdot A2^{2} \cdot A1^{2} \cdot A4^{2} - 30 \cdot A3^{2} \cdot A2 \cdot A1^{2} \cdot A4 + 23 \cdot A3^{2} \cdot A1^{2} - 16 \cdot A3 \cdot A2^{4} \cdot A1 \cdot A4^{4} + 20 \cdot A3 \cdot A2^{3} \cdot A1 \cdot A4^{3} - 24 \cdot A3 \cdot A2^{2} \cdot A1 \cdot A4^{2} + 25 \cdot A3 \cdot A2 \cdot A1 \cdot A4 - 8 \cdot A3 \cdot A1 + 8 \cdot A2^{4} \cdot A4^{4} - 20 \cdot A2^{3} \cdot A4^{3} + 18 \cdot A2^{2} \cdot A4^{2} - 7 \cdot A2 \cdot A4 + 1)/ D$$

$$RI(4,5):=(2 \cdot A2 \cdot A1^{3} \cdot (-3 \cdot A3^{3} \cdot A1^{3} + 4 \cdot A3^{2} \cdot A2 \cdot A1^{2} \cdot A4 + 11 \cdot A3^{2} \cdot A1^{2} - 12 \cdot A3 \cdot A2 \cdot A1 \cdot A4 - 9 \cdot A3 \cdot A1 + 4 \cdot A2 \cdot A4 + 2))/ D$$

$$RI(4,6):=(A2 \cdot A1^{2} \cdot (A3^{4} \cdot A1^{4} - 6 \cdot A3^{3} \cdot A2 \cdot A1^{3} \cdot A4 + 4 \cdot A3^{2} \cdot A2^{2} \cdot A1^{2} \cdot A4^{2} + 22 \cdot A3^{2} \cdot A2 \cdot A1^{2} \cdot A4 - 11 \cdot A3^{2} \cdot A1^{2} - 12 \cdot A3 \cdot A2^{2} \cdot A1 \cdot A4^{2} - 18 \cdot A3 \cdot A2 \cdot A1 \cdot A4 + 12 \cdot A3 \cdot A1 + 4 \cdot A2^{2} \cdot A4^{2} + 4 \cdot A2 \cdot A4 - 3))/ D$$

$$RI(4,7):=(2 \cdot A2 \cdot A1 \cdot (2 \cdot A3^{4} \cdot A1^{4} - 4 \cdot A3^{3} \cdot A2 \cdot A1^{3} \cdot A4 - 8 \cdot A3^{3} \cdot A1^{3} + 4 \cdot A3^{2} \cdot A2^{2} \cdot A1^{2} \cdot A4^{2} + 8 \cdot A3^{2} \cdot A2 \cdot A1^{2} \cdot A4 + 9 \cdot A3^{2} \cdot A1^{2} - 12 \cdot A3 \cdot A2^{2} \cdot A1 \cdot A4^{2} + 8 \cdot A3 \cdot A2 \cdot A1 \cdot A4 - 5 \cdot A3 \cdot A1 + 4 \cdot A2^{2} \cdot A4^{2} - 4 \cdot A2 \cdot A4 + 1))/ D$$

$$RI(4,8):=(A2 \cdot (-2 \cdot A3^{5} \cdot A1^{5} + 14 \cdot A3^{4} \cdot A2 \cdot A1^{4} \cdot A4 + 3 \cdot A3^{4} \cdot A1^{4} - 20 \cdot A3^{3} \cdot A2^{2} \cdot A1^{3} \cdot A4^{2} - 38 \cdot A3^{3} \cdot A2 \cdot A1^{3} \cdot A4 + 10 \cdot A3^{3} \cdot A1^{3} + 8 \cdot A3^{2} \cdot A2^{3} \cdot A1^{2} \cdot A4^{3} + 48 \cdot A3^{2} \cdot A2^{2} \cdot A1^{2} \cdot A4^{2} + 8 \cdot A3^{2} \cdot A2 \cdot A1^{2} \cdot A4 - 13 \cdot A3^{2} \cdot A1^{2} - 24 \cdot A3 \cdot A2^{3} \cdot A1 \cdot A4^{3} + 16 \cdot A3 \cdot A2^{2} \cdot A1 \cdot A4^{2} - 14 \cdot A3 \cdot A2 \cdot A1 \cdot A4 + 6 \cdot A3 \cdot A1 + 8 \cdot A2^{3} \cdot A4^{3} - 12 \cdot A2^{2} \cdot A4^{2} + 6 \cdot A2 \cdot A4 - 1))/ D$$

$$RI(4,9):=(A2^{2} \cdot A1^{3} \cdot (-3 \cdot A3^{3} \cdot A1^{3} + 20 \cdot A3^{2} \cdot A2 \cdot A1^{2} \cdot A4 - 21 \cdot A3^{2} \cdot A1^{2} - 24 \cdot A3 \cdot A2^{2} \cdot A1 \cdot A4^{2} + 18 \cdot A3 \cdot A2 \cdot A1 \cdot A4 + 27 \cdot A3 \cdot A1 + 8 \cdot A2^{3} \cdot A4^{3} - 16 \cdot A2^{2} \cdot A4^{2} + 10 \cdot A2 \cdot A4 - 10))/ D$$

$$RI(4,10):=(A2^{2} \cdot A1^{2} \cdot (9 \cdot A3^{3} \cdot A1^{3} - 28 \cdot A3^{2} \cdot A2 \cdot A1^{2} \cdot A4 - A3^{2} \cdot A1^{2} + 24 \cdot A3 \cdot A2^{2} \cdot A1 \cdot A4^{2} + 6 \cdot A3 \cdot A2 \cdot A1 \cdot A4 - 9 \cdot A3 \cdot A1 - 8 \cdot A2^{3} \cdot A4^{3} + 16 \cdot A2^{2} \cdot A4^{2} - 18 \cdot A2 \cdot A4 + 6))/ D$$

$$RI(4,11):=(A2^{2}*A1*(2*A3^{4}*A1^{4} - 14*A3^{3}*A2*A1^{3}*A4 + 12*A3^{3}*A1^{3} +20*A3^{2}*A2^{2}*A1^{2}*A4^{2} - 12*A3^{2}*A2*A1^{2}*A4 - 15*A3^{2}*A1^{2} - 8*A3*A2^{3}*A1*A4^{3} + 4*A3*A2^{2}*A1*A4^{2} + 2*A3*A2*A1*A4 + 7*A3*A1 + 8*A2^{3}*A4^{3} - 20*A2^{2}*A4^{2} + 14*A2*A4 - 3))/ D$$

$$RI(4,12):=(A2^{2}*(- 15*A3^{4}*A1^{4} + 50*A3^{3}*A2*A1^{3}*A4 + 5*A3^{3}*A1^{3} -52*A3^{2}*A2^{2}*A1^{2}*A4^{2} - 10*A3^{2}*A2*A1^{2}*A4 + 6*A3^{2}*A1^{2} + 16*A3*A2^{3}*A1*A4^{3} + 4*A3*A2^{2}*A1*A4^{2} - 3*A3*A1 - 8*A2^{3}*A4^{3} + 12*A2^{2}*A4^{2} - 6*A2*A4 + 1))/ D$$

$$RI(5,1):=(A4*(-2*A3^{5}*A1^{5} + 14*A3^{4}*A2*A1^{4}*A4 + 3*A3^{4}*A1^{4} - 20*A3^{3}*A2^{2}*A1^{3}*A4^{2} - 38*A3^{3}*A2*A1^{3}*A4 + 10*A3^{3}*A1^{3} + 8*A3^{2}*A2^{3}*A1^{2}*A4^{3} + 48*A3^{2}*A2^{2}*A1^{2}*A4^{2} + 8*A3^{2}*A2*A1^{2}*A4 - 13*A3^{2}*A1^{2} - 24*A3*A2^{3}*A1*A4^{3} + 16*A3*A2^{2}*A1*A4^{2} - 14*A3*A2*A1*A4 + 6*A3*A1 + 8*A2^{3}*A4^{3} - 12*A2^{2}*A4^{2} + 6*A2*A4 - 1))/ D$$

$$RI(5,2):=(2*A3*A4*(2*A3^{4}*A1^{4} -4*A3^{3}*A2*A1^{3}*A4 - 8*A3^{3}*A1^{3} + 4*A3^{2}*A2^{2}*A1^{2}*A4^{2} + 8*A3^{2}*A2*A1^{2}*A4 + 9*A3^{2}*A1^{2} - 12*A3*A2^{2}*A1*A4^{2} + 8*A3*A2*A1*A4 - 5*A3*A1 + 4*A2^{2}*A4^{2} - 4*A2*A4 + 1))/ D$$

$$RI(5,3):=(A3^{2}*A4*(A3^{4}*A1^{4} -6*A3^{3}*A2*A1^{3}*A4 + 4*A3^{2}*A2^{2}*A1^{2}*A4^{2} +22*A3^{2}*A2*A1^{2}*A4 - 11*A3^{2}*A1^{2} - 12*A3*A2^{2}*A1*A4^{2} - 18*A3*A2*A1*A4 + 12*A3*A1 + 4*A2^{2}*A4^{2} + 4*A2*A4 - 3))/ D$$

$$RI(5,4):=(2*A3^{3}*A4*(-3*A3^{3}*A1^{3} + 4*A3^{2}*A2*A1^{2}*A4 + 11*A3^{2}*A1^{2} -12*A3*A2*A1*A4 - 9*A3*A1 + 4*A2*A4 + 2))/ D$$

