

This item was submitted to Loughborough University as a PhD thesis by the author and is made available in the Institutional Repository (<https://dspace.lboro.ac.uk/>) under the following Creative Commons Licence conditions.



For the full text of this licence, please go to:
<http://creativecommons.org/licenses/by-nc-nd/2.5/>

BLLID No:- D56866/85

**LOUGHBOROUGH
UNIVERSITY OF TECHNOLOGY
LIBRARY**

AUTHOR/FILING TITLE

MORGAN, A

ACCESSION/COPY NO.

008555/01

VOL. NO.

CLASS MARK

T

LOAN COPY

19 OCT 2000

000 8555 01



This book was bound by

Badminton Press

18 Half Croft, Syston, Leicester, LE7 8LD

Telephone: Leicester (0533) 602918.

Development of Numerical Methods for the
Solution of Integral Equations

by

Anthony Morgan

A Doctoral Thesis

*Submitted in partial fulfilment of the
requirements for the award of Doctor of Philosophy
of the Loughborough University of Technology.*

August 1984.

Supervisors: J. Hyslop, Ph.D. and G.A. Evans, D.Phil.
Department of Mathematics.

© by A.P.G. Morgan, 1984.

Leuchter	Security
Date	June 1985
Page	008008/01
008555/01	

It is with the greatest pleasure that the author would like to thank his long-standing supervisors Dr.J.Hyslop and Dr.G.A.Evans with whom many interesting and enjoyable tutorials were spent. To Professor C.Storey special thanks are due for making departmental facilities available and providing the impetus which finally resulted in the completion of this work. The dexterity of Miss N.F.Jones was astounding. Without her help and that of Mrs.Betty Wright, who was a staunch ally in the closing stages, many more candles would have been burnt at both ends. To each and everyone, thankyou.

CONTENTS

	Page
1. INTRODUCTION AND BACKGROUND	1
2. SUCCESSIVE APPROXIMATIONS	9
3. SOME BASIC THEORY	65
4. ALTERNATIVE WAYS OF IMPLEMENTING SUCCESSIVE APPROXIMATIONS	112
5. SOME NEW IMPLEMENTATIONS OF NEWTON'S METHOD	136
6. THE SOLUTION OF SOME SINGULAR INTEGRAL EQUATIONS	152
7. APPLICATION TO SYSTEMS OF EQUATIONS	162
8. CONCLUSIONS	172
APPENDICES	176
REFERENCES	203

ABSTRACT

Recent surveys have revealed that the majority of numerical methods for the solution of integral equations use one of two main techniques for generating a set of simultaneous equations for their solution. Either the unknown function is expanded as a combination of basis set functions and the resulting coefficients found, or the integral is discretized using quadrature formulae. The latter results in simultaneous equations for the solution at the quadrature abscissae.

The thesis proposes techniques based on various direct iterative methods, including refinements of residual correction which hold no restrictions for nonlinear integral equations. New implementations of successive approximations and Newton's method appear. The latter compares particularly well with other versions as the evaluation of the Jacobian can be made equivalent to the solution of matrix equations of relatively small dimensions. The method can be adapted to the solution of first-kind equations and has been applied to systems of integral equations. The schemes are designed to be adaptive with the aid of the progressive quadrature rules of Patterson or Clenshaw and Curtis and interpolation formulae. The Clenshaw-Curtis rule is particularly favoured as it delivers error estimates.

A very powerful routine for the solution of a wide range of

integral equations has resulted with the inclusion of a new efficient method for calculating singular integrals.

Some work is devoted to the conversion of differential to integral or integro-differential equations and comparing the merits of solving a problem in its original and converted forms.

Many equations are solved as test examples throughout the thesis of which several are of physical significance. They include integral equations for the slowing down of neutrons, the Lane-Emden equation, an equation arising from a chemical reactor problem, Chandrasekhar's isotropic scattering of radiation equation and the Blasius equation in boundary layer theory.

CHAPTER 1

INTRODUCTION AND BACKGROUND

1.1 INTRODUCTION

The research contained in this thesis was originally prompted by work on the one-electron equations of the Hartree-Fock self consistent field model for the energies of atomic systems [133]. These equations were converted to integral equations and recursive integration schemes applied to obtain variational solutions to the eigenvalue problem. The numerical integration scheme used was based on the well-known procedure of Clenshaw and Curtis [4] which has several useful advantages in the present context over many alternative methods. As well as delivering high accuracy results in many cases - Gaussian quadrature accuracy regularly being approached - the configuration of abscissae makes it possible to double the quadrature order without wasting previously computed integrand values. Since local error estimates may be cheaply obtained, continuous monitoring of the quadrature makes it possible to create very accurate automatic numerical schemes.

The success of the technique for eigenvalue problems suggested that it might be applicable to the direct solution of integral equations by the classical Neumann series. This method possesses a very important advantage over many other methods in that it may be applied directly to nonlinear problems. This is a highly desirable feature of an integral equation solver since so many physical problems give rise to nonlinear equations. In contrast, many existing methods rely on converting the integral equation to a system of equations

which can become very difficult to solve if nonlinearities arise.

Initial investigations (Evans, Hyslop, Morgan [90]) confirmed that the Neumann series or successive approximations approach when implemented with the Clenshaw-Curtis based quadrature scheme was capable of delivering very accurate results with relatively few function evaluations and arithmetic manipulations. In that original paper the points at which solution values were found were made to correspond to quadrature abscissae as in the original work of Nyström [7]. Since then schemes have been developed which treat the solution collocation points and quadrature abscissae as being independent of one another. This allows a much more flexible approach and aids in the construction of automatic, adaptive and progressive numerical schemes. Such schemes are currently being developed, but the results in this thesis will be derived using hand set parameters. Within the theoretical limits of a particular method, computational cheapness is often sacrificed to allow the solution of as wide a range of problems as possible. These would include its ability to solve nonlinear problems and those possessing singularities.

Convergence problems associated with the use of the Neumann series, particularly in connection with Fredholm equations do occur and have been overcome in several ways. The creation of equivalence classes of integral equations has achieved limited success. A formulation of the shooting method for

integral equations has also proved useful for those equations which arise from boundary value problems. Scope for further development of this technique, not necessarily in conjunction with successive approximations does exist. The application of an accelerator [76] has also proved to be beneficial in many cases. For those situations in which the successive approximation methods break down, new direct iterative schemes have been devised. New implementations of Newton's method have proved to be very successful in this respect. The underlying numerical schemes are the same as before, but the basic method is changed from Neumann series to Newton's method. At the cost of some extra computational expense it has been possible to produce a method (Evans, Hyslop, Morgan [65]) which is again capable of solving both nonlinear and linear integral equations very accurately.

Since many physical situations yield singular integral equations, the quadrature schemes have been extended (Evans, Hyslop, Morgan [12]) to cater for this occurrence. Again generality is given priority over the need to keep a scheme computationally inexpensive but the use of powerful integrators such as the Clenshaw-Curtis [4] or Patterson [5] rules does help in keeping work to a minimum. The result has been the creation of a very powerful singular integral evaluator which is useful in solving many singular integral equations.

It is well known that many problems are expressible in both differential and integral equation form. Some research has

also been devoted to the conversion of differential to integral equations and some comparisons made of the solutions to both types of equation.

Finally, the numerical schemes which have been devised have been extended in order to treat systems of equations.

Computations were carried out in Algol68 using an ICL 1904S* and Honeywell computers. Generally actual programs are not included in the thesis although some comment is made in the conclusions as to a possible automatic integral equation package.

The method of successive approximations is considered to be an obvious starting point for nonlinear equation problems, and will therefore form the basis of the next chapter in which numerical schemes for the solution of such problems will be created. Schemes will initially be created using linear examples and the work will then be extended to the solution of nonlinear equations.

Before going on to the details of a numerical method in Chapter 2, integral equations will be classified in the next section.

1.2 CLASSIFICATION OF INTEGRAL EQUATIONS

Many of the integral equations which will be treated by the schemes described in chapters 2 to 7 are of the second kind. A general form for such an equation is the Urysohn integral equation

$$f(x) = g(x) + \int_a^b K(x, y; f(x), f(y)) dy, \quad x \in [a, b], \quad (1)$$

where f is the unknown function and g and K are given functions. The function g is known as the driving or free term, and K as the kernel of the integral equation. The range of definition is $[a, b]$. Equation (1) is a nonlinear integral equation since K may involve nonlinear functions of f . More commonly K will only depend on $f(y)$ although kernels depending on $f(x)$ do occur (see Chandrasekhar's integral equation from radiative transfer theory (A0.4-1)). If $g(x)$ is identically equal to zero then the integral equation is said to be homogeneous. If the kernel K can be expressed in the form

$$K(x, y; f(x), f(y)) = L(x, y) f(y) \quad (2)$$

the equation (1) is known as a linear integral equation. The integral in (1) is a definite integral and all equations containing fixed limits of integration are known as Fredholm integral equations. If one of the limits of integration is variable (it usually takes the value x) then the equation is a Volterra integral equation. Circumstances do also arise in

which both limits of integration are variable (as for example in equation (2.3.4-3) relating to the integral equations for the slowing down of neutrons) and these too are Volterra integral equations.

If one or both limits of integration are infinite or if the free term or kernel become infinite for any values of x, y , or f then the equation is a singular integral equation. Examples are:

$$f(x) = g(x) + \int_{-\infty}^{\infty} e^{-|x-y|} f(y) dy \quad (3)$$

[15] and

$$f(x) = 1 + \int_0^1 \ln|x-y| f(y) dy. \quad (4)$$

[97]. Equations possessing kernels of the form

$$K(x, y) = \frac{H(x, y)}{|x-y|^\alpha}, \quad 0 < \alpha < 1, \text{ } H \text{ bounded} \quad (5)$$

are called weakly singular. If $\alpha=1$ then the equation involves a Cauchy principal value integral which must be evaluated in the form

$$\int_a^b \frac{H(x, y)}{|x-y|} dy = \lim_{\epsilon \rightarrow 0} \left(\int_a^{x-\epsilon} \frac{H(x, y)}{|x-y|} dy + \int_{x+\epsilon}^b \frac{H(x, y)}{|x-y|} dy \right), \quad (6)$$

It is possible to reduce a weakly singular equation to one which is nonsingular by producing an iterated kernel $K_r(x, y)$ of the original kernel $K(x, y)$ (see [15]). Morse and Feshbach

[133] define equations which cannot be made nonsingular as being intrinsically singular.

Equations which take the form

$$g(x) = \int_a^{\psi(x,b)} K(x,y;f(y))dy, \quad x,y \in [a,b] \quad (7)$$

are called integral equations of the first kind. If $\psi(x,b)=x$ then the equation is of Volterra type, otherwise if $\psi(x,b)=b$ then it is a Fredholm equation.

CHAPTER 2

SUCCESSIVE APPROXIMATIONS

2.1 INTRODUCTION

This chapter will deal with numerical implementations of successive approximations. The schemes will be introduced to solve nonsingular second kind equations of Volterra and Fredholm type. Applications to equations possessing singularities appear in chapter 6. In general the method may only be applied to second kind equations. There are exceptions however. Linear first-kind Volterra equations for example, may be converted to second kind equations if the driving term and kernel are continuously differentiable and $K(x,x) \neq 0$. Additionally first kind Fredholm equations possessing positive definite symmetric kernels can also be treated successfully by this approach.

As mentioned in the introduction to the thesis, many methods for the numerical solution of second kind equations are descendants of the Nyström theory [7]. Consequently, they involve the solution of sets of equations and may therefore require prohibitive amounts of storage space if high accuracy quadratures are needed. Iterative methods seek to overcome this difficulty as the equivalent matrix elements do not require storage, but are calculated as required. Brakhage [8] was the first to use such a scheme and since then authors such as Atkinson [9] and Marsh and Wadsworth [10] have produced more sophisticated modifications. Systems of equations still occur and this fact causes problems in the solution of nonlinear equations. Linearization of the nonlinear integral operators has been used to alleviate this

complication, but the discussion of an improved implementation of such ideas will be delayed to Chapter 5. Picard's method of integrating successive approximations, which is of great importance for the existence theory of ordinary differential equations [26] can be readily employed in solving integral equations and it should be stressed that the method is as easy to apply to nonlinear as to linear equations. Surprisingly very little appears on numerical algorithms for its implementation in the literature. This may be partly due to the fact that the iteration does not always converge. Nevertheless the method has been used. See for example Lock [27] or Siekman [28]. The major criticism of their prescriptions is the lack of powerful quadrature rules in evaluating integrals. Siekman for example uses Simpson's rule. To illustrate the techniques, some practical examples are included and comparisons made with previous work. In particular the problem of neutron slowing down gives a good environment in which to test the method, [90].

Baker [11] p.422 expresses the feeling "that any sustained application of an analytical iterative technique can be cumbersome and ill suited to a numerical method". The following sections describe superior implementations of the method and reveal that in these forms, successive approximations provides an excellent means of solving many integral equations in a fast and efficient manner. Following a brief outline of the numerical details, the chapter is laid out in four main parts which deal with the application of successive approximations to both linear and nonlinear

Volterra equations and similar sections relating to Fredholm equations.

2.2 A NUMERICAL SCHEME

The following paragraphs clarify the way in which the method of successive approximations is implemented and also indicate the way in which other methods will be adopted. Investigations will begin with Volterra equations since the method of successive approximations copes most successfully with this type of equation. Linear equations will be dealt with first. In suggesting numerical schemes for the solution of the Volterra integral equation

$$f(x) = g(x) + \int_a^x K(x, y) f(y) dy \quad (1)$$

authors frequently mention the possibility of treating the equation as being of Fredholm type in which the kernel $K(x, y)$ equals zero for y greater than x . They advise against the use of this device since it introduces a jump discontinuity into the integrand unless $K(x, x)$ equals zero. See for example [1] or [11]. If the kernel $K(x, y)$ is defined for y greater than x , it is possible to derive a numerical scheme which overcomes these difficulties associated with Volterra equations. This is made possible by employing the discretization

$$f_N^{[r+1]}(x_i) = g(x_i) + \lambda \sum_{j=0}^N w_j(x_i) K(x_i, x_j) f_N^{[r]}(x_j), \quad i=0(1)N, \quad (2)$$

$$r=0(1)R,$$

to be used subsequently in solving Fredholm equations. The

indefinite integration formulae of Clenshaw and Curtis [4], (A2-2)-(A2-9) can be implemented in the standard quadrature manner of El-gendi [63] to obtain the correct values for the indefinite integrals. The notorious problems inherent in classical step-by-step methods - namely the calculation of accurate solution values near to the initial point - are thus conveniently eliminated. Naturally this device will not be appropriate for kernels such as $K(x,y)=\ln(z-y)$ since this kernel becomes undefined for $y>x$. Alternative measures will subsequently be taken to secure the solution of these equations. A linear transformation maps the interval of integration $[a,b]$ on to $[-1,1]$ and the grid-points x_i are allocated according to the formula

$$x_i = ((b-a)s_i + a + b)/2, \quad i=0(1)N, \quad (3)$$

where

$$s_i = -\cos(i\pi/N), \quad i=0(1)N. \quad (4)$$

The error estimates (A2-10,11) or other appropriate estimates ([4],[49],[50]) indicate how large a value of N should be used. N is chosen to ensure that the local truncation errors generated by the quadrature are less than $\epsilon=10^{-1}\epsilon^*$ where ϵ^* is the error tolerance imposed on the iteration cycle. This condition is included in an attempt to prevent errors which are of the same magnitude as ϵ^* being submitted to the next step of the iteration and hence once more to the quadrature rule only to be magnified in the course of the arithmetic manipulations. A similar bound is used in halting the

iteration cycle which terminates when r equals R for some R satisfying the inequality

$$|f_N^{[R]}(x_i) - f_N^{[R-1]}(x_i)| < \epsilon |f_N^{[R]}(x_i)|, \quad i=0(1)N. \quad (5)$$

Condition (5) may be implemented in two ways. Either the iteration continues at all points until (5) is satisfied for each of them, or it continues only for those points at which convergence has not yet occurred. The latter option proves to be computationally less expensive than the former, but the device of using $\epsilon=10^{-1}\epsilon^*$ as an error bound should be strengthened to $\epsilon=10^{-n}\epsilon^*$, $n>1$, for convergence to be satisfied. Experience has shown that $n=2$ is usually sufficient, but unless this condition is used the accuracy attained at points which have converged early in the iteration process impairs the attainable accuracy at points which have yet to converge. Inequality (5) involves relative errors and will be the most commonly used criterion although comparison of some methods will be carried out in terms of absolute errors for which the solution values must satisfy

$$|f_N^{[R]}(x_i) - f_N^{[R-1]}(x_i)| < \epsilon. \quad (6)$$

Whenever $|f_N^{[r]}(x_i)| < 10^{-11}$ for any r , condition (6) will always be used. Note that the values of the $(R-1)^{\text{th}}$ iteration will already be of the desired accuracy but the value of R will always be given in results since this iteration must be performed in order to check convergence. Note also that although two iterations may have converged to a certain accuracy, they need not have converged to the true solution.