$$RI(5,5):=(-2*A3^{5}*A1^{5} - 6*A3^{4}*A2*A1^{4}*A4 + 13*A3^{4}*A1^{4} + 12*A3^{3}*A2^{2}*A1^{3}*A4^{2} + 22*A3^{3}*A2*A1^{3}*A4 - 28*A3^{3}*A1^{3} - 8*A3^{2}*A2^{3}*A1^{2}*A4^{3} - 24*A3^{2}*A2^{2}*A1^{2}*A4^{2} - 24*A3^{2}*A2*A1^{2}*A4 + 23*A3^{2}*A1^{2}$$

```
                       3         3               2         2
      + 24*A3*A2 *A1*A4  - 24*A3*A2 *A1*A4  + 22*A3*A2*A1*A4 -
                   3    3          2     2
     8*A3*A1 - 8*A2 *A4  + 12*A2 *A4  - 6*A2*A4 + 1)/ D

                  5    5      4         4          4    4        3     2   3
RI(5,6):=(A3*(A3 *A1 -8*A3 *A2*A1 *A4 - 7*A3 *A1  + 16*A3 *A2 *A1
               2           3        3            3    3        2    3    2
          *A4  + 24*A3 *A2*A1 *A4 + 17*A3 *A1  - 8*A3 *A2 *A1 *
           3          2    2    2    2         2        2         2
          A4  - 44*A3 *A2 *A1 *A4  - 6*A3 *A2*A1 *A4 - 17*A3 *
           2            3        3          2       2
          A1  + 24*A3*A2 *A1*A4  + 4*A3*A2 *A1*A4  - 6*A3*A2*A1*
                                 3    3          2     2
          A4 + 7*A3*A1 - 8*A2 *A4  + 4*A2 *A4  + 2*A2*A4 - 1))/D

             2     4    4      3        3            3    3         2    2
RI(5,7):=(A3 *(A3 *A1 +6*A3 *A2*A1 *A4 - 6*A3 *A1  - 12*A3 *A2 *
             2    2          2       2           2    2            2
           A1 *A4  - 14*A3 *A2*A1 *A4 + 11*A3 *A1  + 36*A3*A2 *
                2                                     2    2
           A1*A4  - 6*A3*A2*A1*A4 - 6*A3*A1 - 12*A2 *A4  + 4*A2*
           A4 + 1))/ D

             3     4    4         3        3            3    3      2    2
RI(5,8):=(A3 *(-A3 *A1  + 6*A3 *A2*A1 *A4 + 6*A3 *A1 -4*A3 *A2 *
             2    2          2       2           2    2            2
           A1 *A4  - 30*A3 *A2*A1 *A4 - 11*A3 *A1  + 12*A3*A2 *
                2                                    2    2
           A1*A4  + 42*A3*A2*A1*A4 + 6*A3*A1 - 4*A2 *A4 -12*A2
           *A4 - 1))/ D

                        5    5         4       4            4    4        3
RI(5,9):=(A2*(-2*A3 *A1  + 14*A3 *A2*A1 *A4 + 3*A3 *A1  - 20*A3 *
             2    3   2          3       3            3    3        2
          A2 *A1 *A4  - 38*A3 *A2*A1 *A4 + 10*A3 *A1  + 8*A3 *
            3    2   3         2    2    2    2         2       2
          A2 *A1 *A4  + 48*A3 *A2 *A1 *A4  + 8*A3 *A2*A1 *A4 -
               2    2          3        3          2       2
          13*A3 *A1  - 24*A3*A2 *A1*A4  + 16*A3*A2 *A1*A4  - 14*
                                         3    3          2    2
          A3*A2*A1*A4 + 6*A3*A1 + 8*A2 *A4  - 12*A2 *A4  + 6*A2*
          A4 - 1))/ D

                               4    4       3        3          3    3      2
RI(5,10):=(2*A3*A2*(2*A3 *A1 -4*A3 *A2*A1 *A4 - 8*A3 *A1 +4*A3 *
             2    2   2         2       2           2    2            2
          A2 *A1 *A4  + 8*A3 *A2*A1 *A4 + 9*A3 *A1  - 12*A3*A2 *
               2                                    2    2
          A1*A4  + 8*A3*A2*A1*A4 - 5*A3*A1 + 4*A2 *A4  - 4*A2*A4
           + 1))/ D

              2        4    4      3       3            2    2    2   2
RI(5,11):=(A3 *A2*(A3 *A1 -6*A3 *A2*A1 *A4 + 4*A3 *A2 *A1 *A4  +
               2       2           2    2            2        2
          22*A3 *A2*A1 *A4 - 11*A3 *A1  - 12*A3*A2 *A1*A4  - 18*
```

```
                                               2   2
        A3*A2*A1*A4 + 12*A3*A1 + 4*A2 *A4  + 4*A2*A4 - 3))/ D

            3               3   3          2       2             2   2
RI(5,12):=(2*A3 *A2*( - 3*A3 *A1  + 4*A3 *A2*A1 *A4 + 11*A3 *A1 -
        12*A3*A2*A1*A4 - 9*A3*A1 + 4*A2*A4 + 2))/ D

                        4   4        3       3           3   3    2
RI(6,1):=(2*A1*A4*(2*A3 *A1  - 4*A3 *A2*A1 *A4 - 8*A3 *A1 +4*A3 *
             2   2   2       2   2            2   2            2
        A2 *A1 *A4  + 8*A3 *A2*A1 *A4 + 9*A3 *A1  - 12*A3*A2 *
            2                                   2   2
        A1*A4  + 8*A3*A2*A1*A4 - 5*A3*A1 + 4*A2 *A4  - 4*A2*A4
        + 1))/ D

                  5   5       4       4           4   4        3   2
RI(6,2):=(A4*(-A3 *A1  + 8*A3 *A2*A1 *A4 + 3*A3 *A1  - 16*A3 *A2 *
           3  2        3     3          3   3        2    3   2
        A1 *A4  - 16*A3 *A2*A1 *A4 - A3 *A1  + 8*A3 *A2 *A1 *
           3           2   2   2   2          2       2          2   2
        A4  + 36*A3 *A2 *A1 *A4  - 10*A3 *A2*A1 *A4 - A3 *A1
                     3      3               2      2
         - 24*A3*A2 *A1*A4  + 20*A3*A2 *A1*A4  - 10*A3*A2*A1*
                             3   3         2   2
        A4 + 3*A3*A1 + 8*A2 *A4  - 12*A2 *A4  + 6*A2*A4-1))/D

                          4   4       3   3        2   2   2   2
RI(6,3):=(2*A3*A4*( - A3 *A1  + 3*A3 *A1  + 4*A3 *A2 *A1 *A4 -4*
            2       2             2      2
        A3 *A2*A1 *A4 - 12*A3*A2 *A1*A4  + 12*A3*A2*A1*A4 - 3*
                         2   2
        A3*A1 + 4*A2 *A4  - 4*A2*A4 + 1))/ D

             2       4   4        3       3          2   2   2   2
RI(6,4):=(A3 *A4*(A3 *A1  - 6*A3 *A2*A1 *A4 + 4*A3 *A2 *A1 *A4 +22
            2       2          2   2          2       2
        *A3 *A2*A1 *A4 - 11*A3 *A1  - 12*A3*A2 *A1*A4  - 18*A3*
                                       2   2
        A2*A1*A4 + 12*A3*A1 + 4*A2 *A4  + 4*A2*A4 - 3))/ D

                   5   5      4       4          4   4        3   2   3
RI(6,5):=(A1*(A3 *A1 -8*A3 *A2*A1 *A4 - 7*A3 *A1  + 16*A3 *A2 *A1
            2        3      3           3   3       2    3   2
        *A4  + 24*A3 *A2*A1 *A4 + 17*A3 *A1  - 8*A3 *A2 *A1 *
           3           2   2   2   2         2       2            2
        A4  - 44*A3 *A2 *A1 *A4  - 6*A3 *A2*A1 *A4 - 17*A3 *
           2              3      3             2      2
        A1  + 24*A3*A2 *A1*A4  + 4*A3*A2 *A1*A4  - 6*A3*A2*A1*
                             3   3        2   2
        A4 + 7*A3*A1 - 8*A2 *A4  + 4*A2 *A4  + 2*A2*A4 - 1))/D

              5   5       4   4        3       3             3   3
RI(6,6):=(-A3 *A1  + 7*A3 *A1  + 8*A3 *A2*A1 *A4 - 17*A3 *A1 -8*
          2   3   2   3         2   2   2   2          2       2
       A3 *A2 *A1 *A4  + 12*A3 *A2 *A1 *A4  - 30*A3 *A2*A1 *A4 +
             2   2            3      3            2      2
       17*A3 *A1  + 24*A3*A2 *A1*A4  - 36*A3*A2 *A1*A4  + 26*A3*
```

```
                                 3   3        2   2
       A2*A1*A4 - 7*A3*A1 - 8*A2 *A4  + 12*A2 *A4  - 6*A2*A4 + 1)
      / D

                   4      4        4   4        3   2   3   2
RI(6,7):=(A3*(-2*A3 *A2*A1 *A4 - A3 *A1  + 12*A3 *A2 *A1 *A4 -6*
             3      3          3   3       2   3   2   3        2
           A3 *A2*A1 *A4 + 6*A3 *A1  - 8*A3 *A2 *A1 *A4  - 32*A3
              2   2   2        2      2           2   2
           *A2 *A1 *A4  + 36*A3 *A2*A1 *A4 - 11*A3 *A1  + 24*A3*
             3      3                                  3   3
           A2 *A1*A4  - 18*A3*A2*A1*A4 + 6*A3*A1 - 8*A2 *A4  + 4*
             2   2
           A2 *A4  + 2*A2*A4 - 1))/ D

            2    4   4       3      3          3   3        2   2
RI(6,8):=(A3 *(A3 *A1  + 6*A3 *A2*A1 *A4 - 6*A3 *A1  - 12*A3 *A2 *
              2   2        2      2           2   2           2
            A1 *A4  - 14*A3 *A2*A1 *A4 + 11*A3 *A1  + 36*A3*A2 *
                 2                                  2   2
            A1*A4  - 6*A3*A2*A1*A4 - 6*A3*A1 - 12*A2 *A4  + 4*A2*
            A4 + 1))/ D

                       4   4     3      3          3   3       2
RI(6,9):=(2*A2*A1*(2*A3 *A1 -4*A3 *A2*A1 *A4 - 8*A3 *A1  + 4*A3 *
            2   2   2       2      2          2   2           2
          A2 *A1 *A4  + 8*A3 *A2*A1 *A4 + 9*A3 *A1  - 12*A3*A2 *
               2                                 2   2
          A1*A4  + 8*A3*A2*A1*A4 - 5*A3*A1 + 4*A2 *A4  - 4*A2*A4
           + 1))/ D

                  5   5       4      4          4   4        3   2
RI(6,10):=(A2*(-A3 *A1  + 8*A3 *A2*A1 *A4 + 3*A3 *A1  - 16*A3 *A2
               3   2        3      3        3   3       2   3   2
            *A1 *A4  - 16*A3 *A2*A1 *A4 - A3 *A1  + 8*A3 *A2 *A1
               3        2   2   2   2        2      2        2
            *A4  + 36*A3 *A2 *A1 *A4  - 10*A3 *A2*A1 *A4 - A3 *
              2           3      3           2      2
            A1  - 24*A3*A2 *A1*A4  + 20*A3*A2 *A1*A4  - 10*A3*A2*
                                  3   3        2   2
            A1*A4 + 3*A3*A1 + 8*A2 *A4  - 12*A2 *A4  + 6*A2*A4 -
            1))/ D

                       4   4       3   3       2   2   2   2
RI(6,11):=(2*A3*A2*(-A3 *A1  + 3*A3 *A1  + 4*A3 *A2 *A1 *A4  - 4*
             2      2              2      2
           A3 *A2*A1 *A4 - 12*A3*A2 *A1*A4  + 12*A3*A2*A1*A4 - 3*
                       2   2
           A3*A1 + 4*A2 *A4  - 4*A2*A4 + 1))/ D

             2       4   4     3      3          2   2   2   2
RI(6,12):=(A3 *A2*(A3 *A1 -6*A3 *A2*A1 *A4 + 4*A3 *A2 *A1 *A4  +
                2      2           2   2           2      2
           22*A3 *A2*A1 *A4 - 11*A3 *A1  - 12*A3*A2 *A1*A4  - 18*
                                        2   2
           A3*A2*A1*A4 + 12*A3*A1 + 4*A2 *A4  + 4*A2*A4 - 3))/ D
```

```
               2      4    4         3       3              2    2    2    2
RI(7,1):=(A1 *A4*(A3 *A1   - 6*A3 *A2*A1 *A4 + 4*A3 *A2 *A1 *A4 +22
            2       2               2    2              2       2
         *A3 *A2*A1 *A4 - 11*A3 *A1   - 12*A3*A2 *A1*A4   - 18*A3*
                                        2    2
         A2*A1*A4 + 12*A3*A1 + 4*A2 *A4   + 4*A2*A4 - 3))/ D

                               4    4          3    3          2    2    2    2
RI(7,2):=(2*A1*A4*( - A3 *A1   + 3*A3 *A1   + 4*A3 *A2 *A1 *A4   - 4*
           2       2               2         2
         A3 *A2*A1 *A4 - 12*A3*A2 *A1*A4   + 12*A3*A2*A1*A4 - 3*
                    2    2
         A3*A1 + 4*A2 *A4   - 4*A2*A4 + 1))/ D

                   5    5        4       4             4    4          3    2
RI(7,3):=(A4*(-A3 *A1   + 8*A3 *A2*A1 *A4 + 3*A3 *A1   - 16*A3 *A2 *
            3   2          3      3           3   3         2    3    2
          A1 *A4   - 16*A3 *A2*A1 *A4 - A3 *A1   + 8*A3 *A2 *A1 *
            3            2    2   2    2           2       2         2   2
          A4   + 36*A3 *A2 *A1 *A4   - 10*A3 *A2*A1 *A4 - A3 *A1
                        3         3           2         2
           - 24*A3*A2 *A1*A4   + 20*A3*A2 *A1*A4   - 10*A3*A2*A1*
                                  3    3            2    2
          A4 + 3*A3*A1 + 8*A2 *A4   - 12*A2 *A4   + 6*A2*A4 - 1))/D

                              4    4          3        3             3    3     2
RI(7,4):=(2*A3*A4*(2*A3 *A1   - 4*A3 *A2*A1 *A4 - 8*A3 *A1 +4*A3 *
            2    2    2          2        2          2    2           2
         A2 *A1 *A4   + 8*A3 *A2*A1 *A4 + 9*A3 *A1   - 12*A3*A2 *
              2                                          2    2
         A1*A4   + 8*A3*A2*A1*A4 - 5*A3*A1 + 4*A2 *A4   - 4*A2*A4
          + 1))/ D

               2     4    4       3       3            3    3          2    2
RI(7,5):=(A1 *(A3 *A1 +6*A3 *A2*A1 *A4 - 6*A3 *A1   - 12*A3 *A2 *
              2   2          2       2            2   2           2
           A1 *A4   - 14*A3 *A2*A1 *A4 + 11*A3 *A1   + 36*A3*A2 *
                 2                                        2    2
           A1*A4   - 6*A3*A2*A1*A4 - 6*A3*A1 - 12*A2 *A4   + 4*A2*
           A4 + 1))/ D

                       4       4           4    4          3    2    3    2
RI(7,6):=(A1*(-2*A3 *A2*A1 *A4 - A3 *A1   + 12*A3 *A2 *A1 *A4 -6*
            3       3           3    3         2    3    2    3           2
          A3 *A2*A1 *A4 + 6*A3 *A1   - 8*A3 *A2 *A1 *A4   - 32*A3
             2    2    2          2       2            2    2
          *A2 *A1 *A4   + 36*A3 *A2*A1 *A4 - 11*A3 *A1   + 24*A3*
            3         3                                       3    3
          A2 *A1*A4   - 18*A3*A2*A1*A4 + 6*A3*A1 - 8*A2 *A4   + 4*
            2    2
          A2 *A4   + 2*A2*A4 - 1))/ D

             5    5         4    4          3       3              3    3
RI(7,7):=(-A3 *A1   + 7*A3 *A1   + 8*A3 *A2*A1 *A4 - 17*A3 *A1 -8*
          2    3    2    3           2    2    2    2           2       2
       A3 *A2 *A1 *A4   + 12*A3 *A2 *A1 *A4   - 30*A3 *A2*A1 *A4 +
```

```
            2   2           3      3           2      2
       17*A3 *A1  + 24*A3*A2 *A1*A4  - 36*A3*A2 *A1*A4  + 26*A3*
                                3   3        2   2
       A2*A1*A4 - 7*A3*A1 - 8*A2 *A4  + 12*A2 *A4  - 6*A2*A4 + 1)
      / D

                5   5     4      4          4   4      3   2   3
RI(7,8):=(A3*(A3 *A1 -8*A3 *A2*A1 *A4 - 7*A3 *A1 +16*A3 *A2 *A1
              2        3      3           3   3       2   3   2
           *A4  + 24*A3 *A2*A1 *A4 + 17*A3 *A1  - 8*A3 *A2 *A1 *
             3        2   2   2   2       2      2           2
           A4  - 44*A3 *A2 *A1 *A4  - 6*A3 *A2*A1 *A4 - 17*A3 *
             2           3      3          2      2
           A1  + 24*A3*A2 *A1*A4  + 4*A3*A2 *A1*A4  - 6*A3*A2*A1*
                              3   3       2   2
           A4 + 7*A3*A1 - 8*A2 *A4  + 4*A2 *A4  + 2*A2*A4-1))/D

               2    4   4     3      3          2   2   2   2
RI(7,9):=(A2*A1 *(A3 *A1 -6*A3 *A2*A1 *A4 + 4*A3 *A2 *A1 *A4 +22
             2      2           2   2           2      2
          *A3 *A2*A1 *A4 - 11*A3 *A1  - 12*A3*A2 *A1*A4  - 18*A3*
                                    2   2
          A2*A1*A4 + 12*A3*A1 + 4*A2 *A4  + 4*A2*A4 - 3))/ D

                       4   4       3   3       2   2   2   2
RI(7,10):=(2*A2*A1*(-A3 *A1  + 3*A3 *A1  + 4*A3 *A2 *A1 *A4 - 4*
             2      2              2      2
           A3 *A2*A1 *A4 - 12*A3*A2 *A1*A4  + 12*A3*A2*A1*A4 - 3*
                       2   2
           A3*A1 + 4*A2 *A4  - 4*A2*A4 + 1))/ D

                  5   5     4      4          4   4        3   2
RI(7,11):=(A2*(-A3 *A1 +8*A3 *A2*A1 *A4 + 3*A3 *A1  - 16*A3 *A2
               3   2        3      3        3   3       2   3   2
            *A1 *A4  - 16*A3 *A2*A1 *A4 - A3 *A1  + 8*A3 *A2 *A1
               3        2   2   2   2        2      2        2
            *A4  + 36*A3 *A2 *A1 *A4  - 10*A3 *A2*A1 *A4 - A3 *
              2           3      3           2      2
            A1  - 24*A3*A2 *A1*A4  + 20*A3*A2 *A1*A4  - 10*A3*A2*
                                  3   3        2   2
            A1*A4 + 3*A3*A1 + 8*A2 *A4  - 12*A2 *A4  + 6*A2*A4 -
            1))/ D

                        4   4     3      3          3   3     2
RI(7,12):=(2*A3*A2*(2*A3 *A1 -4*A3 *A2*A1 *A4 - 8*A3 *A1 +4*A3 *
             2   2   2       2      2          2   2           2
           A2 *A1 *A4  + 8*A3 *A2*A1 *A4 + 9*A3 *A1  - 12*A3*A2 *
                2                                 2   2
           A1*A4  + 8*A3*A2*A1*A4 - 5*A3*A1 + 4*A2 *A4  - 4*A2*A4
            + 1))/ D

              3          3   3       2      2           2   2
RI(8,1):=(2*A1 *A4*(-3*A3 *A1  + 4*A3 *A2*A1 *A4 + 11*A3 *A1 -12
          *A3*A2*A1*A4 - 9*A3*A1 + 4*A2*A4 + 2))/ D
```

```
            2       4   4     3      3          2   2   2   2
RI(8,2):=(A1 *A4*(A3 *A1 -6*A3 *A2*A1 *A4 + 4*A3 *A2 *A1 *A4 +22
             2      2           2   2           2      2
          *A3 *A2*A1 *A4 - 11*A3 *A1  - 12*A3*A2 *A1*A4  - 18*A3*
                                    2   2
          A2*A1*A4 + 12*A3*A1 + 4*A2 *A4  + 4*A2*A4 - 3))/ D

                       4   4     3      3          3   3     2
RI(8,3):=(2*A1*A4*(2*A3 *A1 -4*A3 *A2*A1 *A4 - 8*A3 *A1 +4*A3 *
            2   2   2       2      2          2   2           2
          A2 *A1 *A4  + 8*A3 *A2*A1 *A4 + 9*A3 *A1  - 12*A3*A2 *
               2                                 2   2
          A1*A4  + 8*A3*A2*A1*A4 - 5*A3*A1 + 4*A2 *A4  - 4*A2*A4
           + 1))/ D

                   5   5        4      4          4   4      3
RI(8,4):=(A4*(-2*A3 *A1  + 14*A3 *A2*A1 *A4 + 3*A3 *A1 -20*A3 *
             2   3   2        3      3           3   3       2
           A2 *A1 *A4  - 38*A3 *A2*A1 *A4 + 10*A3 *A1  + 8*A3 *
             3   2   3        2   2   2   2       2      2
           A2 *A1 *A4  + 48*A3 *A2 *A1 *A4  + 8*A3 *A2*A1 *A4 -
                2   2           3      3           2      2
           13*A3 *A1  - 24*A3*A2 *A1*A4  + 16*A3*A2 *A1*A4 -14*
                                       3   3        2   2
           A3*A2*A1*A4 + 6*A3*A1 + 8*A2 *A4  - 12*A2 *A4  + 6*A2*
           A4 - 1))/ D

            3     4   4       3      3          3   3     2   2
RI(8,5):=(A1 *(-A3 *A1  + 6*A3 *A2*A1 *A4 + 6*A3 *A1 -4*A3 *A2 *
              2   2        2      2           2   2           2
            A1 *A4  - 30*A3 *A2*A1 *A4 - 11*A3 *A1  + 12*A3*A2 *
                 2                                  2   2
            A1*A4  + 42*A3*A2*A1*A4 + 6*A3*A1 - 4*A2 *A4  - 12*A2
            *A4 - 1))/ D

            2    4   4     3      3          3   3        2   2
RI(8,6):=(A1 *(A3 *A1 +6*A3 *A2*A1 *A4 - 6*A3 *A1  - 12*A3 *A2 *
              2   2        2      2           2   2           2
            A1 *A4  - 14*A3 *A2*A1 *A4 + 11*A3 *A1  + 36*A3*A2 *
                 2                                  2   2
            A1*A4  - 6*A3*A2*A1*A4 - 6*A3*A1 - 12*A2 *A4  + 4*A2*
            A4 + 1))/ D

                5   5     4      4        4   4        3   2   3
RI(8,7):=(A1*(A3 *A1 -8*A3 *A2*A1 *A4-7*A3 *A1  + 16*A3 *A2 *A1
              2        3      3           3   3       2   3   2
           *A4  + 24*A3 *A2*A1 *A4 + 17*A3 *A1  - 8*A3 *A2 *A1 *
             3        2   2   2   2       2      2           2
           A4  - 44*A3 *A2 *A1 *A4  - 6*A3 *A2*A1 *A4 - 17*A3 *
             2           3      3          2      2
           A1  + 24*A3*A2 *A1*A4  + 4*A3*A2 *A1*A4  - 6*A3*A2*A1*
                              3   3       2   2
           A4 + 7*A3*A1 - 8*A2 *A4  + 4*A2 *A4  + 2*A2*A4 - 1))/ D

               5   5       4      4           4   4        3   2
RI(8,8):=(-2*A3 *A1  - 6*A3 *A2*A1 *A4 + 13*A3 *A1  + 12*A3 *A2 *
```

```
  3   2          3       3             3   3         2   3   2
A1 *A4  + 22*A3 *A2*A1 *A4 - 28*A3 *A1   - 8*A3 *A2 *A1 *
 3         2    2    2    2           2      2           2   2
A4  - 24*A3 *A2 *A1 *A4  - 24*A3 *A2*A1 *A4 + 23*A3 *A1
             3      3             2      2
 + 24*A3*A2 *A1*A4  - 24*A3*A2 *A1*A4  + 22*A3*A2*A1*A4 -
            3   3           2   2
8*A3*A1 - 8*A2 *A4  + 12*A2 *A4  - 6*A2*A4 + 1)/ D

                     3          3   3      2      2           2   2
RI(8,9):=(2*A2*A1 *( - 3*A3 *A1 +4*A3 *A2*A1 *A4 + 11*A3 *A1 -12
        *A3*A2*A1*A4 - 9*A3*A1 + 4*A2*A4 + 2))/ D

                  2    4   4      3      3          2   2   2   2
RI(8,10):=(A2*A1 *(A3 *A1 -6*A3 *A2*A1 *A4 + 4*A3 *A2 *A1 *A4  +
              2      2          2   2          2      2
         22*A3 *A2*A1 *A4 - 11*A3 *A1  - 12*A3*A2 *A1*A4  - 18*
                                      2   2
         A3*A2*A1*A4 + 12*A3*A1 + 4*A2 *A4  + 4*A2*A4 - 3))/ D

                             4   4      3      3          3   3     2
RI(8,11):=(2*A2*A1*(2*A3 *A1 -4*A3 *A2*A1 *A4 - 8*A3 *A1 +4*A3 *
             2   2   2       2      2          2   2           2
         A2 *A1 *A4  + 8*A3 *A2*A1 *A4 + 9*A3 *A1  - 12*A3*A2 *
              2                                   2   2
         A1*A4  + 8*A3*A2*A1*A4 - 5*A3*A1 + 4*A2 *A4  - 4*A2*A4
          + 1))/ D

                     5   5       4      4          4   4        3
RI(8,12):=(A2*(-2*A3 *A1  + 14*A3 *A2*A1 *A4 + 3*A3 *A1  - 20*A3 *
            2   3   2         3      3           3   3        2
          A2 *A1 *A4  - 38*A3 *A2*A1 *A4 + 10*A3 *A1  + 8*A3 *
            3   2   3         2   2   2   2       2      2
          A2 *A1 *A4  + 48*A3 *A2 *A1 *A4  + 8*A3 *A2*A1 *A4 -
              2   2           3      3           2      2
          13*A3 *A1  - 24*A3*A2 *A1*A4  + 16*A3*A2 *A1*A4  - 14
                                         3   3         2   2
          *A3*A2*A1*A4 + 6*A3*A1 + 8*A2 *A4  - 12*A2 *A4  + 6*
          A2*A4 - 1))/ D

             2          4   4        3      3          3   3       2
RI(9,1):=(A4 *(-15*A3 *A1  + 50*A3 *A2*A1 *A4 + 5*A3 *A1  - 52*A3
            2   2   2        2      2          2   2
          *A2 *A1 *A4  - 10*A3 *A2*A1 *A4 + 6*A3 *A1  + 16*A3*
            3      3          2      2                  3   3
          A2 *A1*A4  + 4*A3*A2 *A1*A4  - 3*A3*A1 - 8*A2 *A4  +
               2   2
          12*A2 *A4  - 6*A2*A4 + 1))/ D

                 2      4   4      3      3           3   3       2
RI(9,2):=(A3*A4 *(2*A3 *A1 -14*A3 *A2*A1 *A4 + 12*A3 *A1  + 20*A3
            2   2   2         2      2          2   2           3
        *A2 *A1 *A4  - 12*A3 *A2*A1 *A4 - 15*A3 *A1  - 8*A3*A2
              3          2      2
        *A1*A4  + 4*A3*A2 *A1*A4  + 2*A3*A2*A1*A4 + 7*A3*A1 + 8
            3   3          2   2
        *A2 *A4  - 20*A2 *A4  + 14*A2*A4 - 3))/ D
```

```
          2   2      3   3          2       2        2   2
RI(9,3):=(A3 *A4 *(9*A3 *A1  - 28*A3 *A2*A1 *A4 - A3 *A1 +24*A3*
           2         2                                  3   3
        A2 *A1*A4  + 6*A3*A2*A1*A4 - 9*A3*A1 - 8*A2 *A4  + 16*
           2   2
        A2 *A4  - 18*A2*A4 + 6))/ D

          3   2       3   3          2       2           2   2
RI(9,4):=(A3 *A4 *(-3*A3 *A1  + 20*A3 *A2*A1 *A4 - 21*A3 *A1 -24
               2        2                                    3   3
        *A3*A2 *A1*A4  + 18*A3*A2*A1*A4 + 27*A3*A1 + 8*A2 *A4
                 2   2
         - 16*A2 *A4  + 10*A2*A4 - 10))/ D

                    5   5          4       4          4   4         3
RI(9,5):=(A4*(-2*A3 *A1  + 14*A3 *A2*A1 *A4 + 3*A3 *A1  - 20*A3 *
            2   3   2          3       3              3   3        2
         A2 *A1 *A4  - 38*A3 *A2*A1 *A4 + 10*A3 *A1  + 8*A3 *
            3   2   3          2   2   2   2         2       2
         A2 *A1 *A4  + 48*A3 *A2 *A1 *A4  + 8*A3 *A2*A1 *A4 -
              2   2             3       3            2       2
         13*A3 *A1  - 24*A3*A2 *A1*A4  + 16*A3*A2 *A1*A4  - 14*
                                           3   3          2   2
         A3*A2*A1*A4 + 6*A3*A1 + 8*A2 *A4  - 12*A2 *A4  + 6*A2*
         A4 - 1))/ D

                             4   4      3       3          3   3       2
RI(9,6):=(2*A3*A4*(2*A3 *A1 -4*A3 *A2*A1 *A4 - 8*A3 *A1  + 4*A3 *
           2   2   2        2       2          2   2             2
        A2 *A1 *A4  + 8*A3 *A2*A1 *A4 + 9*A3 *A1  - 12*A3*A2 *
             2                                    2   2
        A1*A4  + 8*A3*A2*A1*A4 - 5*A3*A1 + 4*A2 *A4  - 4*A2*A4
         + 1))/ D

          2        4   4       3       3          2   2   2   2
RI(9,7):=(A3 *A4*(A3 *A1  - 6*A3 *A2*A1 *A4 + 4*A3 *A2 *A1 *A4 +22
            2       2         2   2           2       2
        *A3 *A2*A1 *A4 - 11*A3 *A1  - 12*A3*A2 *A1*A4  - 18*A3*
                                        2   2
        A2*A1*A4 + 12*A3*A1 + 4*A2 *A4  + 4*A2*A4 - 3))/ D

            3             3   3       2       2           2   2
RI(9,8):=(2*A3 *A4*( - 3*A3 *A1 +4*A3 *A2*A1 *A4 + 11*A3 *A1 -12
        *A3*A2*A1*A4 - 9*A3*A1 + 4*A2*A4 + 2))/ D

               5   5          4       4           4   4          3   2
RI(9,9):=(-2*A3 *A1  + 9*A3 *A2*A1 *A4 + 13*A3 *A1  - 38*A3 *A2 *
         3   2          3       3           3   3          2   3   2
      A1 *A4  + 17*A3 *A2*A1 *A4 - 28*A3 *A1  + 44*A3 *A2 *A1 *
         3           2   2   2   2          2       2          2   2
      A4  - 14*A3 *A2 *A1 *A4  - 30*A3 *A2*A1 *A4 + 23*A3 *A1
                      4          4          3       3             2       2
       - 16*A3*A2 *A1*A4  + 20*A3*A2 *A1*A4  - 24*A3*A2 *A1*A4
                                           4   4          3   3
       + 25*A3*A2*A1*A4 - 8*A3*A1 + 8*A2 *A4  - 20*A2 *A4  + 18*
         2   2
      A2 *A4  - 7*A2*A4 + 1)/ D
```