2.3 APPLICATION TO LINEAR VOLTERRA EQUATIONS OF THE SECOND KIND

The theory of successive approximations as applied to linear Volterra equations is well documented ([16, 15, 17, 18]). It is known that the method converges for all values of λ when g and K are continuous functions at least square integrable. If g and K are taken to be continuous on the interval of definition and therefore bounded and satisfy

$$|g(x)| \leq m \quad a \leq x \leq b, \quad (7)$$

$$|K(x, y)| \leq M \quad a \leq x, y \leq b \quad (8)$$

then the Neumann series for $f(x)$ is bounded in the following way

$$\begin{aligned} |g(x) + \sum_{r=1}^{\infty} \lambda^r f_r(x)| &\leq m \sum_{r=0}^{\infty} |\lambda|^r M^r (b-a)^r / r! \\ &= m \exp[|\lambda| M(b-a)] \end{aligned} \quad (9)$$

where

$$f_r(x) = \int_a^x K(x, y) f_{r-1}(y) dy \quad (10)$$

and

$$f_0(x) = f^{[0]}(x) = g(x). \quad (11)$$

The Neumann series is absolutely and uniformly convergent. The

error which results by terminating the Neumann series at the R^{th} iteration satisfies

$$\begin{aligned} E_R(x) &= f(x) - f^{[R]}(x) \\ &= \sum_{r=R+1}^{\infty} \lambda^r f_r(x). \end{aligned} \quad (12)$$

It is readily seen that

$$|E_R(x)| \leq \frac{\rho^{R+1}}{(R+1)!} \exp(\rho) \quad (13)$$

where $\rho = |\lambda M(b-a)|$. Similar expressions may be obtained when the functions are taken to be square integrable. If the integrals are taken in the Lebesgue sense one can only say that the Neumann series is absolutely and uniformly convergent almost everywhere. Tricomi [15] uses the term almost uniformly convergent. Note that although convergence of the method is guaranteed for all values of λ it may be slow since the rate of convergence is governed by the size of the operator norm

$$\|K\|_2^2 = \iint |K(x,y)|^2 dx dy \quad (14)$$

The rate of convergence can however often be increased by the use of an accelerator (see chapter 4) or even by using a Gauss-Seidel type updating procedure.

2.3.1 THE ACCUMULATION OF QUADRATURE ERRORS IN SOLVING LINEAR VOLTERRA EQUATIONS

Excluding roundoff errors and the error incurred by terminating the iteration, the only errors which may manifest themselves will be those due to the quadrature rule. For the sake of generality assume that a quadrature rule Q of order N is used which approximates the integral $\int_a^x I(x,y)dy$. This information is displayed by

$$Q(N,a,x,I) \approx \int_a^x I(x,y)dy \quad (1)$$

where

$$Q(N,a,x,I) = \sum_{j=0}^N w_j(x) I(x,x_j) \quad (2)$$

Denoting the error by $E_Q(N,a,x,I)$ one obtains

$$\int_a^x I(x,y)dy = Q(N,a,x,I) + E_Q(N,a,x,I). \quad (3)$$

Let

$$E^{[r]}(x) = |f^{[r]}(x) - f_N^{[r]}(x)|, \quad r=1,2,\dots \quad (4)$$

and begin the iteration with

$$f_N^{[0]}(x_i) = g(x_i) = f^{[0]}(x_i), \quad i=0(1)N. \quad (5)$$

Then

$$E^{[0]}(x) = 0, \quad (6)$$

$$E^{[1]}(x) = \left| \int_a^x K(x, y) g(y) dy - Q(N, a, x, Kg) \right| \quad (7)$$

$$= E_Q(N, a, x, Kg),$$

$$E^{[2]}(x) = \left| \int_a^x K(x, y) f^{[1]}(y) dy - Q(N, a, x, Kf_N^{[1]}) \right|, \dots \quad (8)$$

$$E^{[r]}(x) = \left| \int_a^x K(x, y) f^{[r-1]}(y) dy - Q(N, a, x, Kf_N^{[r-1]}) \right|, \dots \quad (9)$$

The error between true and approximate r^{th} iteration can be recast in a slightly different fashion to give

$$E^{[r]}(x) = \left| \int_a^x K(x, y) f^{[r-1]}(y) dy - Q(N, a, x, Kf^{[r-1]}) + \right. \quad (10)$$

$$\left. Q(N, a, x, Kf^{[r-1]}) - Q(N, a, x, Kf_N^{[r-1]}) \right|.$$

Using (2) the following inequality may be produced.

$$E^{[r]}(x) \leq |E_Q(N, a, x, Kf^{[r-1]})| + \sum_{j=0}^N |w_j(x) K(x, x_j)| E^{[r-1]}(x_j). \quad (11)$$

By substituting for all previous errors $E^{[j]}(x)$, $j < r$ in (11) and noting that $E^{[0]}(x) = 0$, the error of the r^{th} iteration can be seen to satisfy

$$E^{[r]}(x) \leq |E_Q(N, a, x, Kf^{[r-1]})| +$$

$$\sum_{j=0}^N |w_j(x) K(x, x_j)| |E_Q(N, a, x_j, Kf^{[r-2]})| + \quad (12)$$

$$\sum_{j=0}^N |w_j(x) K(x, x_j)| \sum_{k=0}^N |w_k(x_j) K(x_j, x_k)| |E_Q(N, a, x_k, Kf^{[r-3]})| + \dots$$

A crude error bound may be formed using the following assumptions. Suppose that

$$i) \quad E_Q(N, a, x, Kf^{[r]}) < E, \quad \text{for all } r, x, \quad (13)$$

$$ii) \quad |K(x, y)| \leq W, \quad (14)$$

$$iii) \quad \sum_{j=0}^{\infty} |w_j(x)| \leq W \leq (b-a). \quad (15)$$

Then if $WK \neq 1$

$$E^{[r]}(x) \leq E \frac{1 - (WK)^{(r-1)}}{1 - WK} \quad (16)$$

and

$$E^{[r]}(x) \leq (r-2)E \quad (17)$$

if $WK = 1$.

Unless the coefficients of E in (16) or (17) are very much greater than an order of magnitude then the error $E^{[r]}(x)$ will

be less than ϵ^* . For example if

$$E < 10^{-2} \epsilon^*$$

then so long as the coefficient in (16) or (17) is less than 100, $E^{[r]}(x)$ will be less than ϵ^* .

2.3.2 GAUSS-SEIDEL UPDATING OF LINEAR VOLTERRA EQUATIONS

This method of calculating solution values which is explained in section 3.3.3 (DIG) is seen as a means of accelerating the rate of convergence of the successive approximations. Details of the first two iterations for the simple case $N=2$ clearly show why the method achieves its success. For simplicity let equation (2.2-2) be written in the form

$$f_i^{[r]} = g_i + \sum_{j=0}^N w_{ij} K_{ij} f_j^{[r-1]} \quad i=1(1)N, r=1, 2, 3, \dots \quad (1)$$

where

$$f_i^{[r]} = f^{[r]}(x_i), \quad w_{ij} = w_j(x_i), \quad K_{ij} = K(x_i, x_j), \quad g_i = g(x_i). \quad (2)$$

Then successive approximations using Gauss-Seidel updating take the following form for $N=2$:-

$$f_i^{[0]} = [g_i], \quad i=0(1)2, \quad (3)$$

$$f_0^{[1]} = [g_0],$$

$$f_1^{[1]} = [g_1 + \sum_{j=0}^2 w_{1j} K_{1j} g_j], \quad (4)$$

$$f_2^{[1]} = [g_2 + \sum_{j=0}^2 w_{2j} K_{2j} g_j] + w_{21} K_{21} \sum_{j=0}^2 w_{1j} K_{1j} g_j,$$

$$f_0^{[2]} = [g_0],$$

$$f_1^{[2]} = [g_1 + \sum_{j=0}^2 w_{1j} K_{1j} g_j + \sum_{j=1}^2 w_{1j} K_{1j} \sum_{k=0}^2 w_{jk} K_{jk} g_k] + w_{12} K_{12} w_{21} K_{21} \sum_{j=0}^2 w_{1j} K_{1j} g_j, \quad (5)$$

$$f_2^{[2]} = [g_2 + \sum_{j=0}^2 w_{2j} K_{2j} g_j + \sum_{j=1}^2 w_{21} K_{21} \sum_{k=0}^2 w_{jk} K_{jk} g_k] + w_{21} K_{21} \sum_{j=1}^2 w_{1j} K_{1j} \sum_{k=0}^2 w_{jk} K_{jk} g_k +$$

$$w_{22} K_{22} w_{21} K_{21} \sum_{j=0}^2 w_{1j} K_{1j} g_j + w_{21} K_{21} w_{12} K_{12} w_{21} K_{21} \sum_{j=0}^2 w_{1j} K_{1j} g_j, \dots$$