$$RI(9,10):=(A3*(A3^{5}*A1^{5}-10*A3^{4}*A2*A1^{4}*A4-7*A3^{4}*A1^{4}+30*A3^{3}*A2^{2}*$$
$$A1^{3}*A4^{2}+12*A3^{3}*A2*A1^{3}*A4+17*A3^{3}*A1^{3}-28*A3^{2}*A2^{3}*$$
$$A1^{2}*A4^{3}-32*A3^{2}*A2^{2}*A1^{2}*A4^{2}+9*A3^{2}*A2*A1^{2}*A4-17*$$
$$A3^{2}*A1^{2}+8*A3*A2^{4}*A1*A4^{4}+20*A3*A2^{3}*A1*A4^{3}+2*A3*$$
$$A2^{2}*A1*A4^{2}-13*A3*A2*A1*A4+7*A3*A1-8*A2^{4}*A4^{4}+$$
$$12*A2^{3}*A4^{3}-10*A2^{2}*A4^{2}+5*A2*A4-1))/\ D$$

$$RI(9,11):=(A3^{2}*(A3^{4}*A1^{4}-3*A3^{3}*A2*A1^{3}*A4-6*A3^{3}*A1^{3}+16*A3^{2}*A2^{2}*$$
$$A1^{2}*A4^{2}-13*A3^{2}*A2*A1^{2}*A4+11*A3^{2}*A1^{2}-24*A3*A2^{3}*A1$$
$$*A4^{3}+30*A3*A2^{2}*A1*A4^{2}+3*A3*A2*A1*A4-6*A3*A1+8*$$
$$A2^{4}*A4^{4}-16*A2^{3}*A4^{3}+6*A2^{2}*A4^{2}-2*A2*A4+1))/\ D$$

$$RI(9,12):=(A3^{3}*(-A3^{4}*A1^{4}+9*A3^{3}*A2*A1^{3}*A4+6*A3^{3}*A1^{3}-24*A3^{2}*$$
$$A2^{2}*A1^{2}*A4^{2}-9*A3^{2}*A2*A1^{2}*A4-11*A3^{2}*A1^{2}+24*A3*A2^{3}$$
$$*A1*A4^{3}-6*A3*A2^{2}*A1*A4^{2}+15*A3*A2*A1*A4+6*A3*A1$$
$$-8*A2^{4}*A4^{4}+16*A2^{3}*A4^{3}-14*A2^{2}*A4^{2}-2*A2*A4-1))$$
$$/\ D$$

$$RI(10,1):=(A1*A4^{2}*(2*A3^{4}*A1^{4}-14*A3^{3}*A2*A1^{3}*A4+12*A3^{3}*A1^{3}+20*$$
$$A3^{2}*A2^{2}*A1^{2}*A4^{2}-12*A3^{2}*A2*A1^{2}*A4-15*A3^{2}*A1^{2}-8*A3$$
$$*A2^{3}*A1*A4^{3}+4*A3*A2^{2}*A1*A4^{2}+2*A3*A2*A1*A4+7*A3*$$
$$A1+8*A2^{3}*A4^{3}-20*A2^{2}*A4^{2}+14*A2*A4-3))/\ D$$

$$RI(10,2):=(A4^{2}*(-6*A3^{4}*A1^{4}+22*A3^{3}*A2*A1^{3}*A4+4*A3^{3}*A1^{3}-28*A3^{2}$$
$$*A2^{2}*A1^{2}*A4^{2}-4*A3^{2}*A2*A1^{2}*A4-3*A3^{2}*A1^{2}+8*A3*A2^{3}*$$
$$A1*A4^{3}+20*A3*A2^{2}*A1*A4^{2}-18*A3*A2*A1*A4+3*A3*A1$$
$$-8*A2^{3}*A4^{3}+12*A2^{2}*A4^{2}-6*A2*A4+1))/\ D$$

$$RI(10,3):=(A3*A4^{2}*(-A3^{4}*A1^{4}+6*A3^{3}*A2*A1^{3}*A4-9*A3^{3}*A1^{3}-4*A3^{2}*$$
$$A2^{2}*A1^{2}*A4^{2}+6*A3^{2}*A2*A1^{2}*A4+12*A3^{2}*A1^{2}-12*A3*A2^{2}$$
$$*A1*A4^{2}+12*A3*A2*A1*A4-3*A3*A1+8*A2^{3}*A4^{3}-20*$$

```
           2    2
        A2 *A4   + 14*A2*A4 - 3))/ D

           2    2       3    3          2          2          2    2
RI(10,4):=(A3 *A4 *(9*A3 *A1   - 28*A3 *A2*A1 *A4 - A3 *A1 +24*A3*
           2          2                                   3    3
        A2 *A1*A4   + 6*A3*A2*A1*A4 - 9*A3*A1 - 8*A2 *A4   + 16*
           2    2
        A2 *A4   - 18*A2*A4 + 6))/ D

                          4    4       3          3          3    3       2
RI(10,5):=(2*A1*A4*(2*A3 *A1 -4*A3 *A2*A1 *A4 - 8*A3 *A1   + 4*A3 *
           2    2    2          2          2          2    2           2
        A2 *A1 *A4   + 8*A3 *A2*A1 *A4 + 9*A3 *A1   - 12*A3*A2 *
              2                                       2    2
        A1*A4   + 8*A3*A2*A1*A4 - 5*A3*A1 + 4*A2 *A4   - 4*A2*A4
         + 1))/ D

                      5    5       4          4          4    4       3    2
RI(10,6):=(A4*(-A3 *A1   + 8*A3 *A2*A1 *A4 + 3*A3 *A1   - 16*A3 *A2
             3    2       3          3       3    3        2    3    2
         *A1 *A4   - 16*A3 *A2*A1 *A4 - A3 *A1   + 8*A3 *A2 *A1
            3          2    2    2    2          2          2         2
         *A4   + 36*A3 *A2 *A1 *A4   - 10*A3 *A2*A1 *A4 - A3 *
           2           3          3          2          2
         A1   - 24*A3*A2 *A1*A4   + 20*A3*A2 *A1*A4   - 10*A3*A2*
                                  3    3          2    2
         A1*A4 + 3*A3*A1 + 8*A2 *A4   - 12*A2 *A4   + 6*A2*A4 -
         1))/ D

                         4    4       3    3          2    2    2    2
RI(10,7):=(2*A3*A4*(-A3 *A1   + 3*A3 *A1   + 4*A3 *A2 *A1 *A4   - 4*
           2          2                  2          2
        A3 *A2*A1 *A4 - 12*A3*A2 *A1*A4   + 12*A3*A2*A1*A4 - 3*
                          2    2
        A3*A1 + 4*A2 *A4   - 4*A2*A4 + 1))/ D

           2          4    4       3          3          2    2    2    2
RI(10,8):=(A3 *A4*(A3 *A1 -6*A3 *A2*A1 *A4 + 4*A3 *A2 *A1 *A4   +
              2          2          2    2          2          2
        22*A3 *A2*A1 *A4 - 11*A3 *A1   - 12*A3*A2 *A1*A4   - 18*
                                           2    2
        A3*A2*A1*A4 + 12*A3*A1 + 4*A2 *A4   + 4*A2*A4 - 3))/ D

                    5    5       4          4          4    4        3    2
RI(10,9):=(A1*(A3 *A1 -10*A3 *A2*A1 *A4 - 7*A3 *A1   + 30*A3 *A2 *
            3    2       3          3          3    3        2    3
         A1 *A4   + 12*A3 *A2*A1 *A4 + 17*A3 *A1   - 28*A3 *A2 *
            2    3       2    2    2    2         2          2
         A1 *A4   - 32*A3 *A2 *A1 *A4   + 9*A3 *A2*A1 *A4 - 17*
            2    2           4    4            3          3
         A3 *A1   + 8*A3*A2 *A1*A4   + 20*A3*A2 *A1*A4   + 2*A3*
            2          2                                    4    4
         A2 *A1*A4   - 13*A3*A2*A1*A4 + 7*A3*A1 - 8*A2 *A4   +
               3    3          2    2
         12*A2 *A4   - 10*A2 *A4   + 5*A2*A4 - 1))/ D
```

```
             5    5        4        4             4    4          3   2
RI(10,10):=(-A3 *A1  + 6*A3 *A2*A1 *A4 + 7*A3 *A1  - 22*A3 *A2 *
          3   2        3        3            3   3          2   3   2
        A1 *A4  + 4*A3 *A2*A1 *A4 - 17*A3 *A1  + 20*A3 *A2 *A1 *
          3           2    2   2   2           2          2             2   2
        A4  + 16*A3 *A2 *A1 *A4  - 27*A3 *A2*A1 *A4 + 17*A3 *A1
                    4      4          3         3              2       2
         - 8*A3*A2 *A1*A4  + 4*A3*A2 *A1*A4  - 18*A3*A2 *A1*A4
                                                   4   4          3    3
         + 23*A3*A2*A1*A4 - 7*A3*A1 + 8*A2 *A4  - 20*A2 *A4  +
            2    2
        18*A2 *A4  - 7*A2*A4 + 1)/ D

                   4        4           4    4          3    2    3   2
RI(10,11):=(A3*(-A3 *A2*A1 *A4 - A3 *A1  + 6*A3 *A2 *A1 *A4  + 3*
             3        3            3   3           2    3   2   3
          A3 *A2*A1 *A4 + 6*A3 *A1  - 4*A3 *A2 *A1 *A4  - 38*
            2    2   2   2           2          2              2   2
          A3 *A2 *A1 *A4  + 24*A3 *A2*A1 *A4 - 11*A3 *A1  + 36*
                3        3            2         2
          A3*A2 *A1*A4  - 12*A3*A2 *A1*A4  - 15*A3*A2*A1*A4 + 6
                          4    4          3   3           2   2
          *A3*A1 - 8*A2 *A4  + 12*A2 *A4  - 10*A2 *A4  + 5*A2*
          A4 - 1))/ D

             2     4    4        3         3            3    3          2    2
RI(10,12):=(A3 *(A3 *A1 -3*A3 *A2*A1 *A4 - 6*A3 *A1  + 16*A3 *A2 *
             2   2          2         2            2   2               3
          A1 *A4  - 13*A3 *A2*A1 *A4 + 11*A3 *A1  - 24*A3*A2 *
                3           2       2
          A1*A4  + 30*A3*A2 *A1*A4  + 3*A3*A2*A1*A4 - 6*A3*A1
                  4    4          3   3          2   2
           + 8*A2 *A4  - 16*A2 *A4  + 6*A2 *A4  - 2*A2*A4 + 1))
       / D

              2    2        3    3          2         2           2   2
RI(11,1):=(A1 *A4 *(9*A3 *A1  - 28*A3 *A2*A1 *A4 - A3 *A1 +24*A3*
             2       2                                     3   3
          A2 *A1*A4  + 6*A3*A2*A1*A4 - 9*A3*A1 - 8*A2 *A4  + 16*
             2   2
          A2 *A4  - 18*A2*A4 + 6))/ D

                 2      4    4        3         3            3   3          2
RI(11,2):=(A1*A4 *(-A3 *A1  + 6*A3 *A2*A1 *A4 - 9*A3 *A1  - 4*A3 *
             2   2   2         2         2            2   2               2
          A2 *A1 *A4  + 6*A3 *A2*A1 *A4 + 12*A3 *A1  - 12*A3*A2
                 2                                      3   3
          *A1*A4  + 12*A3*A2*A1*A4 - 3*A3*A1 + 8*A2 *A4  - 20*
             2   2
          A2 *A4  + 14*A2*A4 - 3))/ D

             2       4    4         3         3           3    3         2
RI(11,3):=(A4 *(-6*A3 *A1  + 22*A3 *A2*A1 *A4 + 4*A3 *A1  - 28*A3
             2    2   2        2         2           2   2             3
          *A2 *A1 *A4  - 4*A3 *A2*A1 *A4 - 3*A3 *A1  + 8*A3*A2 *
```

```
             3            2         2
        A1*A4   + 20*A3*A2 *A1*A4   - 18*A3*A2*A1*A4 + 3*A3*A1
                  3    3          2    2
         - 8*A2 *A4   + 12*A2 *A4   - 6*A2*A4 + 1))/ D

                   2       4    4          3       3             3    3
RI(11,4):=(A3*A4 *(2*A3 *A1   - 14*A3 *A2*A1 *A4 + 12*A3 *A1 +20*
             2    2    2    2           2       2            2    2
           A3 *A2 *A1 *A4   - 12*A3 *A2*A1 *A4 - 15*A3 *A1   - 8*A3
              3      3            2       2
           *A2 *A1*A4   + 4*A3*A2 *A1*A4   + 2*A3*A2*A1*A4 + 7*A3*
                     3    3           2    2
           A1 + 8*A2 *A4   - 20*A2 *A4   + 14*A2*A4 - 3))/ D

              2        4    4       3       3           2    2    2    2
RI(11,5):=(A1 *A4*(A3 *A1 -6*A3 *A2*A1 *A4 + 4*A3 *A2 *A1 *A4   +
                 2       2           2    2           2        2
           22*A3 *A2*A1 *A4 - 11*A3 *A1   - 12*A3*A2 *A1*A4   - 18*
                                                2    2
           A3*A2*A1*A4 + 12*A3*A1 + 4*A2 *A4   + 4*A2*A4 - 3))/ D

                               4    4        3    3          2    2    2    2
RI(11,6):=(2*A1*A4*(-A3 *A1   + 3*A3 *A1   + 4*A3 *A2 *A1 *A4   - 4*
             2       2             2       2
           A3 *A2*A1 *A4 - 12*A3*A2 *A1*A4   + 12*A3*A2*A1*A4 - 3*
                          2    2
           A3*A1 + 4*A2 *A4   - 4*A2*A4 + 1))/ D

                        5    5          4       4          4    4          3    2
RI(11,7):=(A4*(-A3 *A1   + 8*A3 *A2*A1 *A4 + 3*A3 *A1   - 16*A3 *A2
               3    2         3       3        3    3          2    3    2
            *A1 *A4   - 16*A3 *A2*A1 *A4 - A3 *A1   + 8*A3 *A2 *A1
               3          2    2    2    2          2       2           2
            *A4   + 36*A3 *A2 *A1 *A4   - 10*A3 *A2*A1 *A4 - A3 *
              2            3        3            2        2
            A1   - 24*A3*A2 *A1*A4   + 20*A3*A2 *A1*A4   - 10*A3*A2*
                                      3    3          2    2
            A1*A4 + 3*A3*A1 + 8*A2 *A4   - 12*A2 *A4   + 6*A2*A4 -
            1))/ D

                                 4    4       3       3           3    3          2
RI(11,8):=(2*A3*A4*(2*A3 *A1 -4*A3 *A2*A1 *A4 - 8*A3 *A1   + 4*A3 *
             2    2    2          2       2           2    2          2
           A2 *A1 *A4   + 8*A3 *A2*A1 *A4 + 9*A3 *A1   - 12*A3*A2 *
               2                                          2    2
           A1*A4   + 8*A3*A2*A1*A4 - 5*A3*A1 + 4*A2 *A4   - 4*A2*A4
            + 1))/ D

              2      4    4        3       3           3    3           2    2
RI(11,9):=(A1 *(A3 *A1 -3*A3 *A2*A1 *A4 - 6*A3 *A1   + 16*A3 *A2 *
             2    2           2        2            2    2           3
           A1 *A4   - 13*A3 *A2*A1 *A4 + 11*A3 *A1   - 24*A3*A2 *A1
              3              2        2
           *A4   + 30*A3*A2 *A1*A4   + 3*A3*A2*A1*A4 - 6*A3*A1 + 8*
             4    4          3    3         2    2
           A2 *A4   - 16*A2 *A4   + 6*A2 *A4   - 2*A2*A4 + 1))/ D
```

```
                   4      4        4   4        3   2   3   2
RI(11,10):=(A1*(-A3 *A2*A1 *A4 - A3 *A1   + 6*A3 *A2 *A1 *A4   + 3*
              3      3          3   3        2   3   2   3
            A3 *A2*A1 *A4 + 6*A3 *A1   - 4*A3 *A2 *A1 *A4   - 38*
              2   2   2   2         2      2           2   2
            A3 *A2 *A1 *A4   + 24*A3 *A2*A1 *A4 - 11*A3 *A1   + 36*
                 3      3            2      2
            A3*A2 *A1*A4   - 12*A3*A2 *A1*A4   - 15*A3*A2*A1*A4 + 6
                         4   4         3   3         2   2
            *A3*A1 - 8*A2 *A4   + 12*A2 *A4   - 10*A2 *A4   + 5*A2*
            A4 - 1))/ D

               5   5        4      4          4   4         3   2
RI(11,11):=(-A3 *A1   + 6*A3 *A2*A1 *A4 + 7*A3 *A1   - 22*A3 *A2 *
           3   2        3      3           3   3         2   3   2
         A1 *A4   + 4*A3 *A2*A1 *A4 - 17*A3 *A1   + 20*A3 *A2 *A1 *
           3         2   2   2   2         2      2           2   2
         A4   + 16*A3 *A2 *A1 *A4   - 27*A3 *A2*A1 *A4 + 17*A3 *A1
                   4      4           3      3            2      2
          - 8*A3*A2 *A1*A4   + 4*A3*A2 *A1*A4   - 18*A3*A2 *A1*A4
                                           4   4         3   3
          + 23*A3*A2*A1*A4 - 7*A3*A1 + 8*A2 *A4   - 20*A2 *A4   +
              2   2
         18*A2 *A4   - 7*A2*A4 + 1)/ D

                  5   5        4      4          4   4        3   2
RI(11,12):=(A3*(A3 *A1   -10*A3 *A2*A1 *A4 - 7*A3 *A1   +30*A3 *A2 *
              3   2         3      3           3   3         2   3
            A1 *A4   + 12*A3 *A2*A1 *A4 + 17*A3 *A1   - 28*A3 *A2 *
              2   3         2   2   2   2        2      2
            A1 *A4   - 32*A3 *A2 *A1 *A4   + 9*A3 *A2*A1 *A4 - 17*
              2   2           4      4            3      3
            A3 *A1   + 8*A3*A2 *A1*A4   + 20*A3*A2 *A1*A4   + 2*A3*
              2      2                                   4   4
            A2 *A1*A4   - 13*A3*A2*A1*A4 + 7*A3*A1 - 8*A2 *A4   +
                 3   3         2   2
            12*A2 *A4   - 10*A2 *A4   + 5*A2*A4 - 1))/ D

             3   2         3   3         2      2           2   2
RI(12,1):=(A1 *A4 *( - 3*A3 *A1   + 20*A3 *A2*A1 *A4 - 21*A3 *A1   -
                    2      2                                    3
            24*A3*A2 *A1*A4   + 18*A3*A2*A1*A4 + 27*A3*A1 + 8*A2 *
              3         2   2
            A4   - 16*A2 *A4   + 10*A2*A4 - 10))/ D

             2   2      3   3         2      2        2   2
RI(12,2):=(A1 *A4 *(9*A3 *A1   - 28*A3 *A2*A1 *A4 - A3 *A1 +24*A3*
              2      2                                   3   3
            A2 *A1*A4   + 6*A3*A2*A1*A4 - 9*A3*A1 - 8*A2 *A4   + 16*
              2   2
            A2 *A4   - 18*A2*A4 + 6))/ D

                2      4   4         3      3           3   3
RI(12,3):=(A1*A4 *(2*A3 *A1   - 14*A3 *A2*A1 *A4 + 12*A3 *A1   +20*
              2   2   2   2         2      2           2   2
            A3 *A2 *A1 *A4   - 12*A3 *A2*A1 *A4 - 15*A3 *A1   - 8*A3
```

```
              3      3           2      2
           *A2 *A1*A4  + 4*A3*A2 *A1*A4  + 2*A3*A2*A1*A4 + 7*A3*
                    3   3          2   2
           A1 + 8*A2 *A4  - 20*A2 *A4  + 14*A2*A4 - 3))/ D

             2          4   4         3      3          3   3
RI(12,4):=(A4 *( - 15*A3 *A1  + 50*A3 *A2*A1 *A4 + 5*A3 *A1 -52*
             2   2   2   2          2      2          2   2
           A3 *A2 *A1 *A4  - 10*A3 *A2*A1 *A4 + 6*A3 *A1  + 16*A3
              3      3           2      2                 3   3
           *A2 *A1*A4  + 4*A3*A2 *A1*A4  - 3*A3*A1 - 8*A2 *A4  +
                2   2
           12*A2 *A4  - 6*A2*A4 + 1))/ D

               3          3   3        2      2           2   2
RI(12,5):=(2*A1 *A4*(-3*A3 *A1  + 4*A3 *A2*A1 *A4 + 11*A3 *A1  -
           12*A3*A2*A1*A4 - 9*A3*A1 + 4*A2*A4 + 2))/ D

             2       4   4         3      3          2   2   2   2
RI(12,6):=(A1 *A4*(A3 *A1  - 6*A3 *A2*A1 *A4 + 4*A3 *A2 *A1 *A4 +
                2      2            2   2          2         2
           22*A3 *A2*A1 *A4 - 11*A3 *A1  - 12*A3*A2 *A1*A4  - 18*
                                            2   2
           A3*A2*A1*A4 + 12*A3*A1 + 4*A2 *A4  + 4*A2*A4 - 3))/ D

                            4   4      3      3         3   3     2
RI(12,7):=(2*A1*A4*(2*A3 *A1 -4*A3 *A2*A1 *A4 - 8*A3 *A1 +4*A3 *
             2   2   2         2      2         2   2            2
           A2 *A1 *A4  + 8*A3 *A2*A1 *A4 + 9*A3 *A1  - 12*A3*A2 *
                 2                                    2   2
           A1*A4  + 8*A3*A2*A1*A4 - 5*A3*A1 + 4*A2 *A4  - 4*A2*A4
            + 1))/ D

                        5   5         4      4          4   4        3
RI(12,8):=(A4*(-2*A3 *A1  + 14*A3 *A2*A1 *A4 + 3*A3 *A1  - 20*A3 *
              2   3   2          3      3          3   3         2
            A2 *A1 *A4  - 38*A3 *A2*A1 *A4 + 10*A3 *A1  + 8*A3 *
              3   2   3           2   2   2   2         2      2
            A2 *A1 *A4  + 48*A3 *A2 *A1 *A4  + 8*A3 *A2*A1 *A4 -
                 2   2             3      3             2      2
            13*A3 *A1  - 24*A3*A2 *A1*A4  + 16*A3*A2 *A1*A4  - 14
                                           3   3          2   2
            *A3*A2*A1*A4 + 6*A3*A1 + 8*A2 *A4  - 12*A2 *A4  + 6*
            A2*A4 - 1))/ D

              3     4   4         3      3          3   3          2
RI(12,9):=(A1 *(-A3 *A1  + 9*A3 *A2*A1 *A4 + 6*A3 *A1  - 24*A3 *
              2   2   2         2      2           2   2            3
           A2 *A1 *A4  - 9*A3 *A2*A1 *A4 - 11*A3 *A1  + 24*A3*A2
                  3           2      2
           *A1*A4  - 6*A3*A2 *A1*A4  + 15*A3*A2*A1*A4 + 6*A3*A1
                 4   4          3   3          2   2
            - 8*A2 *A4  + 16*A2 *A4  - 14*A2 *A4  - 2*A2*A4 - 1))
           / D
```

```
              2    4   4     3      3          3   3        2   2
RI(12,1Ø):=(A1 *(A3 *A1 -3*A3 *A2*A1 *A4 - 6*A3 *A1  + 16*A3 *A2 *
              2   2        2      2           2   2           3
            A1 *A4  - 13*A3 *A2*A1 *A4 + 11*A3 *A1  - 24*A3*A2 *
                 3           2      2
            A1*A4  + 3Ø*A3*A2 *A1*A4  + 3*A3*A2*A1*A4 - 6*A3*A1
                   4   4        3   3       2   2
             + 8*A2 *A4  - 16*A2 *A4  + 6*A2 *A4  - 2*A2*A4 + 1))
            / D

                  5   5      4      4          4   4      3   2
RI(12,11):=(A1*(A3 *A1 -1Ø*A3 *A2*A1 *A4 - 7*A3 *A1 +3Ø*A3 *A2 *
              3   2        3      3           3   3        2   3
            A1 *A4  + 12*A3 *A2*A1 *A4 + 17*A3 *A1  - 28*A3 *A2 *
              2   3        2   2   2   2       2      2
            A1 *A4  - 32*A3 *A2 *A1 *A4  + 9*A3 *A2*A1 *A4 - 17*
              2   2          4      4           3      3
            A3 *A1  + 8*A3*A2 *A1*A4  + 2Ø*A3*A2 *A1*A4  + 2*A3*
              2      2                                  4   4
            A2 *A1*A4  - 13*A3*A2*A1*A4 + 7*A3*A1 - 8*A2 *A4  +
                 3   3        2   2
            12*A2 *A4  - 1Ø*A2 *A4  + 5*A2*A4 - 1))/ D

                 5   5       4      4           4   4        3   2
RI(12,12):=(-2*A3 *A1  + 9*A3 *A2*A1 *A4 + 13*A3 *A1  - 38*A3 *A2
            3   2        3      3           3   3        2   3
         *A1 *A4  + 17*A3 *A2*A1 *A4 - 28*A3 *A1  + 44*A3 *A2 *
           2   3        2   2   2   2        2      2           2
         A1 *A4  - 14*A3 *A2 *A1 *A4  - 3Ø*A3 *A2*A1 *A4 + 23*A3
            2           4      4           3      3           2
         *A1  - 16*A3*A2 *A1*A4  + 2Ø*A3*A2 *A1*A4  - 24*A3*A2 *
              2                                  4   4        3
         A1*A4  + 25*A3*A2*A1*A4 - 8*A3*A1 + 8*A2 *A4  - 2Ø*A2 *
           3        2   2
         A4  + 18*A2 *A4  - 7*A2*A4 + 1)/ D
```