The terms in square brackets are those which would be obtained in ordinary iteration. The effect of Gauss-Seidel updating is to introduce terms into the series at earlier stages of the iteration than they would appear in the ordinary case. In practice it has been observed that points close to the initial point x_0 benefit less from the use of Gauss-Seidel updating (in fact the solution takes longer to converge in some examples) than those at the top of the range. It will be seen in section 2.5.1 that this does not apply to Fredholm equations. The reason for this situation is that although the solution value at x_0 of a Volterra equation is already exact at the 0th iteration taking the value g_0 it does not contribute a quadrature term to successive points. Hence for example $f_1^{[1]}$ is exactly the same in Gauss-Seidel and ordinary iteration. It therefore takes longer for the extra terms to appear. Gauss-Seidel iteration can be most succinctly expressed in matrix notation. Let \underline{W} be the matrix of quadrature weights combined

with kernel values. Thus

$$\underline{W} = \left[w_{ij} K_{ij} \right], \quad i, j = 0(1)N. \quad (6)$$

Note that $w_{0j} = 0$, $(j = 0(1)N)$. Let \underline{L} be the lower triangular matrix of \underline{W} excluding the diagonal. That is

$$\underline{L} = \left[w_{ij} K_{ij} \right] \quad i = 1(1)N, \quad j = 0(1)i-1, \quad (7)$$

all other elements being zero. Gauss-Seidel iteration can then be written as

$$\underline{f}^{[r]} = \underline{g} + (\underline{I} + \underline{L} + \underline{L}^2) \underline{W} \underline{f}^{[r-1]} \quad r = 1, 2, 3, \dots \quad (8)$$

The error analysis of the previous section applies equally to Gauss-Seidel updating as it does to ordinary successive approximations, provided that inequality (2.3.1-14) holds for integrands comprising the most recent solution values as well as those for the previous iteration.

2.3.3 RESULTS FOR SOME SIMPLE EXAMPLES

Baker[11] provides a set of simple linear test examples (p.748) for which the present scheme is appropriate. The equations are listed in appendix A0.1, nos.(1)-(7). Comparison is made with a step-by-step method involving combinations of the trapezium, Simpson's and the 3/8th rules ([11], p.775 Table 6.5), the TS38 rule for brevity. Baker (p.759) gives a detailed account of these particular methods which basically involve subdividing the range $[a, b]$ into N intervals of width h and then employing

simple quadratures to approximate the integral over $[a,b]$. Setting $x=kh$, $k=1,2,\dots,N$ and applying the earlier scheme produces a numerical approximation of the form

$$\tilde{f}(kh) = g(kh) + \lambda \sum_{j=0}^k w_{kj} K(kh, jh; f(jh)), \quad k=1(1)N \quad (1)$$

to the exact equation

$$f(x) = g(x) + \lambda \int_a^x K(x, y) f(y) dy. \quad (2)$$

The w_{kj} are the weights associated with each particular quadrature rule. Small values of k incur severe limitations on the obtainable accuracy at the corresponding points since only single or perhaps two-fold applications of the appropriate rules ($k=1$ necessitates the trapezium rule, $k=2$ requires an application of Simpson's rule or other three point formulae such as that provided by two applications of the Trapezium rule) can be used. These points will be called the starting values for the step-by-step method and should not be confused with starting values for iterative methods.

Results for $N=64$ ($h=1/32$) employing the TS38 rule appear in table 1. The table reveals that in general only the solution at a distance h from the initial point suffers from any marked lack of accuracy, exactly as would be predicted by analysis of the errors. The most accurate values occur in the middle of the range, becoming slightly less accurate towards the top of the interval. In contrast, the errors incurred by successive approximations remain more or less constant throughout the range. A glance at Table 2 also demonstrates that far fewer function

Table 1.
Some linear Volterra equations solved by the TS38 scheme

x eqn. no.	h	2h	0.25	1.00	1.75	2.00
1	2.6(-6)	1.1(-7)	1.3(-7)	3.0(-7)	6.5(-7)	8.5(-7)
2	4.0(-8)	3.6(-11)	2.2(-10)	6.4(-9)	3.6(-8)	4.8(-8)
3	5.0(-6)	6.6(-9)	4.6(-8)	2.1(-7)	3.7(-7)	4.2(-7)
4	5.1(-6)	6.6(-9)	4.7(-8)	2.4(-7)	5.9(-7)	7.7(-7)
5	3.3(-5)	2.8(-6)	3.9(-6)	1.3(-5)	3.5(-5)	4.8(-5)
6	3.8(-10)	3.6(-10)	1.8(-9)	1.4(-8)	4.6(-8)	6.4(-8)
7	2.5(-6)	1.0(-7)	7.8(-8)	4.2(-8)	4.2(-8)	4.6(-8)

evaluations are required in this approach. The new scheme requires $N(N+1)$ evaluations of the kernel and $(N+1)$ of the driving term making a total of $(N+1)^2$. The step-by-step method involves the calculation of $k+1$ values for each $f(kh)$, $k=1(1)N$. Including the N driving term evaluations this amounts to $N(N+5)/2$ function evaluations. For the same N the step-by-step approach clearly requires fewer function evaluations than successive approximations. However the successive approximations scheme only requires $N=16$ or equivalently 289 function evaluations to obtain full figure accuracy. In contrast the step-by-step approach produces results which are inferior to those of the new scheme by at least a factor of 10^2 rising to 10^4 in the starting values even when $N=64$. 2,208 function evaluations need to be carried out for this value of N , thus demonstrating that successive approximations combined with El-gendi's form of Clenshaw-Curtis quadrature is far more

Table 2.

Application of successive approximations to the linear Volterra equations

eqn no.	x=0.2928932		x=1.0		x=1.7071067		x=2.0	
	R	E	R	E	R	E	R	E
<u>NGRID=4, $\epsilon=1.0(-4)$</u>								
1	8	3.6(-4)	10	1.3(-3)	8	8.7(-4)	8	1.6(-4)
2	4	3.6(-3)	4	1.0(-3)	4	1.9(-3)	4	2.6(-3)
3	4	5.6(-4)	4	2.4(-3)	4	3.4(-5)	4	9.0(-4)
4	4	4.0(-4)	4	1.5(-3)	4	3.4(-5)	4	6.6(-4)
5	15	9.5(-2)	15	4.7(-1)	17	9.6(-1)	17	1.6
6	8	2.0(-4)	10	6.0(-4)	9	4.3(-4)	8	8.6(-4)
7	18	8.6(-4)	20	3.4(-3)	20	2.4(-3)	7	4.4(-4)
<u>NGRID=4, $\epsilon=1.0(-7)$</u>								
1	10	1.4(-8)	9	4.6(-8)	11	2.5(-8)	10	1.3(-7)
2	4	8.1(-6)	4	3.5(-6)	5	5.6(-6)	4	4.6(-8)
3	4	9.1(-8)	5	9.2(-8)	4	1.1(-8)	4	6.6(-9)
4	4	1.2(-8)	5	2.5(-8)	4	1.5(-8)	4	3.8(-9)
5	16	1.2(-5)	15	2.2(-5)	16	1.8(-5)	16	2.0(-5)
6	10	1.0(-8)	9	2.6(-8)	11	2.6(-8)	11	1.7(-8)
7	12	2.8(-7)	12	9.8(-7)	11	8.0(-7)	8	6.2(-8)
<u>NGRID=16, $\epsilon=1.0(-9)$</u>								
1	9	5.8(-11)	10	5.8(-11)	11	1.2(-10)	10	6.4(-10)
2	4	7.3(-12)	5	9.1(-12)	5	7.3(-12)	4	1.5(-11)
3	4	0.0	5	0.0	4	0.0	4	2.9(-11)
4	4	0.0	5	0.0	4	1.5(-11)	4	2.9(-11)
5	13	5.8(-10)	13	9.3(-10)	15	4.2(-9)	15	3.7(-9)
6	9	4.4(-11)	10	5.8(-11)	11	2.9(-11)	11	2.9(-11)
7	10	1.1(-10)	9	3.2(-10)	9	7.9(-10)	7	1.6(-9)

ϵ = absolute error requested, E = absolute error attained, R = minimum number of iterations required to satisfy $|f_N^{[R]}(x) - f_N^{[R-1]}(x)| < \epsilon$.

efficient than the step-by-step method, at least from this point of view. Multiplications and divisions can constitute a significant factor in the expense of a computation. Several may be necessary to perform just one of the above mentioned function evaluations, but for purposes of simplicity in the analysis a "function evaluation" will denote a general measure of the arithmetic work expended in finding the value of any function. In the present context it is additionally possible to make a straight comparison of the number of multiplications/divisions involved in calculations of the two schemes. For the sake of brevity they will simply be referred to as multiplications. The step-by-step algorithm needs $2 \sum_{k=1}^N (k+1) = N(N+3)$ multiplications. For $N = 64$ this amounts to 4,288 multiplications. The number of multiplications involved in evaluating the direct iterative scheme is $(N+1)(\sum_{j=1}^N R_j + N)$ where R_j is the number of iterations required to achieve convergence at each point x_j . This figure follows from the fact that $N(N+1)$ multiplications must initially be carried out in order to combine the terms $w_j(x_i)K(x_i, x_j)$, $(i=1(1)N, j=0(1)N)$ and then a further $N+1$ multiplications must be performed in order to complete the calculation of the quadrature at each point for each of the iterations. Taking $N=16$ and the computationally most expensive example (5) it can be seen that a total of $17\{(10+12+3 \times 13+5 \times 14+6 \times 15)+16\}=4029$ multiplications were necessary. Even this example was more economically solved by the successive approximations scheme than by the step-by-step method. Using the same calculations, equation (4) which was characteristic of several other examples required a mere 1445

multiplications which is approximately a third of the work required in the step-by-step approach.

It should be noted that the above analysis applies equally well to Gauss-Seidel iteration as it does to ordinary iteration. The values given in Table 2 were produced by means of the former iteration. The effects of Gauss-Seidel iteration predicted in section 2.3.2 can clearly be seen upon comparison of both types of iteration. Equation (5) will again be used as an example. The numbers of iterations required at each point of the 16 point scheme beginning with $x_1 = 1 - \cos(\pi/16)$ were 5, 7, 9, 10, 12, 13, 15, 16, 17, 18, 19, 20, 21, 22, 22, and 10, 12, 13, 13, 14, 14, 14, 13, 14, 14, 15, 15, 15, 15, 15, 15 for normal and Gauss-Seidel iteration respectively. It was not until the seventh point that Gauss-Seidel iteration performed better than the ordinary variety, but the total numbers of iterations 247 and 221 emphasise that Gauss-Seidel iteration was the more economical one to use. The same trend was observed in all the examples and Gauss-Seidel iteration has therefore been adopted as the preferred method of solution.

It must be pointed out that the equations considered contain only well behaved functions and therefore do not provide as stringent a test as might be required in order to prove the worth of the proposed new scheme. Also the step-by-step method could be improved upon by using a block-by-block method, so avoiding the use of a low order quadrature for the first few steps. However the step-by-step method acts as a familiar yardstick for comparisons. Many other equations

have been successfully solved by the method, including numerous interesting examples of physical significance. Among these are the integral equations for the slowing down of neutrons, [90] which provide a more rigorous test of the new scheme and are considered in detail in the next section

2.3.4 INTEGRAL EQUATIONS FOR THE SLOWING DOWN OF NEUTRONS

The example considered deals with the slowing down of neutrons by elastic scattering in an infinite non-absorbing moderating medium. For details see Weinberg and Wigner [51]. Of particular interest is the fact that the solution possesses a finite discontinuity thus providing a very stringent test of any quadrature scheme. Basically, Q_0 source neutrons per unit volume at energy E_0 are slowed down to energy E as a result of successive collisions with moderating nuclei of mass A . Defining the lethargy variable u by means of

$$u = \ln(E_0/E), \quad (1)$$

the collision density $F(u)$ is given by the integral equation

$$F(u) = \gamma Q_0 \exp(-u) + \gamma \int_0^u \exp(v-u) F(v) dv, \quad 0 < u < \epsilon \quad (2)$$

and by the equation

$$F(u) = \gamma \int_{u-\epsilon}^u \exp(v-u) F(v) dv, \quad u > \epsilon \quad (3)$$

In these equations

$$\gamma = (1-\alpha)^{-1} \quad (4)$$

where

$$\alpha = (A-1)^2 / (A+1)^2 \quad (5)$$

and

$$\epsilon = -\ln(\alpha) \quad (6)$$

represents the maximum lethargy gain per collision. The discontinuity occurs at $u=\epsilon$ and is of magnitude βQ_0 where

$$\beta = \alpha / (1-\alpha). \quad (7)$$

This occurs because the source neutrons may contribute directly to the collision density in the lethargy range $0 < u < \epsilon$, but not to the range $u > \epsilon$. The solution to equation (2) is readily seen to be

$$F(u) = \gamma Q_0 \exp(\beta u), \quad 0 < u < \epsilon \quad (8)$$

upon differentiation. Equation (3) is more difficult to treat, but can be found in analytical form throughout the range of definition by splitting the integral at the points $\epsilon, 2\epsilon, 3\epsilon, \dots$. Thus if F_r is the solution in $[(r-1)\epsilon, r\epsilon]$ then

$$F_{r+1}(u) = \gamma \left[\int_{u-\epsilon}^{r\epsilon} \exp(v-u) F_r(v) dv + \int_{r\epsilon}^u \exp(v-u) F_r(v) dv \right]. \quad (9)$$

Beginning with equation (8), known values for F_r may be

substituted into the first integral in (9). The equation may then be transformed to a differential equation and the solution of $F_{r+1}(u)$ found. This process is somewhat cumbersome and is not to be recommended. The classical solution was obtained by Placzek [52] using a recurrence relation approach involving the so called Placzek functions.

An alternative solution was obtained by Teichmann [53] and by Eidemann [54] using the Laplace Transform Convolution Theorem, the result being in the form

$$F(u) = \gamma Q_0 \exp(\beta u) + \gamma Q_0 \sum_{k=1}^{\infty} (-\beta)^k \exp(\beta z) \left[\frac{z^k}{k!} + \frac{z^{k-1}}{(k-1)!} \right] H(z), \quad (10)$$

where

$$z = u - ke$$

and H is the Heaviside function defined by

$$\begin{aligned} H(z) &= 1, & z > 0, \\ &= 0, & z < 0. \end{aligned} \quad (11)$$

Alternative forms of equations (2) and (3) exist and provide useful comparisons from the point of view of computational ease and efficiency. By introducing the slowing down density $q(E)$, representing the number of neutrons slowing down past energy E , [52] or [55], and on noting that, in the absence of absorption or leakage $q(E)$ must equal the source density Q_0 , it is possible to recast equations (2) and (3) in the simpler forms

$$F(u) = \gamma Q_0 + \beta \int_0^u F(v) dv, \quad 0 < u < \epsilon, \quad (12)$$

$$F(u) = Q_0 + \beta \int_{u-\epsilon}^u F(v) dv, \quad u > \epsilon. \quad (13)$$

The asymptotic collision density F may readily be obtained from (13) and is found to be

$$F = Q_0 / (1 - \beta \epsilon) \quad u \rightarrow \infty. \quad (14)$$

In practice the limiting value is rapidly attained for all values of A and after about three collisions the collision density is constant to within 1% even for the largest values of A . The maximum lethargy considered is 6ϵ where the collision density has sensibly reached its constant asymptotic form. Due to the non-standard form of equations (3) and (13) it is logical to subdivide the range of integration into successive collision density lethargy intervals $[0, \epsilon]$, $[\epsilon, 2\epsilon]$, ..., $[5\epsilon, 6\epsilon]$. This involves a particular example of the discretization D6 to be described in Chapter 3. In this case the positioning of the subintervals is made to coincide with these points and the integrals are also evaluated in a specialised way. Denoting each of the subinterval boundaries by u_i so that

$$u_i = i\epsilon, \quad i = 0(1)6, \quad (15)$$

and assuming that $u \in [u_{k-1}, u_k]$, $k = 2(1)6$, it is convenient to consider the integrals required in (3) and (13) as sums of two integrals as follows:-

$$\int_{u-\epsilon}^u \Psi(v) dv = \int_{u-\epsilon}^{u_{k-1}} \Psi(v) dv + \int_{u_{k-1}}^u \Psi(v) dv. \quad (16)$$

This form bears a strong resemblance to that used on Fredholm equations possessing "split kernels" in section 2.5.3. Using equations (3.4.1-7,8) one can employ the same sets of weights for both integrals appearing on the right hand side of (16). The grid points however will be different for each subinterval and on assigning a quadrature order N_l to interval l , they may be calculated via the usual linear transformation(2.2-3). Thus

$$u_{l,j} = \frac{u-\epsilon}{2} \left(\cos\left(\frac{j\pi}{N_l}\right) + 2l - 1 \right), \quad l=1(1)6, \quad j=0(1)N_l \quad (17)$$

Note that whenever $u=u_l$, the simpler definite integration weights (3.4.1-4,5) may be used.