APPENDIX B

SYMBOLIC INVERSION OF THE (4 X 4) POINTS BLOCK MATRIX (4.2.36b)

RI(i,j), (i=1,16 and j=1,16), are the elements of the matrix $(R0)^{-1}$, (see page 166), and D = det(R0). $A=\alpha$.

$$D := 400 A^{12} - 1260 A^{10} + 1481 A^{8} - 804 A^{6} + 206 A^{4} - 24 A^{2} + 1$$

$$RI(1,1) := (120 A^{12} - 544 A^{10} + 839 A^{8} - 562 A^{6} + 168 A^{4} - 22 A^{2} + 1)/D$$

$$RI(1,2) := (A (140 A^{10} - 358 A^{8} + 321 A^{6} - 121 A^{4} + 19 A^{2} - 1))/D$$

$$RI(1,3) := (A^{2} (-40 A^{10} + 118 A^{8} - 137 A^{6} + 73 A^{4} - 15 A^{2} + 1))/D$$

$$RI(1,4) := (A^{3} (25 A^{6} - 35 A^{4} + 11 A^{2} - 1))/D$$

$$RI(1,5) := (A (140 A^{10} - 358 A^{8} + 321 A^{6} - 121 A^{4} + 19 A^{2} - 1))/D$$

$$RI(1,6) := (2 A^{2} (-40 A^{10} + 143 A^{8} - 172 A^{6} + 84 A^{4} - 16 A^{2} + 1))/D$$

$$RI(1,7) := (A^{3} (-100 A^{8} + 215 A^{6} - 149 A^{4} + 37 A^{2} - 3))/D$$

$$RI(1,8) := (2 A^{4} (20 A^{8} - 59 A^{6} + 56 A^{4} - 19 A^{2} + 2))/D$$

$$RI(1,9) := (A^{2} (-40 A^{10} + 118 A^{8} - 137 A^{6} + 73 A^{4} - 15 A^{2} + 1))/D$$

$$RI(1,10) := (A^{3} (-100 A^{8} + 215 A^{6} - 149 A^{4} + 37 A^{2} - 3))/D$$

$$RI(1,11) := (2 A^{4} (40 A^{8} - 93 A^{6} + 77 A^{4} - 27 A^{2} + 3))/D$$

$$RI(1,12) := (2 A^{5} (30 A^{6} - 61 A^{4} + 36 A^{2} - 5))/D$$

$$RI(1,13) := (A^{3} (25 A^{6} - 35 A^{4} + 11 A^{2} - 1))/D$$

$$RI(1,14) := (2 A^{4} (20 A^{8} - 59 A^{6} + 56 A^{4} - 19 A^{2} + 2))/D$$

$$RI(1,15) := (2 A^{5} (30 A^{6} - 61 A^{4} + 36 A^{2} - 5))/D$$

$$RI(1,16) := (4 A^{6} (-30 A^{6} + 61 A^{4} - 36 A^{2} + 5))/D$$

$$RI(2,1) := (A (140 A^{10} - 358 A^{8} + 321 A^{6} - 121 A^{4} + 19 A^{2} - 1))/D$$

$$RI(2,2) := (80 A^{12} - 426 A^{10} + 702 A^{8} - 489 A^{6} + 153 A^{4} - 21 A^{2} + 1)/D$$

$$RI(2,3) := (A (140 A^{10} - 333 A^{8} + 286 A^{6} - 110 A^{4} + 18 A^{2} - 1))/D$$

$$RI(2,4) := (A^{2} (-40 A^{10} + 118 A^{8} - 137 A^{6} + 73 A^{4} - 15 A^{2} + 1))/D$$

$$RI(2,5) := (2 A^{2} (-40 A^{10} + 143 A^{8} - 172 A^{6} + 84 A^{4} - 16 A^{2} + 1))/D$$

$$RI(2,6) := (A (40 A^{10} - 143 A^{8} + 172 A^{6} - 84 A^{4} + 16 A^{2} - 1))/D$$

$$RI(2,7) := (2 A^{2} (-20 A^{10} + 84 A^{8} - 116 A^{6} + 65 A^{4} - 14 A^{2} + 1))/D$$

$$RI(2,8) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$$

$$RI(2,9) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$$

$$RI(2,10) := (A^{2}*(40*A^{10} - 68*A^{8} + 17*A^{6} + 19*A^{4} - 9*A^{2} +1))/D$$

$$RI(2,11) := (A^{3}*(- 40*A^{8} + 93*A^{6} - 77*A^{4} + 27*A^{2} - 3))/ D$$

$$RI(2,12) := (2*A^{4}*(40*A^{8} - 93*A^{6} + 77*A^{4} - 27*A^{2} + 3))/ D$$

$$RI(2,13) := (2*A^{4}*(20*A^{8} - 59*A^{6} + 56*A^{4} - 19*A^{2} + 2))/ D$$

$$RI(2,14) := (A^{3}*(60*A^{8} - 97*A^{6} + 37*A^{4} + A^{2} - 1))/ D$$

$$RI(2,15) := (2*A^{4}*(- 40*A^{8} + 63*A^{6} - 16*A^{4} - 9*A^{2} + 2))/D$$

$$RI(2,16) := (2*A^{5}*(30*A^{6} - 61*A^{4} + 36*A^{2} - 5))/ D$$

$$RI(3,1) := (A^{2}*(- 40*A^{10} + 118*A^{8} - 137*A^{6} + 73*A^{4} - 15*A^{2} +1))/D$$

$$RI(3,2) := (A*(140*A^{10} - 333*A^{8} + 286*A^{6} - 110*A^{4} + 18*A^{2} -1))/D$$

$$RI(3,3) := (80*A^{12} - 426*A^{10} + 702*A^{8} - 489*A^{6} + 153*A^{4} - 21*A^{2} +1)/D$$

$$RI(3,4) := (A*(140*A^{10} - 358*A^{8} + 321*A^{6} - 121*A^{4} + 19*A^{2} - 1))/D$$

$$RI(3,5) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$$

$$RI(3,6) := (2*A^{2}*(-20*A^{10} + 84*A^{8} - 116*A^{6} + 65*A^{4} - 14*A^{2} +1))/D$$

$$RI(3,7) := (A*(40*A^{10} - 143*A^{8} + 172*A^{6} - 84*A^{4} + 16*A^{2} - 1))/ D$$

$$RI(3,8) := (2*A^{2}*(-40*A^{10} + 143*A^{8} - 172*A^{6} + 84*A^{4} - 16*A^{2} +1))/D$$

$$RI(3,9) := (2*A^{4}*(40*A^{8} - 93*A^{6} + 77*A^{4} - 27*A^{2} + 3))/ D$$

$$RI(3,10) := (A^{3}*(- 40*A^{8} + 93*A^{6} - 77*A^{4} + 27*A^{2} - 3))/ D$$

$$RI(3,11) := (A^{2}*(40*A^{10} - 68*A^{8} + 17*A^{6} + 19*A^{4} - 9*A^{2} +1))/ D$$

$$RI(3,12) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$$

$$RI(3,13) := (2*A^{5}*(30*A^{6} - 61*A^{4} + 36*A^{2} - 5))/ D$$

$$RI(3,14) := (2*A^{4}*(- 40*A^{8} + 63*A^{6} - 16*A^{4} - 9*A^{2} + 2))/ D$$

$$RI(3,15) := (A^{3}*(60*A^{8} - 97*A^{6} + 37*A^{4} + A^{2} - 1))/ D$$

$$RI(3,16) := (2*A^{4}*(20*A^{8} - 59*A^{6} + 56*A^{4} - 19*A^{2} + 2))/ D$$

$$RI(4,1) := (A^{3}*(25*A^{6} - 35*A^{4} + 11*A^{2} - 1))/ D$$

$$RI(4,2) := (A^{2}*(-40*A^{10} + 118*A^{8} - 137*A^{6} + 73*A^{4} - 15*A^{2} +1))/D$$

$$RI(4,3) := (A*(140*A^{10} - 358*A^{8} + 321*A^{6} - 121*A^{4} + 19*A^{2} -1))/D$$

$$RI(4,4) := (120*A^{12} - 544*A^{10} + 839*A^{8} - 562*A^{6} + 168*A^{4} - 22*A^{2} + 1)/D$$

$$RI(4,5) := (2*A^{4}*(20*A^{8} - 59*A^{6} + 56*A^{4} - 19*A^{2} + 2))/ D$$

$$RI(4,6) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$$

$$RI(4,7) := (2*A^{2}*(-40*A^{10} + 143*A^{8} - 172*A^{6} + 84*A^{4} - 16*A^{2} + 1))/D$$

$$RI(4,8) := (A*(140*A^{10} - 358*A^{8} + 321*A^{6} - 121*A^{4} + 19*A^{2} - 1))/D$$

$$RI(4,9) := (2*A^{5}*(30*A^{6} - 61*A^{4} + 36*A^{2} - 5))/ D$$

$$RI(4,10) := (2*A^{4}*(40*A^{8} - 93*A^{6} + 77*A^{4} - 27*A^{2} + 3))/ D$$

$$RI(4,11) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$$

$$RI(4,12) := (A^{2}*(-40*A^{10} + 118*A^{8} - 137*A^{6} + 73*A^{4} - 15*A^{2} + 1))/D$$

$$RI(4,13) := (4*A^{6}*(- 30*A^{6} + 61*A^{4} - 36*A^{2} + 5))/ D$$

$$RI(4,14) := (2*A^{5}*(30*A^{6} - 61*A^{4} + 36*A^{2} - 5))/ D$$

$$RI(4,15) := (2*A^{4}*(20*A^{8} - 59*A^{6} + 56*A^{4} - 19*A^{2} + 2))/ D$$

$$RI(4,16) := (A^{3}*(25*A^{6} - 35*A^{4} + 11*A^{2} - 1))/ D$$

$$RI(5,1) := (A*(140*A^{10} - 358*A^{8} + 321*A^{6} - 121*A^{4} + 19*A^{2} - 1))/D$$

$$RI(5,2) := (2*A^{2}*(-40*A^{10} + 143*A^{8} - 172*A^{6} + 84*A^{4} - 16*A^{2} + 1))/D$$

$$RI(5,3) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$$

$$RI(5,4) := (2*A^{4}*(20*A^{8} - 59*A^{6} + 56*A^{4} - 19*A^{2} + 2))/ D$$

$$RI(5,5) := (80*A^{12} - 426*A^{10} + 702*A^{8} - 489*A^{6} + 153*A^{4} - 21*A^{2} + 1)/D$$

$$RI(5,6) := (A*(40*A^{10} - 143*A^{8} + 172*A^{6} - 84*A^{4} + 16*A^{2} - 1))/D$$

$$RI(5,7) := (A^{2}*(40*A^{10} - 68*A^{8} + 17*A^{6} + 19*A^{4} - 9*A^{2} + 1))/ D$$

$$RI(5,8) := (A^{3}*(60*A^{8} - 97*A^{6} + 37*A^{4} + A^{2} - 1))/ D$$

$$RI(5,9) := (A*(140*A^{10} - 333*A^{8} + 286*A^{6} - 110*A^{4} + 18*A^{2} - 1))/D$$

$$RI(5,10) := (2*A^{2}*(-20*A^{10} + 84*A^{8} - 116*A^{6} + 65*A^{4} - 14*A^{2} + 1))/D$$

$$RI(5,11) := (A^{3}*(- 40*A^{8} + 93*A^{6} - 77*A^{4} + 27*A^{2} - 3))/ D$$

$$RI(5,12) := (2*A^{4}*(- 40*A^{8} + 63*A^{6} - 16*A^{4} - 9*A^{2} + 2))/ D$$

$$RI(5,13) := (A^{2}*(-40*A^{10} + 118*A^{8} - 137*A^{6} + 73*A^{4} - 15*A^{2} + 1))/D$$

$$RI(5,14) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$$

$$RI(5,15) := (2*A^{4}*(40*A^{8} - 93*A^{6} + 77*A^{4} - 27*A^{2} + 3))/ D$$

$$RI(5,16) := (2*A^{5}*(30*A^{6} - 61*A^{4} + 36*A^{2} - 5))/ D$$

$$RI(6,1) := (2*A^{2}*(-40*A^{10} + 143*A^{8} - 172*A^{6} + 84*A^{4} - 16*A^{2} +1))/D$$

$$RI(6,2) := (A*(40*A^{10} - 143*A^{8} + 172*A^{6} - 84*A^{4} + 16*A^{2} - 1))/ D$$

$$RI(6,3) := (2*A^{2}*(-20*A^{10} + 84*A^{8} - 116*A^{6} + 65*A^{4} - 14*A^{2} +1))/D$$

$$RI(6,4) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$$

$$RI(6,5) := (A*(40*A^{10} - 143*A^{8} + 172*A^{6} - 84*A^{4} + 16*A^{2} - 1))/ D$$

$$RI(6,6) := (120*A^{12} - 494*A^{10} + 719*A^{8} - 470*A^{6} + 144*A^{4} - 20*A^{2} +1)/D$$

$$RI(6,7) := (A*(100*A^{10} - 240*A^{8} + 209*A^{6} - 83*A^{4} + 15*A^{2} - 1))/D$$

$$RI(6,8) := (A^{2}*(40*A^{10} - 68*A^{8} + 17*A^{6} + 19*A^{4} - 9*A^{2} +1))/ D$$

$$RI(6,9) := (2*A^{2}*(-20*A^{10} + 84*A^{8} - 116*A^{6} + 65*A^{4} - 14*A^{2} +1))/D$$

$$RI(6,10) := (A*(100*A^{10} - 240*A^{8} + 209*A^{6} - 83*A^{4} + 15*A^{2} - 1))/D$$

$$RI(6,11) := (2*A^{2}*(-60*A^{10} + 147*A^{8} - 132*A^{6} + 56*A^{4} - 12*A^{2} +1))/D$$

$$RI(6,12) := (A^{3}*(- 40*A^{8} + 93*A^{6} - 77*A^{4} + 27*A^{2} - 3))/ D$$

$$RI(6,13) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$$

$$RI(6,14) := (A^{2}*(40*A^{10} - 68*A^{8} + 17*A^{6} + 19*A^{4} - 9*A^{2} +1))/ D$$

$$RI(6,15) := (A^{3}*(- 40*A^{8} + 93*A^{6} - 77*A^{4} + 27*A^{2} - 3))/ D$$

$$RI(6,16) := (2*A^{4}*(40*A^{8} - 93*A^{6} + 77*A^{4} - 27*A^{2} + 3))/ D$$

$$RI(7,1) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$$

$$RI(7,2) := (2*A^{2}*(-20*A^{10} + 84*A^{8} - 116*A^{6} + 65*A^{4} - 14*A^{2} +1))/D$$

$$RI(7,3) := (A*(40*A^{10} - 143*A^{8} + 172*A^{6} - 84*A^{4} + 16*A^{2} - 1))/ D$$

$$RI(7,4) := (2*A^{2}*(-40*A^{10} + 143*A^{8} - 172*A^{6} + 84*A^{4} - 16*A^{2} +1))/D$$

$$RI(7,5) := (A^{2}*(40*A^{10} - 68*A^{8} + 17*A^{6} + 19*A^{4} - 9*A^{2} +1))/ D$$

$$RI(7,6) := (A*(100*A^{10} - 240*A^{8} + 209*A^{6} - 83*A^{4} + 15*A^{2} - 1))/D$$

$$RI(7,7) := (120*A^{12} - 494*A^{10} + 719*A^{8} - 470*A^{6} + 144*A^{4} - 20*A^{2} +1)/D$$

$$RI(7,8) := (A*(40*A^{10} - 143*A^{8} + 172*A^{6} - 84*A^{4} + 16*A^{2} - 1))/ D$$

$$RI(7,9) := (A^{3}*(- 40*A^{8} + 93*A^{6} - 77*A^{4} + 27*A^{2} - 3))/ D$$

$$RI(7,10) := (2*A^{2}*(-60*A^{10} + 147*A^{8} - 132*A^{6} + 56*A^{4} - 12*A^{2} +1))/D$$

$$RI(7,11) := (A*(100*A^{10} - 240*A^{8} + 209*A^{6} - 83*A^{4} + 15*A^{2} - 1))/D$$

$$\mathrm{RI}(7,12) := (2*A^{2}*(-20*A^{10} + 84*A^{8} - 116*A^{6} + 65*A^{4} - 14*A^{2} + 1))/D$$

$$\mathrm{RI}(7,13) := (2*A^{4}*(40*A^{8} - 93*A^{6} + 77*A^{4} - 27*A^{2} + 3))/D$$

$$\mathrm{RI}(7,14) := (A^{3}*(- 40*A^{8} + 93*A^{6} - 77*A^{4} + 27*A^{2} - 3))/D$$

$$\mathrm{RI}(7,15) := (A^{2}*(40*A^{10} - 68*A^{8} + 17*A^{6} + 19*A^{4} - 9*A^{2} + 1))/D$$

$$\mathrm{RI}(7,16) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/D$$

$$\mathrm{RI}(8,1) := (2*A^{4}*(20*A^{8} - 59*A^{6} + 56*A^{4} - 19*A^{2} + 2))/D$$

$$\mathrm{RI}(8,2) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/D$$

$$\mathrm{RI}(8,3) := (2*A^{2}*(-40*A^{10} + 143*A^{8} - 172*A^{6} + 84*A^{4} - 16*A^{2} + 1))/D$$

$$\mathrm{RI}(8,4) := (A*(140*A^{10} - 358*A^{8} + 321*A^{6} - 121*A^{4} + 19*A^{2} - 1))/D$$

$$\mathrm{RI}(8,5) := (A^{3}*(60*A^{8} - 97*A^{6} + 37*A^{4} + A^{2} - 1))/D$$

$$\mathrm{RI}(8,6) := (A^{2}*(40*A^{10} - 68*A^{8} + 17*A^{6} + 19*A^{4} - 9*A^{2} + 1))/D$$

$$\mathrm{RI}(8,7) := (A*(40*A^{10} - 143*A^{8} + 172*A^{6} - 84*A^{4} + 16*A^{2} - 1))/D$$

$$\mathrm{RI}(8,8) := (80*A^{12} - 426*A^{10} + 702*A^{8} - 489*A^{6} + 153*A^{4} - 21*A^{2} + 1)/D$$

$$\mathrm{RI}(8,9) := (2*A^{4}*(- 40*A^{8} + 63*A^{6} - 16*A^{4} - 9*A^{2} + 2))/D$$

$$\mathrm{RI}(8,10) := (A^{3}*(- 40*A^{8} + 93*A^{6} - 77*A^{4} + 27*A^{2} - 3))/D$$

$$\mathrm{RI}(8,11) := (2*A^{2}*(-20*A^{10} + 84*A^{8} - 116*A^{6} + 65*A^{4} - 14*A^{2} + 1))/D$$

$$\mathrm{RI}(8,12) := (A*(140*A^{10} - 333*A^{8} + 286*A^{6} - 110*A^{4} + 18*A^{2} - 1))/D$$

$$\mathrm{RI}(8,13) := (2*A^{5}*(30*A^{6} - 61*A^{4} + 36*A^{2} - 5))/D$$

$$\mathrm{RI}(8,14) := (2*A^{4}*(40*A^{8} - 93*A^{6} + 77*A^{4} - 27*A^{2} + 3))/D$$

$$\mathrm{RI}(8,15) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/D$$

$$\mathrm{RI}(8,16) := (A^{2}*(-40*A^{10} + 118*A^{8} - 137*A^{6} + 73*A^{4} - 15*A^{2} + 1))/D$$

$$\mathrm{RI}(9,1) := (A^{2}*(- 40*A^{10} + 118*A^{8} - 137*A^{6} + 73*A^{4} - 15*A^{2} + 1))/D$$

$$\mathrm{RI}(9,2) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/D$$

$$\mathrm{RI}(9,3) := (2*A^{4}*(40*A^{8} - 93*A^{6} + 77*A^{4} - 27*A^{2} + 3))/D$$

$$\mathrm{RI}(9,4) := (2*A^{5}*(30*A^{6} - 61*A^{4} + 36*A^{2} - 5))/D$$

$$\mathrm{RI}(9,5) := (A*(140*A^{10} - 333*A^{8} + 286*A^{6} - 110*A^{4} + 18*A^{2} - 1))/D$$

$$\mathrm{RI}(9,6) := (2*A^{2}*(-20*A^{10} + 84*A^{8} - 116*A^{6} + 65*A^{4} - 14*A^{2} + 1))/D$$

$$\mathrm{RI}(9,7) := (A^{3}*(- 40*A^{8} + 93*A^{6} - 77*A^{4} + 27*A^{2} - 3))/D$$

$RI(9,8) := (2*A^{4}*(- 40*A^{8} + 63*A^{6} - 16*A^{4} - 9*A^{2} + 2))/ D$

$RI(9,9) := (80*A^{12} - 426*A^{10} + 702*A^{8} - 489*A^{6} + 153*A^{4} - 21*A^{2} +1)/D$

$RI(9,10) := (A*(40*A^{10} - 143*A^{8} + 172*A^{6} - 84*A^{4} + 16*A^{2} -1))/D$

$RI(9,11) := (A^{2}*(40*A^{10} - 68*A^{8} + 17*A^{6} + 19*A^{4} - 9*A^{2} +1))/ D$

$RI(9,12) := (A^{3}*(60*A^{8} - 97*A^{6} + 37*A^{4} + A^{2} - 1))/ D$

$RI(9,13) := (A*(140*A^{10} - 358*A^{8} + 321*A^{6} - 121*A^{4} + 19*A^{2} -1))/D$

$RI(9,14) := (2*A^{2}*(-40*A^{10} + 143*A^{8} - 172*A^{6} + 84*A^{4} - 16*A^{2} +1))/D$

$RI(9,15) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$

$RI(9,16) := (2*A^{4}*(20*A^{8} - 59*A^{6} + 56*A^{4} - 19*A^{2} + 2))/ D$

$RI(10,1) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$

$RI(10,2) := (A^{2}*(40*A^{10} - 68*A^{8} + 17*A^{6} + 19*A^{4} - 9*A^{2} +1))/ D$

$RI(10,3) := (A^{3}*(- 40*A^{8} + 93*A^{6} - 77*A^{4} + 27*A^{2} - 3))/ D$

$RI(10,4) := (2*A^{4}*(40*A^{8} - 93*A^{6} + 77*A^{4} - 27*A^{2} + 3))/ D$

$RI(10,5) := (2*A^{2}*(-20*A^{10} + 84*A^{8} - 116*A^{6} + 65*A^{4} - 14*A^{2} +1))/D$

$RI(10,6) := (A*(100*A^{10} - 240*A^{8} + 209*A^{6} - 83*A^{4} + 15*A^{2} - 1))/D$

$RI(10,7) := (2*A^{2}*(-60*A^{10} + 147*A^{8} - 132*A^{6} + 56*A^{4} - 12*A^{2} +1))/D$

$RI(10,8) := (A^{3}*(- 40*A^{8} + 93*A^{6} - 77*A^{4} + 27*A^{2} - 3))/ D$

$RI(10,9) := (A*(40*A^{10} - 143*A^{8} + 172*A^{6} - 84*A^{4} + 16*A^{2} - 1))/ D$

$RI(10,10) := (120*A^{12} - 494*A^{10} + 719*A^{8} - 470*A^{6} + 144*A^{4} -20*A^{2} + 1)/ D$

$RI(10,11) := (A*(100*A^{10} - 240*A^{8} + 209*A^{6} - 83*A^{4} + 15*A^{2} -1))/D$

$RI(10,12) := (A^{2}*(40*A^{10} - 68*A^{8} + 17*A^{6} + 19*A^{4} - 9*A^{2} +1))/ D$

$RI(10,13) := (2*A^{2}*(-40*A^{10} + 143*A^{8} - 172*A^{6} + 84*A^{4} - 16*A^{2} +1))/D$

$RI(10,14) := (A*(40*A^{10} - 143*A^{8} + 172*A^{6} - 84*A^{4} + 16*A^{2} - 1))/D$

$RI(10,15) := (2*A^{2}*(-20*A^{10} + 84*A^{8} - 116*A^{6} + 65*A^{4} - 14*A^{2} +1))/D$

$RI(10,16) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$

$RI(11,1) := (2*A^{4}*(40*A^{8} - 93*A^{6} + 77*A^{4} - 27*A^{2} + 3))/ D$

$RI(11,2) := (A^{3}*(- 40*A^{8} + 93*A^{6} - 77*A^{4} + 27*A^{2} - 3))/ D$

$$RI(11,3) := (A^{2}*(40*A^{10} - 68*A^{8} + 17*A^{6} + 19*A^{4} - 9*A^{2} +1))/ D$$

$$RI(11,4) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$$

$$RI(11,5) := (A^{3}*(- 40*A^{8} + 93*A^{6} - 77*A^{4} + 27*A^{2} - 3))/ D$$

$$RI(11,6) := (2*A^{2}*(-60*A^{10} + 147*A^{8} - 132*A^{6} + 56*A^{4} - 12*A^{2} +1))/D$$

$$RI(11,7) := (A*(100*A^{10} - 240*A^{8} + 209*A^{6} - 83*A^{4} + 15*A^{2} - 1))/D$$

$$RI(11,8) := (2*A^{2}*(-20*A^{10} + 84*A^{8} - 116*A^{6} + 65*A^{4} - 14*A^{2} +1))/D$$

$$RI(11,9) := (A^{2}*(40*A^{10} - 68*A^{8} + 17*A^{6} + 19*A^{4} - 9*A^{2} +1))/ D$$

$$RI(11,10) := (A*(100*A^{10} - 240*A^{8} + 209*A^{6} - 83*A^{4} + 15*A^{2} -1))/D$$

$$RI(11,11) := (120*A^{12} - 494*A^{10} + 719*A^{8} - 470*A^{6} + 144*A^{4} -20*A^{2} + 1)/ D$$

$$RI(11,12) := (A*(40*A^{10} - 143*A^{8} + 172*A^{6} - 84*A^{4} + 16*A^{2} - 1))/D$$

$$RI(11,13) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$$

$$RI(11,14) := (2*A^{2}*(-20*A^{10} + 84*A^{8} - 116*A^{6} + 65*A^{4} - 14*A^{2} +1))/D$$

$$RI(11,15) := (A*(40*A^{10} - 143*A^{8} + 172*A^{6} - 84*A^{4} + 16*A^{2} -1))/D$$

$$RI(11,16) := (2*A^{2}*(-40*A^{10} + 143*A^{8} - 172*A^{6} + 84*A^{4} - 16*A^{2} +1))/D$$

$$RI(12,1) := (2*A^{5}*(30*A^{6} - 61*A^{4} + 36*A^{2} - 5))/ D$$

$$RI(12,2) := (2*A^{4}*(40*A^{8} - 93*A^{6} + 77*A^{4} - 27*A^{2} + 3))/ D$$

$$RI(12,3) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$$

$$RI(12,4) := (A^{2}*(-40*A^{10} + 118*A^{8} - 137*A^{6} + 73*A^{4} - 15*A^{2} +1))/D$$

$$RI(12,5) := (2*A^{4}*(- 40*A^{8} + 63*A^{6} - 16*A^{4} - 9*A^{2} + 2))/ D$$

$$RI(12,6) := (A^{3}*(- 40*A^{8} + 93*A^{6} - 77*A^{4} + 27*A^{2} - 3))/ D$$

$$RI(12,7) := (2*A^{2}*(-20*A^{10} + 84*A^{8} - 116*A^{6} + 65*A^{4} - 14*A^{2} +1))/D$$

$$RI(12,8) := (A*(140*A^{10} - 333*A^{8} + 286*A^{6} - 110*A^{4} + 18*A^{2} -1))/D$$

$$RI(12,9) := (A^{3}*(60*A^{8} - 97*A^{6} + 37*A^{4} + A^{2} - 1))/ D$$

$$RI(12,10) := (A^{2}*(40*A^{10} - 68*A^{8} + 17*A^{6} + 19*A^{4} - 9*A^{2} +1))/ D$$

$$RI(12,11) := (A*(40*A^{10} - 143*A^{8} + 172*A^{6} - 84*A^{4} + 16*A^{2} -1))/D$$

$$RI(12,12) := (80*A^{12} - 426*A^{10} + 702*A^{8} - 489*A^{6} + 153*A^{4}-21*A^{2} +1)/D$$

$$RI(12,13) := (2*A^{4}*(20*A^{8} - 59*A^{6} + 56*A^{4} - 19*A^{2} + 2))/ D$$

$$RI(12,14) := (A^{3}*(-100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/D$$