2.3.4.1 RESULTS AND COMPARISON WITH OTHER SCHEMES

The slowing down densities were computed from the classical forms (2) and (3) or (12) and (13) for various moderator masses ranging from $A=2$ (deuterium) to $A=238$ (uranium). The relative accuracy of the results were tested with respect to Teichmann's analytical values. Table 3 gives the maximum minimum number of iterations required in any of the six lethargy intervals to attain the relative accuracy demanded. Note this is the number, R required to satisfy the inequality (2.2-5) and does not guarantee the accuracy of the final results to the true solution. Only relative errors at the last lethargy point $u=6\epsilon$ are quoted since the errors achieved throughout the total range

$[0, 6\epsilon]$ are of comparable magnitude. The values at this point are considered to be representative of the accuracy attained since they depend on all the previously computed $F(u)$ values for $u < 6\epsilon$. In spite of the finite discontinuity at $u = \epsilon$, rapid convergence is seen to be obtained over the complete lethargy range, particularly in the case of the alternative forms (12) and (13) of the integral equation. It is clear that this increased rate of convergence corresponds closely with the usual measure of convergence for the Neumann series provided by the norm [16]

$$\|K\|_2^2 = \int \int |K(u, v)|^2 du dv. \quad (18)$$

Table 3.
The neutron slowing down problem solved by successive approximations

	Moderator mass	Classical form (2)-(3)		Alternative form (12)-(13)	
	A	R	E	R	E
N=4	2	14	2.9(-3)	5	3.2(-7)
	4	9	4.1(-5)	6	4.3(-7)
	12	8	1.4(-5)	7	3.1(-6)
	16	8	3.5(-6)	8	7.2(-6)
	238	8	2.1(-6)	10	1.6(-6)
N=8	2	15	1.1(-9)	8	9.5(-11)
	4	12	4.7(-9)	9	6.4(-10)
	12	11	2.9(-10)	10	1.2(-8)
	16	11	2.5(-10)	11	1.4(-9)
	238	10	2.8(-10)	14	6.5(-8)

R = Maximum number of iterations required in any lethargy interval.

E = Relative accuracy in collision density (cf. (3.2-5)).

N = Order of quadrature in each lethargy interval.

The relative accuracy asked for was $\{=10^{-4}$ for N=4 and $\{=10^{-9}$ for N=8.

The lethargy intervals considered are $[0, \epsilon], [\epsilon, 2\epsilon], \dots, [5\epsilon, 6\epsilon]$.

Note that the results were obtained by using Gauss-Seidel type iteration.

Performing the required integrations over any of the subintervals reveals that the norm for equations (2) and (3) is

$$N_1 = \gamma \text{sh}(\epsilon) \quad (19)$$

and the corresponding norm for equations (12) and (13) is easily seen to be

$$N_2 = \beta \epsilon. \quad (20)$$

Table 4.
Norms of the kernels

A	ϵ	N_1	N_2
2	2.197	5.000	0.2747
4	1.022	1.889	0.5747
12	0.344	1.198	0.8422
16	0.250	1.142	0.8801
238	0.017	1.008	0.9916

N_1 = Norm for kernels in equations (2) and (3).

N_2 = Norm for kernels in equations (12) and (13).

ϵ = Lethargy interval width.

These norms are included in Table 4 and it is apparent that the variation is consistent with the number of iterations taken. This confirms that the $\|K\|_2$ norm is a reliable measure of the rate of convergence of the method. Again, the power of the Clenshaw-Curtis rules is demonstrated by the fact that accurate results are obtained according to specification (2.2-5) with errors of order 10^{-5} for N as low as $N=4$ in each lethargy interval except for the lowest

moderator masses. The accuracy attained for the smoother alternative forms (12) and (13) is correspondingly higher and is acceptable over the complete moderator range with this minimal number of quadrature points. A slight increase in the number of iterations required, accompanies the request for higher accuracy and $N=8$ points per interval produce acceptable accuracy even for the classical forms (2) and (3) even at low values of A where the lethargy interval $\epsilon = -\ln \alpha$ is largest.

As an alternative comparison, since the equations are linear, the traditional algebraic approach was used as a means of solution [1]. The presence of the discontinuity suggests that an equally spaced quadrature rule such as the trapezium rule would be most convenient. Investigations showed that using as many as $N=512$ points in each of the six lethargy intervals was still only capable of attaining errors $O(10^{-4})$. This compared most unfavourably with the results of the new method. Remembering that the trapezium rule can only be expected to achieve errors $O(h^3)$ more powerful quadrature rules such as the Clenshaw-Curtis prescription were used. Equivalent accuracy to successive approximations was then obtained. This would be expected as the direct iterative method is clearly related to the Nyström method in which the linear equations are solved by Jacobi's method. However the advantage would lie in solving nonlinear equations.

2.4 APPLICATION TO NONLINEAR VOLTERRA EQUATIONS OF THE SECOND KIND

In contrast to linear integral equations which have a simple structure, nonlinear equations can pose formidable problems both theoretically and computationally. However, a fairly complete theory does exist for certain types of equations. The most commonly quoted variety which also have many useful applications are the Hammerstein equations. See [84,18,15,56,81] for example. More general nonlinear equations have been studied by Golomb [85] amongst others. There are three problems associated with the solution of a nonlinear integral equation in general. They are the existence and uniqueness of a solution and the determination of a solution if it exists. It is quite possible for such equations to possess more than one solution. It is then feasible that approximate methods might converge to a value which is not a solution by oscillating between two genuine solutions. In certain circumstances an approximate method might even converge when in fact no solution exists. Convergence analysis is usually difficult for these problems and the approach in solving them will be largely empirical.

Baker [11] (p.686) maintains that the obvious numerical methods for solving nonlinear equations are adaptations of methods used in solving linear equations. Consequently, nonlinear systems of equations arise. Finding the solution of such systems is well known to be a difficult task. See for example Ortega and Rheinboldt [82], Collatz [73], Rall

[14]. Such problems are not encountered in the use of successive approximations.

Various conditions may be used which ensure convergence of the method of successive approximations and also provide theoretical bounds to the error incurred by terminating the iteration at the R^{th} stage. The following simple existence theorem suffices to provide a majorant to the Neumann series of the nonlinear integral equation [56],

$$f(x) = g(x) + \int_a^x K(x, y; f(y)) dy, \quad a \leq x \leq b: - \quad (1)$$

(a) $g(x)$ and $K(x, y)$ are integrable, continuous and therefore bounded in the range of definition, satisfying

$$|g(x)| < G, \quad a \leq x \leq b, \quad (2)$$

$$|K(x, y; z)| < M, \quad a \leq x, y \leq b, \quad (3)$$

and

(b) The kernel satisfies the Lipschitz condition

$$|K(x, y; z_1) - K(x, y; z_2)| \leq L |z_1 - z_2|. \quad (4)$$

Condition (3) additionally ensures that the solution is unique. The successive approximations are defined by

$$f^{[r]}(x) = g(x) + \int_a^x K(x, y; f^{[r-1]}) dy, \quad (5)$$

Beginning with

$$f^{[0]}(x) = g(x).$$

The solution is given by

$$f(x) = \lim_{r \rightarrow \infty} f^{[r]}(x). \quad (6)$$

This limit may be recast in the form

$$f(x) = f^{[0]}(x) + \sum_{r=1}^{\infty} [f^{[r]}(x) - f^{[r-1]}(x)]. \quad (7)$$

Using inequality (4) yields

$$|f^{[r]}(x) - f^{[r-1]}(x)| \leq L \int_a^x |f^{[r-1]}(y) - f^{[r-2]}(y)| dy, \quad (8)$$

whence

$$|f^{[r]}(x) - f^{[r-1]}(x)| \leq \frac{L^r |x-a|^r}{r!} \quad r=1, 2, 3, \dots \quad (9)$$

may be obtained. If $|x-a| < h$ then the series

$$F = G + \sum_{r=1}^{\infty} \frac{(Mh)^r}{r!} \quad (10)$$

bounds the solution. A bound on the error created by terminating the series (6) at the R^{th} stage is given by

$$E_R(x) = |f(x) - f^{[R]}(x)| = \left| \sum_{r=R+1}^{\infty} [f^{[r]}(x) - f^{[r-1]}(x)] \right|$$

$$\leq \sum_{r=R+1}^{\infty} \frac{(Mh)^r}{r!} = \frac{(Mh)^{R+1}}{R+1} \exp(Mh). \quad (11)$$

2.4.1 SUCCESSIVE APPROXIMATIONS APPLIED TO SOME NONLINEAR VOLTERRA INTEGRAL EQUATIONS

This section compares the use of successive approximations with a method to be found in Baker [11] on the two equations

$$f(x) = \exp(x^2) - x + x \exp\left(\frac{x^2}{2}\right) - x \int_0^x y [f(y)]^{1/2} dy \quad (1)$$

$$= \exp(x^2), \quad 0 \leq x \leq 4,$$

(A0.2-1) and

$$f(x) = 2 - \exp(x) + \int_0^x \exp(x-y) [f(y)]^2 dy \quad (2)$$

$$= 1 \quad 0 \leq x \leq 1.$$

(A0.2-2). Both appear in Baker [11]. See also Garey [57].

Baker [11] describes several variants of step-by-step methods for solving nonlinear Volterra equations. They are similar in detail to those described in section 2.3, with the difference that now two quadrature schemes are necessary and will be applied in an analogous manner to predictor-corrector methods for the solution of ordinary differential equations. The particular scheme given in [11] p.831, example (6.21) which was devised by Garey [57] will be used in comparisons with successive approximations. Briefly, the interval $[a,b]$ of definition is subdivided into N subintervals of width $h=(b-a)/N$. An initial value for the solution, $f^{[0]}$ at the points kh ($k=1(1)N$) is found via the discretization

$$\tilde{f}^{[0]}(kh) = \{g(kh) + \sum_{j=0}^{s_k} w_{kj}^p K(kh, jh; \tilde{f}(jh))\} + \sum_{j=s_k+1}^{k-1} w_{kj} K(kh, jh; f(jh)) \quad (3)$$

$$= \gamma_{s_k}^* + \sum_{j=s_k+1}^{k-1} w_{kj} K(kh, jh; \tilde{f}(jh)), \quad s_k \leq k-1$$

Iteration is then carried out at the point $x_k = kh$ until convergence occurs using the equation

$$\tilde{f}^{[r]}(kh) = \gamma_{s_k}^* + \sum_{j=s_k+1}^{k-1} w_{kj} K(kh, jh; \tilde{f}(jh)) + w_{kk} K(kh, kh; \tilde{f}^{[r-1]}(kh)), \quad (4)$$

$r=1, 2, \dots$

The inclusion of $\gamma_{s_k}^*$ in (2) is made possible by arranging that

$$w_{kj}^p = w_{kj}, \quad j=0, 1, \dots, s_k \leq k-1. \quad (5)$$

w_{kj}^p , ($k=1(1)N, j=0(1)k$) are the weights for the "predictor" formula and w_{kj} , ($k=1(1)N, j=0(1)k$) the weights of the "corrector" quadrature. The actual weights for Garey's scheme are given by

a) k even,

$$4w_{k0} = 4w_{k0}^p = w_{kk-3} = w_{kk-3}^p = w_{kj} = w_{kj}^p = 2w_{kj+1} = 2w_{kj+1}^p = \frac{4h}{3},$$

$$j=1, 3, 5, \dots, k-1 \text{ and}$$

(6)

$$6w_{kj+1} = w_{kk+1} = 2h, \quad w_{kk} = 0, \text{ whilst } 2w_{kk-2} = w_{kk-1} =$$

$$4w_{kk} = \frac{4h}{3} \text{ if } k \text{ is sufficiently large.}$$

b) k odd,

$$4w_{k0} = 4w_{k0} - w_{kk-4} - w_{kk-4}^p - w_{kj} - w_{kj}^p = 2w_{kj+1} - 2w_{kj+1}^p = \frac{4h}{3},$$

$j=1,3,5,\dots,k-6$ together with

(7)

$$w_{kk-3}^p = \frac{h}{3}, w_{kk-2}^p = w_{kk-1}^p = \frac{3h}{2}, w_{kk}^p = 0 \text{ and } w_{kk-3} = \frac{17h}{24},$$

$$w_{kk-2} = w_{kk-1} = 3w_{kk} = \frac{9h}{8} \text{ for } k \geq 3.$$

Again, rough estimates of the work involved in using either a predictor-corrector method or successive approximations can be made by comparing function evaluations or multiplications/divisions. The number of function evaluations necessary for the predictor-corrector methods is $\frac{3N(N+1)}{2} + \sum_{k=1}^N R_k$ compared with $N(1 + \sum_{j=0}^N R_j) + 1$ for successive approximations. This assumes that once a particular solution value has converged then all kernel evaluations using that value are stored. The multiplications involved in each scheme are $\sum_{k=1}^N \{2k - s_k + R_k\} - N$ for the predictor-corrector scheme and $(N+1) \sum_{k=1}^N R_k$ for the direct iterative approach. Garey's scheme in particular requires $16 + \frac{1}{2}(N-4)(N+16) + \sum_{k=1}^N R_k$ multiplications.

In practice the modified step-by-step methods will be prone to the same shortcomings as those described in section 2.3. Solution values at points close to the initial point will be difficult to calculate accurately as only low order quadratures may be used. As noted before, the repeated rules are also less accurate than those of Clenshaw-Curtis for comparable numbers of quadrature points. This becomes even more evident in the use of open predictor formulae since they

are in general less accurate than their counterparts of closed type. The process of iterating to convergence in "predictor-corrector" methods and restrictions arising from stability limitations tend to dictate the need for a small step size h . Consequently this implies that N must be large and the methods therefore require many function evaluations (see section 2.3).

2.4.2 RESULTS OF SUCCESSIVE APPROXIMATIONS APPLIED TO SOME NONLINEAR VOLTERRA INTEGRAL EQUATIONS

Results for the solution of the two equations (1) and (2) of the previous section by successive approximations using Clenshaw-Curtis quadrature appear in Table 5. Values of $N=4,8,16$ were used and the solution initially computed over the range $[0,1]$. For comparison see Table 6 which contains the results obtained using Garey's scheme [57] with $N=4,8,16,32$ and 64. It is apparent that even the Clenshaw-Curtis 4-point formula produces respectable errors and the 8-point formula achieves roughly the same order of accuracy as the 32 or 64 point formulae of Garey's method. The solution for (1) over the extended range $[0,4]$ was found by means of direct iteration using the Clenshaw-Curtis 32-point formula. Relative errors were all $O(10^{-9})$ at worst throughout the range and the calculated solution was exact at many points as it was at $x=4$. Considering that the solution increases rapidly from 1 at $x=0$ to $O(10^6)$ at $x=4$ the agreement is excellent. Garey's scheme still produced inferior results using as many as 256 points.

Table 5.

Results for direct iteration

eqn no.	x=0.1464466		x=0.5		x=0.8535534		x=1.0	
	R	E	R	E	R	E	R	E
<u>NGRID=4, $\epsilon=1.0(-4)$</u>								
1	2	8.0(-6)	3	6.2(-5)	4	3.1(-5)	3	7.1(-6)
2	5	3.9(-6)	8	4.1(-5)	10	3.9(-6)	9	1.6(-6)
<u>NGRID=8, $\epsilon=1.0(-7)$</u>								
1	4	5.5(-10)	4	3.9(-9)	5	2.0(-9)	4	5.9(-10)
2	8	2.9(-8)	10	4.8(-10)	11	1.8(-8)	11	1.4(-8)
<u>NGRID=16, $\epsilon=1.0(-9)$</u>								
1	4	1.4(-11)	5	1.1(-11)	5	0.0	5	0.0
2	7	3.8(-10)	10	7.3(-11)	11	1.5(-10)	11	2.0(-10)

NGRID = Number of grid and quadrature points.

ϵ = Relative error requested.

E = Relative error attained.

R = Minimum number of iterations required to satisfy $|f_N^{[R]}(x) - f_N^{[R-1]}(x)| < \epsilon |f_N^{[R]}(x)|$.

Once again these results demonstrate the power of the successive approximations approach. Further evidence of its suitability in solving Volterra integral equations was obtained in the study of the Lane-Emden equation [60,61]. The original equation appeared in the differential form

$$f''(x) + \frac{2}{x} f'(x) + f^v(x) = 0, \quad f(0) = 1, \quad f'(0) = 0, \quad (8)$$

where v takes real values. It describes the thermal behaviour of spherical gas clouds in gravitational equilibrium [61]. The equation (6) was transformed into the integral equation

Table 6.

Results for Carey's Method

	h		0.25		0.50		0.75		1.00	
	R	E	R	E	R	E	R	E	R	E
eqn. no.	<u>N=4, $\epsilon=1.0(-4)$</u>									
1			5	1.2(-4)	6	1.5(-5)	7	1.1(-4)	7	9.1(-5)
2			16	2.0(-3)	13	2.0(-3)	14	4.1(-3)	11	9.0(-3)
	<u>N=8, $\epsilon=1.0(-6)$</u>									
1	4	3.8(-6)	4	1.4(-7)	5	1.1(-6)	5	3.2(-6)	6	6.0(-4)
2	11	2.0(-4)	10	8.2(-5)	8	1.7(-7)	8	3.6(-4)	8	7.7(-4)
	<u>N=16, $\epsilon=1.0(-9)$</u>									
1	4	1.2(-7)	4	9.6(-9)	4	7.0(-8)	5	2.0(-7)	5	3.9(-7)
2	9	2.2(-5)	6	6.0(-6)	6	1.3(-5)	6	2.7(-5)	6	5.7(-5)
	<u>N=32, $\epsilon=1.0(-9)$</u>									
1	3	3.7(-9)	3	6.2(-10)	4	4.4(-9)	4	1.3(-8)	4	2.4(-8)
2	7	2.6(-6)	5	4.1(-7)	5	8.7(-7)	5	1.8(-6)	5	3.9(-6)
	<u>N=64, $\epsilon=1.0(-9)$</u>									
1	3	1.2(-10)	3	4.1(-11)	3	2.7(-10)	4	7.8(-10)	4	1.5(-9)
2	6	3.3(-7)	4	2.7(-8)	4	5.7(-8)	4	1.2(-7)	4	2.6(-7)

N = Number of quadrature points.

h = 1/N.

ϵ = Relative error requested.

E = Relative error attained.