$$RI(12,15) := (2*A^{2}*(-40*A^{10} + 143*A^{8} - 172*A^{6} + 84*A^{4} - 16*A^{2} + 1))/D$$

$$RI(12,16) := (A*(140*A^{10} - 358*A^{8} + 321*A^{6} - 121*A^{4} + 19*A^{2} - 1))/D$$

$$RI(13,1) := (A^{3}*(25*A^{6} - 35*A^{4} + 11*A^{2} - 1))/D$$

$$RI(13,2) := (2*A^{4}*(20*A^{8} - 59*A^{6} + 56*A^{4} - 19*A^{2} + 2))/D$$

$$RI(13,3) := (2*A^{5}*(30*A^{6} - 61*A^{4} + 36*A^{2} - 5))/D$$

$$RI(13,4) := (4*A^{6}*(-30*A^{6} + 61*A^{4} - 36*A^{2} + 5))/D$$

$$RI(13,5) := (A^{2}*(-40*A^{10} + 118*A^{8} - 137*A^{6} + 73*A^{4} - 15*A^{2} + 1))/D$$

$$RI(13,6) := (A^{3}*(-100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/D$$

$$RI(13,7) := (2*A^{4}*(40*A^{8} - 93*A^{6} + 77*A^{4} - 27*A^{2} + 3))/D$$

$$RI(13,8) := (2*A^{5}*(30*A^{6} - 61*A^{4} + 36*A^{2} - 5))/D$$

$$RI(13,9) := (A*(140*A^{10} - 358*A^{8} + 321*A^{6} - 121*A^{4} + 19*A^{2} - 1))/D$$

$$RI(13,10) := (2*A^{2}*(-40*A^{10} + 143*A^{8} - 172*A^{6} + 84*A^{4} - 16*A^{2} + 1))/D$$

$$RI(13,11) := (A^{3}*(-100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/D$$

$$RI(13,12) := (2*A^{4}*(20*A^{8} - 59*A^{6} + 56*A^{4} - 19*A^{2} + 2))/D$$

$$RI(13,13) := (120*A^{12} - 544*A^{10} + 839*A^{8} - 562*A^{6} + 168*A^{4} - 22*A^{2} + 1)/D$$

$$RI(13,14) := (A*(140*A^{10} - 358*A^{8} + 321*A^{6} - 121*A^{4} + 19*A^{2} - 1))/D$$

$$RI(13,15) := (A^{2}*(-40*A^{10} + 118*A^{8} - 137*A^{6} + 73*A^{4} - 15*A^{2} + 1))/D$$

$$RI(13,16) := (A^{3}*(25*A^{6} - 35*A^{4} + 11*A^{2} - 1))/D$$

$$RI(14,1) := (2*A^{4}*(20*A^{8} - 59*A^{6} + 56*A^{4} - 19*A^{2} + 2))/D$$

$$RI(14,2) := (A^{3}*(60*A^{8} - 97*A^{6} + 37*A^{4} + A^{2} - 1))/D$$

$$RI(14,3) := (2*A^{4}*(-40*A^{8} + 63*A^{6} - 16*A^{4} - 9*A^{2} + 2))/D$$

$$RI(14,4) := (2*A^{5}*(30*A^{6} - 61*A^{4} + 36*A^{2} - 5))/D$$

$$RI(14,5) := (A^{3}*(-100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/D$$

$$RI(14,6) := (A^{2}*(40*A^{10} - 68*A^{8} + 17*A^{6} + 19*A^{4} - 9*A^{2} + 1))/D$$

$$RI(14,7) := (A^{3}*(-40*A^{8} + 93*A^{6} - 77*A^{4} + 27*A^{2} - 3))/D$$

$$RI(14,8) := (2*A^{4}*(40*A^{8} - 93*A^{6} + 77*A^{4} - 27*A^{2} + 3))/D$$

$$RI(14,9) := (2*A^{2}*(-40*A^{10} + 143*A^{8} - 172*A^{6} + 84*A^{4} - 16*A^{2} +1))/D$$

$$RI(14,10) := (A*(40*A^{10} - 143*A^{8} + 172*A^{6} - 84*A^{4} + 16*A^{2} -1))/D$$

$$RI(14,11) := (2*A^{2}*(-20*A^{10} + 84*A^{8} - 116*A^{6} + 65*A^{4} - 14*A^{2} +1))/D$$

$$RI(14,12) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$$

$$RI(14,13) := (A*(140*A^{10} - 358*A^{8} + 321*A^{6} - 121*A^{4} + 19*A^{2} -1))/D$$

$$RI(14,14) := (80*A^{12} - 426*A^{10} + 702*A^{8} - 489*A^{6} + 153*A^{4} -21*A^{2}+1)/D$$

$$RI(14,15) := (A*(140*A^{10} - 333*A^{8} + 286*A^{6} - 110*A^{4} + 18*A^{2} -1))/D$$

$$RI(14,16) := (A^{2}*(-40*A^{10} + 118*A^{8} - 137*A^{6} + 73*A^{4} - 15*A^{2} +1))/D$$

$$RI(15,1) := (2*A^{5}*(30*A^{6} - 61*A^{4} + 36*A^{2} - 5))/ D$$

$$RI(15,2) := (2*A^{4}*(- 40*A^{8} + 63*A^{6} - 16*A^{4} - 9*A^{2} + 2))/ D$$

$$RI(15,3) := (A^{3}*(60*A^{8} - 97*A^{6} + 37*A^{4} + A^{2} - 1))/ D$$

$$RI(15,4) := (2*A^{4}*(20*A^{8} - 59*A^{6} + 56*A^{4} - 19*A^{2} + 2))/ D$$

$$RI(15,5) := (2*A^{4}*(40*A^{8} - 93*A^{6} + 77*A^{4} - 27*A^{2} + 3))/ D$$

$$RI(15,6) := (A^{3}*(- 40*A^{8} + 93*A^{6} - 77*A^{4} + 27*A^{2} - 3))/ D$$

$$RI(15,7) := (A^{2}*(40*A^{10} - 68*A^{8} + 17*A^{6} + 19*A^{4} - 9*A^{2} +1))/ D$$

$$RI(15,8) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$$

$$RI(15,9) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$$

$$RI(15,10) := (2*A^{2}*(-20*A^{10} + 84*A^{8} - 116*A^{6} + 65*A^{4} - 14*A^{2} +1))/D$$

$$RI(15,11) := (A*(40*A^{10} - 143*A^{8} + 172*A^{6} - 84*A^{4} + 16*A^{2} -1))/ D$$

$$RI(15,12) := (2*A^{2}*(-40*A^{10} + 143*A^{8} - 172*A^{6} + 84*A^{4} - 16*A^{2} +1))/D$$

$$RI(15,13) := (A^{2}*(-40*A^{10} + 118*A^{8} - 137*A^{6} + 73*A^{4} - 15*A^{2} +1))/D$$

$$RI(15,14) := (A*(140*A^{10} - 333*A^{8} + 286*A^{6} - 110*A^{4} + 18*A^{2} -1))/D$$

$$RI(15,15) := (80*A^{12} - 426*A^{10} + 702*A^{8} - 489*A^{6} + 153*A^{4} - 21*A^{2} +1)/D$$

$$RI(15,16) := (A*(140*A^{10} - 358*A^{8} + 321*A^{6} - 121*A^{4} + 19*A^{2} -1))/D$$

$$RI(16,1) := (4*A^{6}*(- 30*A^{6} + 61*A^{4} - 36*A^{2} + 5))/ D$$

$$RI(16,2) := (2*A^{5}*(30*A^{6} - 61*A^{4} + 36*A^{2} - 5))/ D$$

$$RI(16,3) := (2*A^{4}*(20*A^{8} - 59*A^{6} + 56*A^{4} - 19*A^{2} + 2))/ D$$

$$RI(16,4) := (A^{3}*(25*A^{6} - 35*A^{4} + 11*A^{2} - 1))/ D$$

$$RI(16,5) := (2*A^{5}*(30*A^{6} - 61*A^{4} + 36*A^{2} - 5))/ D$$

$$RI(16,6) := (2*A^{4}*(40*A^{8} - 93*A^{6} + 77*A^{4} - 27*A^{2} + 3))/ D$$

$$RI(16,7) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$$

$$RI(16,8) := (A^{2}*(-40*A^{10} + 118*A^{8} - 137*A^{6} + 73*A^{4} - 15*A^{2} +1))/D$$

$$RI(16,9) := (2*A^{4}*(20*A^{8} - 59*A^{6} + 56*A^{4} - 19*A^{2} + 2))/ D$$

$$RI(16,10) := (A^{3}*(- 100*A^{8} + 215*A^{6} - 149*A^{4} + 37*A^{2} - 3))/ D$$

$$RI(16,11) := (2*A^{2}*(-40*A^{10} + 143*A^{8} - 172*A^{6} + 84*A^{4} - 16*A^{2} +1))/D$$

$$RI(16,12) := (A*(140*A^{10} - 358*A^{8} + 321*A^{6} - 121*A^{4} + 19*A^{2} -1))/D$$

$$RI(16,13) := (A^{3}*(25*A^{6} - 35*A^{4} + 11*A^{2} - 1))/ D$$

$$RI(16,14) := (A^{2}*(-40*A^{10} + 118*A^{8} - 137*A^{6} + 73*A^{4} - 15*A^{2} +1))/D$$

$$RI(16,15) := (A*(140*A^{10} - 358*A^{8} + 321*A^{6} - 121*A^{4} + 19*A^{2} -1))/D$$

$$RI(16,16) := (120*A^{12} - 544*A^{10} + 839*A^{8} - 562*A^{6} + 168*A^{4} -22*A^{2} + 1)/ D$$

APPENDIX C

A FORTRAN PROGRAM FOR THE SOLUTION OF LAPLACE'S EQUATION BY THE 16-POINT GROUP S.O.R. ITERATIVE METHOD

```
C                 ............................
C          .... THE NOTATIONS ARE AS FOLLOWS  ....
C                 ............................
C
C       N       NUMBER OF MESH POINTS IN THE X & Y DIRECTIONS
C       W       OVER-RELAXATION FACTOR
C       T       THE (K) ITERATIVE VALUES
C       TN      THE (K+1) ITERATIVE VALUES
C       ITER    ITERATION COUNTER
C       ITMAX   UPPER LIMIT OF NUMBER OF ITERATIONS
C
C       ......................................................
C       . THE RED-BLACK ORDERING IS USED TO ORDER THE GROUPS .
C       .                  OF 16 - POINTS                    .
C       ......................................................
C
C
        DIMENSION T(62,62),TN(62,62)
C
C       READ INPUT PARAMETERS
C
        READ(1,*)N,EPS,MK,Z1,Z2,ITMAX
        B=1./6600.
        DO 70 KK=1,MK
        W=Z1+KK*Z2
C
C       PRINT A HEADING
C
        WRITE(1,100) N,W
C
C       ESTABLISH BOUNDARY VALUES AND INITIALISE THE
C       INTERIOR POINTS

        NN=N+1
        DO 15 I=1,NN
          TN(I,1)=100.0
        DO 10 J=2,NN
          TN(I,J)=0.0
 10     CONTINUE
 15     CONTINUE
C
        K1=6
        K2=2
C
C       BEGIN THE ITERATIONS. LIMIT TO (ITMAX) ITERATIONS,
C       OR UNTIL THE CONVERGENCE CONDITION IS SATISFIED
C
        ITER=0
 20     ITER=ITER+1
        IF(ITER.GE.ITMAX) GO TO 60
        DO 25 J=2,N
        DO 25 I=2,N
          T(I,J)=TN(I,J)
 25     CONTINUE
```

```
 3Ø     CONTINUE
        DO 45 I=2,N,4
        L1=K2
        K3=K1
        K1=K2
        K2=K3
        DO 4Ø J=L1,N,8
C
        R1=T(I-1,J)+T(I,J-1)
        R2=T(I-1,J+3)+T(I,J+4)
        R3=T(I+3,J+4)+T(I+4,J+3)
        R4=T(I+4,J)+T(I+3,J-1)
        R5=T(I-1,J+1)
        R6=T(I-1,J+2)
        R7=T(I+1,J+4)
        R8=T(I+2,J+4)
        R9=T(I+4,J+2)
        R1Ø=T(I+4,J+1)
        R11=T(I+2,J-1)
        R12=T(I+1,J-1)
        S1=R5+R12
        S2=R6+R11
        S3=R7+R1Ø
        S4=R8+R9
        S5=R1Ø+R11
        S6=R9+R12
        S7=R5+R8
        S8=R6+R7
        S9=R1+R3
        S1Ø=R2+R4
C
C       .... THE G.S. SOLUTION OF 8 EQUATIONS BY
C               THE NEW 17-POINT MOLECULE          ....
C
         TN(I,J)=(1987.*R1+674.*S1+251.*S2+1Ø1.*S3+88.*S1Ø+
     *            74.*S4+37.*R3)*B
C
         TN(I+2,J)=(2238.*R11+762.*R12+674.*R4+458.*R1Ø+251.*R1+
     *            242.*(R5+R9)+162.*R8+158.*R6+138.*R7+1Ø1.*R3+
     *            74.*R2)*B
C
         TN(I+1,J+1)=(916.*S1+559.*S2+458.*R1+4Ø9.*S3+316.*S4+
     *              242.*S1Ø+158.*R3)*B
C
         TN(I+3,J+1)=(2238.*R1Ø+762.*R9+674.*R4+458.*R11+251.*
     *              R3+242.*(R8+R12)+162.*R5+158.*R7+138.*R6+
     *              1Ø1.*R1+74.*R2)*B
C
         TN(I,J+2)=(2238.*R6+762.*R5+674.*R2+458.*R7+251.*R1+
     *             242.*(R8+R12)+162.*R9+158.*R11+138.*R1Ø+
     *             1Ø1.*R3+74.*R2)*B
C
         TN(I+2,J+2)=(916.*S4+559.*S3+458.*R3+4Ø9.*S2+316.*S1+
     *               242.*S1Ø+158.*R1)*B
C
         TN(I+1,J+3)=(2238.*R7+762.*R8+674.*R2+458.*R6+251.*
     *               R3+242.*(R5+R9)+162.*R12+158.*R1Ø+138.*
     *               R11+1Ø1.*R1+74.*R4)*B
```

```
      TN(I+3,J+3)=(1987.*R3+674.*S4+251.*S3+101.*S2+88.*S10+
     *                   74.*S1+37.*R1)*B
C
C
      TK1=TN(I,J)+TN(I+1,J+1)
      TK2=TN(I+2,J)+TN(I+3,J+1)
      TK3=TN(I+1,J+1)+TN(I+2,J+2)
      TK4=TN(I,J+2)+TN(I+1,J+3)
      TK5=TN(I+2,J+2)+TN(I+3,J+3)
C
C     ....THE G.S. SOLUTION OF THE REMAINING EQUATIONS
C                      BY THE 5-POINT FORMULA      ....
C
      TN(I+1,J)=(TK1+TN(I+2,J)+R12)*0.25
      TN(I+3,J)=(TK2+R4)*0.25
      TN(I,J+1)=(TK1+TN(I,J+2)+R5)*0.25
      TN(I+2,J+1)=(TK2+TK3)*0.25
      TN(I+1,J+2)=(TK3+TK4)*0.25
      TN(I+3,J+2)=(TN(I+3,J+1)+TK5+R9)*0.25
      TN(I,J+3)=(TK4+R2)*0.25
      TN(I+2,J+3)=(TK5+TN(I+1,J+3)+R8)*0.25
C
C
C     APPLY THE OVER-RELAXATION FORMULA
C
      DO 35 JK=J,J+3
      DO 35 IK=I,I+3
 35   TN(IK,JK)=W*(TN(IK,JK)-T(IK,JK))+T(IK,JK)
 40   CONTINUE
 45   CONTINUE
      K3=K1
      K1=K2
      K2=K3
      IF(K2.NE.2) GO TO 30
C
C     ** CONVERGENCE TEST **
C
      DO 50 J=2,N
      DO 50 I=2,N
      IF(ABS(T(I,J)-TN(I,J))/(1.+ABS(TN(I,J))).GT.EPS) GO TO 3
 50   CONTINUE
C
C     PRINT VALUES OF THE ITERATION COUNTER, ITER,
C     AND THE FINAL VALUES OF THE MESH POINTS
C
C
      WRITE(1,110) ITER
C
      DO 55 I=1,NN
      WRITE(1,120)(TN(I,J),J=1,NN)
 55   CONTINUE
C
```

```
C       .... COMMENT IN CASE ITER EXCEEDS ITMAX ....
C
 6Ø     WRITE(1,13Ø) ITMAX
        DO 65 I=1,NN
        WRITE(1,12Ø)(TN(I,J),J=1,NN)
 65     CONTINUE
 7Ø     CONTINUE
 1ØØ    FORMAT(/1X,'N      =',I5/1X,'W      =',F8.5)
 11Ø    FORMAT(/1X,'CONVERGENCE CONDITION HAS BEEN REACHED
     *     AFTER',I3,1X,'ITERATIONS'/1X,'THE VALUES OF T ARE')
 12Ø    FORMAT(//1X,14(F7.3,1X))
 13Ø    FORMAT(/1X,'CONVERGENCE NOT REACHED IN',I4,1X,
     *           'ITERATIONS. CURRENT VALUES OF T ARE')
        CALL EXIT
        END
```

APPENDIX D

A FORTRAN PROGRAM FOR THE SOLUTION OF LAPLACE'S EQUATION BY THE 4-POINT, 2-LINE (IBEB) ITERATIVE METHOD

```
C                       .............................
C                       .THE NOTATIONS ARE AS FOLLOWS.
C                       .............................
C
C     N         NUMBER OF MESH POINTS IN THE X & Y DIRECTIONS
C     W1        THE FIRST RELAXATION FACTOR
C     W2        THE SECOND RELAXATION FACTOR
C     X         THE (K+1) ITERATIVE VALUES
C     HX        THE (K) ITERATIVE VALUES
C     ITER      ITERATION COUNTER
C     ITMAX     UPPER LIMIT OF NUMBER OF ITERATIONS
C     A         LOWER SUBMATRICES OF THE BLOCK TRIDIAGONAL MATRIX RØ
C               OF (5.3.11a)
C     B         DIAGONAL SUBMATRICES OF RØ
C     C         UPPER SUBMATRICES OF RØ

      DIMENSION X(2ØØ),HX(2ØØ),R(26),Z(4,4),B(4,4),C(4,4),A(4,4),
     1 D(4,8),ZZ(4,4),U(6,6),H(4,8),GG(4,8),H1(4),G(4,4,8),XX(4,8)
C
C      INPUT PARAMETERS
C
      N=14
      W1=1.33
      Z1=Ø.675
      Z2=Ø.ØØ1
      EPS=Ø.ØØØØØØ1
      ITMAX=2Ø
      MK=25
C
C      COMPUTE SOME CONSTANTS
C
      N1=N+1
      N2=N+N
      N3=N*N
      N4=N-2
      N5=N+2
      N6=N3-N
      NM=N4/2
C
C      ESTABLISH THE MATRICES A,B & C
C
      DO 1Ø I=1,4
      DO 1Ø J=1,4
       B(I,J)=Ø.
       A(I,J)=Ø.
       C(I,J)=Ø.
       B(I,I)=24.
 1Ø   CONTINUE
C
      E=1./24.
      DO 75 LL=1,MK
      W2=Z1+LL*Z2
      P=-W1
      Q=2.*P
      S=7.*P
C
```

```
      A(1,3)=S
      A(1,4)=Q
      A(2,3)=Q
      A(2,4)=S
      A(3,3)=Q
      A(3,4)=P
      A(4,3)=P
      A(4,4)=Q
C
      C(1,1)=Q
      C(1,2)=P
      C(2,1)=P
      C(2,2)=Q
      C(3,1)=S
      C(3,2)=Q
      C(4,1)=Q
      C(4,2)=S
C
C     PRINT A HEADING
C
      WRITE(2,15) N,W1,W2
 15   FORMAT(//1X,'N    =',I5/1X,'W1   =',F8.5/1X,'W2   =',F8.5)
C
C     ESTABLISH BOUNDARY VALUES & INITIALIZE THE INTERIOR POINTS
C
      DO 20 I=N1,N3
       X(I)=0.
20    CONTINUE
      DO 25 I=1,N
       X(I)=100.
 25   CONTINUE
C
      ITER=0
 30   CONTINUE
      DO 35 I=N1,N3
       HX(I)=X(I)
 35   CONTINUE
C
C     BEGIN THE ITERATIONS LIMIT TO (ITMAX).OR UNTIL THE
C     THE CONVERGENCE CONDITION IS SATISFIED.
C
      ITER=ITER+1
      IF(ITER.GT.ITMAX)GO TO 65
C
      DO 50 L=2,N4,2
      NN1=N*(L-1)+2
      NN2=N*L-2
      NN3=NN2+1
C
      NK=1
C
      DO 40 I=NN1,NN2,2
       R(NK)=W1*(7.*X(I-N)+2.*(X(I+1-N)+HX(I+N2))+HX(I+1+N2))
     1         +24.*(1.-W1)*HX(I)
C
```

```
        R(NK+1)=W1*(X(I+1-N)+2.*(X(I-N)+HX(I+1+N2))+7.*HX(I+N2))
     2            +24.*(1.-W1)*HX(I+N)
C
        R(NK+2)=W1*(7.*X(I+1-N)+2.*(X(I-N)+HX(I+1+N2))+HX(I+N2))
     3            +24.*(1.-W1)*HX(I+1)
C
        R(NK+3)=W1*(X(I-N)+2.*(X(I+1-N)+HX(I+N2))+7.*HX(I+1+N2))
     4            +24.*(1.-W1)*HX(I+1+N)
C
      NK=NK+4
 40   CONTINUE
C
C     CALLING THE SUBROUTINE TO SOLVE THE SYSTEM OF EQUATIONS
C
      CALL SOLVE(B,A,C,R,H,U,G,GG,L,Z,ZZ,N,XX,W1,D)
C
C     APPLY THE S.O.R. FORMULA AND PUT THE NEW VALUES IN ARRAY X.
C
      NK=1
      DO 45 I=NN1,NN3
        X(I)=W2*(R(NK)-HX(I))+HX(I)
        X(I+N)=W2*(R(NK+1)-HX(I+N))+HX(I+N)
      NK=NK+2
 45   CONTINUE
 50   CONTINUE
C
C     ***TEST FOR CONVERGENCY***
C
      DO 55 I=N5,N6
      IF(ABS(X(I)-HX(I))/(1.+ABS(HX(I))).GT.EPS) GO TO 9
 55   CONTINUE
C
      WRITE(2,60) ITER
 60   FORMAT(/1X,'NUMBER OF ITERATION =',I3)
      GO TO 75
C
C     ***IF ITER>ITMAX***
C
 65    WRITE(2,70)ITMAX
 70   FORMAT(/1X,'CONVERGENCE NOT REACHED IN',I4,1X,'ITERATIONS')
 75    CONTINUE
      STOP
      END
C
C
C
C     THIS SUBROUTINE SOLVE RX= D FOR X (WHERE R IS A BLOCK
C     TRIDIAGONAL MATRIX)
C
      SUBROUTINE SOLVE(B,A,C,R,H,U,G,GG,L,Z,ZZ,N,XX,W1,D)
      DIMENSION B(4,4),A(4,4),C(4,4),Z(4,4),ZZ(4,4),H1(4),F(6,6),
     1 R(26),D(4,8),H(4,8),GG(4,8),U(6,6),XX(4,8),G(4,4,8),
     2 WKSPCE(6)
C
      NM=(N-2)/2
      E=1./24.
```

```
      LK=1
      DO 8Ø J=1,NM
      DO 8Ø I=1,4
       D(I,J)=R((J-1)*4+I)
 8Ø   CONTINUE
C
                  -1          -1
C     FIND H1=B1  *D1 & G1=B1  *C1
C
      DO 85 I=1,4
       H(I,1)=D(I,1)*E
      DO 85 J=1,4
       Z(I,J)=C(I,J)*E
       G(I,J,1)=Z(I,J)
 85   CONTINUE
C
C
      DO 14Ø K=2,NM
C
C     FIND B(I)-A(I)*G(I-1) AND PUT THE RESULTS IN F
C
      DO 95 I=1,4
      DO 95 J=1,4
        ZZ(I,J)=Ø.
      DO 9Ø KJ=1,4
        ZZ(I,J)=ZZ(I,J)+A(I,KJ)*Z(KJ,J)
 9Ø   CONTINUE
        F(I,J)=B(I,J)-ZZ(I,J)
 95   CONTINUE
C
C     CALLING THE LIBRARY SUBROUTINE FØ1AAF TO INVERT  MATRIX F
C     AND PUT THE RESULTS IN U
C
      IFAIL=Ø
      CALL FØ1AAF(F,6,4,U,6,WKSPCE,IFAIL)
C
      IF(K.EQ.NM) GO TO 115
C     FIND G(I)=U(I)*C(I)  *SEE (5.4.6)*
C
      DO 1Ø5 I=1,4
      DO 1Ø5 J=1,4
        Z(I,J)=Ø.
      DO 1ØØ KJ=1,4
        Z(I,J)=Z(I,J)+U(I,KJ)*C(KJ,J)
 1ØØ  CONTINUE
 1Ø5  CONTINUE
C
      DO 11Ø I=1,4
      DO 11Ø J=1,4
        G(I,J,K)=Z(I,J)
 11Ø  CONTINUE
C
 115  CONTINUE
C
```

```
C       FIND  D(I)-A(I)*H(I-1)
C
        DO 125 I=1,4
          H1(I)=Ø.Ø
        DO 12Ø J=1,4
          H1(I)=H1(I)+A(I,J)*H(J,K-1)
  12Ø   CONTINUE
          D(I,K)=D(I,K)-H1(I)
  125   CONTINUE
C
C       FIND  H(I)=U(I)*(D(I)-A(I)*H(I-1))   *SEE (5.4.7)*
C
        DO 135 I=1,4
         H(I,K)=Ø.
        DO 13Ø K1=1,4
          H(I,K)=H(I,K)+U(I,K1)*D(K1,K)
  13Ø    CONTINUE
  135   CONTINUE
  14Ø   CONTINUE
C
C        **** NOW OUTPUT RESULT ****
C
        DO 145 I=1,4
          XX(I,NM)=H(I,NM)
  145   CONTINUE
        MM=NM-1
        DO 16Ø II=1,MM
         I=NM-II
        DO 155 KI=1,4
         GG(KI,I)=Ø.
        DO 15Ø KJ=1,4
         GG(KI,I)=GG(KI,I)+G(KI,KJ,I)*XX(KJ,I+1)
  15Ø   CONTINUE
        XX(KI,I)=H(KI,I)-GG(KI,I)
  155   CONTINUE
  16Ø   CONTINUE
C
C
         L1=1
        DO 165 J=1,NM
        DO 165 I=1,4
          R((J-1)*4+I)=XX(I,J)
  165    CONTINUE
C
        RETURN
        END
```

APPENDIX E

A FORTRAN PROGRAM FOR THE SOLUTION OF A NON-LINEAR PROBLEM BY THE A.G.E. METHOD

```
C     .... A FORTRAN PROGRAM FOR THE SOLUTION OF THE ....
C                FOLLOWING NON-LINEAR PROBLEB
C                ...........................


C     U''=U1**3-SIN(X)*(1+SIN(X)**2)
C     U(Ø)=U(PIE)=Ø
C
C     THE EXACT SOLUTION IS GIVEN BY:
C        U(X)=SIN(X)
C
C
C       H       THE GRID SPACING
C       R       ITERATION PARAMETER
C       ITER    ITERATIONS COUNTER
C       UN      THE (K) ITERATIVE VALUES
C       UM      THE INTERMEDIATE VALUES
C       U       THE (K+1) ITERATIVE VALUES
C
C
        DIMENSION U(51),UN(51),X(51),G(51),S(51),C(51),D(51)
     1      ,UM(51)
C
C       READ INPUT PARAMETERS & COMPUTE SOME CONSTANTS
C
        READ(1,*)N,EPS,MK,Z1,Z2
C
        N1=N-1
        N2=N-2
        N3=N-3
        NH=N/2
        NH1=NH-1
        NN=N+1
        H=3.14159/NN
        HH=H*H
        H2=HH*Ø.5
        L=1
        DO 5 I=1,N
         X(I)=L*H
        S(I)=SIN(X(I))
        C(I)=HH*S(I)*(1.+S(I)*S(I))
         L=L+1
5       CONTINUE
        DO 55 KK=1,MK
         R=Z1+KK*Z2
        WRITE(1,1ØØ)N,R
C
C       SET INITIAL CONDITION
        DO 1Ø I=1,N
         U(I)=Ø.8
1Ø      CONTINUE
        U(NN)=Ø.Ø
C
C
C
```

```
C       BEGIN THE ITERATIONS
C
        ITER=0
15      CONTINUE
        ITER=ITER+1
        DO 20 I=1,NN
         UN(I)=U(I)
        G(I)=1.+H2*UN(I)*UN(I)
20      CONTINUE
        DO 25 I=1,NH
         I1=I+I
         I2=I1-1
        D(I)=1./((G(I1)+R)*(G(I2)+R)-1.)
25      CONTINUE
C
C
        R1=C(1)-(G(1)-R)*UN(1)
        R2=C(2)-(G(2)-R)*UN(2)+UN(3)
        UM(1)=((G(2)+R)*R1+R2)*D(1)
        UM(2)=(R1+(G(1)+R)*R2)*D(1)
C
        L=2
        DO 30 I=3,N1,2
         I1=I+1
         I2=I+2
         II=I-1
        R3=C(I)+UN(II)-(G(I)-R)*UN(I)
        R4=C(I1)-(G(I1)-R)*UN(I1)+UN(I2)
        UM(I)=((G(I1)+R)*R3+R4)*D(L)
        UM(I1)=(R3+(G(I)+R)*R4)*D(L)
C
         L=L+1
30      CONTINUE
C
        DO 35 I=1,NN
        G(I)=1.+H2*UM(I)*UM(I)
35      CONTINUE
        DO 40 I=1,NH1
          ID=I+I
          IP=ID+1
        D(I)=1./((G(ID)+R)*(G(IP)+R)-1.)
40      CONTINUE
C
        U(1)=(C(1)-(G(1)-R)*UM(1)+UM(2))/(G(1)+R)
        IJ=1
        DO 45 I=2,N2,2
          I1=I+1
          I2=I+2
          II=I-1
C
        P1=C(I)+UM(II)-(G(I)-R)*UM(I)
        P2=C(I1)-(G(I1)-R)*UM(I1)+UM(I2)
        U(I)=((G(I1)+R)*P1+P2)*D(IJ)
        U(I1)=(P1+(G(I)+R)*P2)*D(IJ)
        IJ=IJ+1
45      CONTINUE
        U(N)=(C(N)+UM(N1)-(G(N)-R)*UM(N))/(G(N)+R)
```

```
C
C       CONVERGENCE TEST
C
        DO 5Ø I=1,N
        IF(ABS(U(I)-UN(I))/(1.+ABS(UN(I))).GT.EPS)GO TO 15
5Ø      CONTINUE
        WRITE(1,11Ø)ITER
55      CONTINUE
1ØØ     FORMAT(/1X,'N    =',I5/1X,'R    =',F8.5)
11Ø     FORMAT(1X,'NUMBER OF ITERATIONS =',I4)
        CALL EXIT
        END
```