R = Minimum number of iterations required to satisfy $|f_N^{[R]}(x) - f_N^{[R-1]}(x)| < \epsilon |f_N^{[R]}(x)|$.
This includes the predictor stage.

$$f(x) = 1 + \int_0^x \gamma(y-x) f^\gamma(y) dy, \quad 0 \leq x \leq 10 \quad (9)$$

by means of Fubini's variation of parameters technique [62] as described in Chapter 3. Simple analytical solutions are available in the cases $\nu=0,1$ and 5. Numerical solutions for other values of ν may be found in [60]. Of particular interest here is the fact that comparison may be made not only with standard methods for the solution of Volterra integral equations but also with standard techniques for the solution of initial value differential equations. The use of direct iteration was again compared with Garey's scheme in solving integral equation (9). Additionally, comparison of results was also made with those obtained by solving differential equation (8) by the powerful Runge-Kutta procedure.

In the linear case $\nu=1$, as many as $N=512$ ($h=0.0195$) points were used over the complete range $[0,10]$ in the step-by-step routine of section 3.3.3, producing relative errors of order 10^{-7} . The same order of accuracy was attained using Garey's predictor-corrector scheme on the nonlinear case $\nu=5$ with as many points. The Runge-Kutta implementation required $N=200$, ($h=0.05$) points to achieve relative errors of order 10^{-8} in the linear case and $N=400$, ($h=0.025$) to obtain comparable accuracy when errors of 10^{-9} were realised.

In comparison, successive approximations required only very few points. Using $N=32$ was sufficient to obtain errors of

order 10^{-10} in the linear case and of order 10^{-8} decreasing to 10^{-10} at $x=10$ in the nonlinear form. Convergence was achieved after at most only 4 iterations in the linear case and 6 iterations for the nonlinear examples. The number of iterations in Garey's scheme was 3 at each point.

So even here where both a Volterra integral equation solver and powerful initial value differential equation solver are available, the present method comes into contention as a possible alternative. It appears that this iterative scheme can provide rapid and accurate approximations to the solutions of Volterra equations whilst requiring comparatively very few function evaluations.

2.5 APPLICATION OF THE METHOD TO LINEAR FREDHOLM EQUATIONS OF THE SECOND-KIND

The numerical scheme originally proposed in section 2.2 is directly applicable to second kind Fredholm equations. The scheme may be implemented with the additional simplification that only the definite integration weights of El-Gendi [63] need be used. Therefore only a comparatively small amount of storage space is necessary. The rules associated with these weights belong to the class J^s for which the weights and abscissae are symmetrical about the mid-point of the range and also belong to the class J^+ which denotes the set of rules possessing only positive weights. Several useful results are connected with this type of rule. The fact that

$$\sum_{j=0}^N |w_j| = \sum_{j=0}^N w_j = (b-a) \quad (1)$$

is useful in proving that quadrature approximations converge to the correct values and providing error bounds. (See [11]). However the likelihood of success when coupled with the method of direct iteration is restricted by the well known limitations associated with the Neumann series of Fredholm equations. In general the equation

$$f(x) = g(x) + \lambda \int_a^b K(x,y) f(y) dy \quad (2)$$

only possesses a convergent Neumann series for values of λ satisfying the condition

$$\rho = |\lambda| \|K\|_2 < 1 \quad (3)$$

This condition imposes quite a severe restriction on the number of Fredholm equations which can be solved by direct iteration. Despite this disadvantage, condition (2) does benefit the method in acting as a guarantee. If the condition is satisfied then the method will always work. In contrast theory indicates that all Volterra equations satisfying conditions (2.3-7,8) may be solved by successive approximations. In practice some Volterra equations exist which also cannot be solved by the method. For example the equation

$$f(x) = 13e^x - 12 - 12 \int_0^x f(y) dy, \quad 0 \leq y \leq x \leq 2 \quad (4)$$

with solution

$$f(x) = e^x$$

satisfies the condition and yet successive approximations fluctuate so wildly that numerical calculations become impossible. Starting with $f^{[0]}(x) = 1$ for simplicity

$$f^{[1]}(x) = 13e^x - 12 - 12x, \quad (5)$$

$$f^{[2]}(x) = 13e^x - 12 - 12\{13[e^x - 1] - 12x - 6x^2\}, \dots \quad (6)$$

Over the range $[0, 2]$ they take the highly oscillatory values $f^{[1]}(1/2) = 3.4$, $f^{[2]}(1/2) = -1.76$, $f^{[1]}(1) = 11.3$, $f^{[2]}(1) = -28.7$, $f^{[1]}(2) = 60.1$, $f^{[2]}(2) = -336.6$ and ensuing iterations increase in magnitude until numerical overflow occurs before the true solution emerges. Obviously, either successive terms in the series must be bounded and tend to zero or they must cancel out to give the correct solution. For equation (3) $p = 12\sqrt{2}$ and the successive terms will be unbounded. The situation is very similar to the calculation of e^{-x} via its series expansion for large values of x .

When condition (3) is satisfied it will be seen that the direct iterative scheme when coupled with Clenshaw-Curtis quadrature as described in section 2.2 produces results in a fast and efficient manner. The theoretical error incurred by terminating the iterative process at the R^{th} iteration may be

bounded by

$$|E_R(x)| \leq \frac{A \|g\|_2 \rho^{R+1}}{(1-\rho)}, \quad (6)$$

where A satisfies

$$\int_a^x |K(x,y)|^2 dy < A^2, \quad a < x < b. \quad (7)$$

2.5.1 GAUSS-SEIDEL UPDATING FOR FREDHOLM EQUATIONS

From the practical aspect, implementation of Gauss-Seidel type successive approximations for Fredholm equations is exactly the same as that described for Volterra equations in section 2.3. However, due to the presence of definite as opposed to indefinite integrals, certain subtle differences appear in the

analysis. The basic matrix equation (2.3.2-8)

$$\underline{f}^{[r]} = \underline{g} + (\underline{I} + \underline{L} + \underline{L}^2) \underline{W} \underline{f}^{[r-1]}, \quad r=1,2,3,\dots \quad (1)$$

still holds, but now

$$\begin{aligned} \underline{W} &= [\underline{w}_{Nj} K_{ij}] = [\underline{w}_j K_{ij}], & 1, j=0(1)N, \\ &= \underline{K} \underline{D} \end{aligned} \quad (2)$$

where \underline{K} is the matrix of kernel values and \underline{D} is the diagonal matrix of definite integration quadrature weights (3.4.1-4,5).

\underline{L} takes the same form as before with the obvious modification to the weights. The iterates of Fredholm equations are different from those of Volterra equations, not only because the weights are fewer in number but also because the value for $f(x_0)$ involves a quadrature in the Fredholm case. This has the effect of introducing successive terms into the series even earlier than in the Volterra case. The second iteration already contains several terms occurring in the third and fourth iterations as well as some of the fifth iteration for the usual Neumann series.

2.5.2 RESULTS AND COMPARISONS FOR SOME SIMPLE LINEAR TEST

EXAMPLES

The method of successive approximations has the distinct advantage of being directly applicable to nonlinear equations. One of the main aims of this thesis is to develop schemes capable of solving such equations. However in order to make fair judgement of the method's merits many linear equations were also solved. The results for a selection of Fredholm equations solved by successive approximations may be found in Table 7. Equations (1), (2), (3), (5), (6), (7) of appendix A0.3 are dealt with. The table includes relative errors achieved, the number of iterations required and other relevant information. λ was set equal to 0.5 in those equations in which it occurred. This ensured that it was within the norm bounds for each example. The results were obtained by updating the solution in the Gauss-Seidel manner described in the previous section.

The Gauss-Seidel updating was also carried out by reversing the order in which function values were updated. Similar results were obtained. As predicted by the analyses of sections 2.2 and 2.5.1, Gauss-Seidel updating has a more beneficial effect on Fredholm equations. However in certain examples a reduction of as few as 4 iterations was observed whereas in others a far greater saving could be achieved. Gauss-Seidel updating reduced the number of iterations in example (7) which is Love's equation from electrostatic theory [11] by 17 iterations from 25 to 12 iterations. There does not appear to be any way of predicting which equations will be most amenable to this modification although it appears from example 2 that when the series is slowly convergent because λ is close to the spectral radius then it will have least effect. Alternative modifications which overcome this problem will be explained in Chapter 4. All the results do however indicate that Gauss-Seidel updating should always be used. The figures appearing in Table 7 show that direct iteration can produce the results to linear Fredholm equations using a minimal number of quadrature points and very few iterations. The low number of quadrature points compares very favourably with those required by many schemes used in the past. Results in Baker [11] concerning Love's equation reinforce this point. By using the trapezium rule only 3-figure agreement with the true result could be achieved using 64 points. A scheme based on the classical Romberg rule required 33 points to deliver 4-figure accuracy and 65 to achieve only 5 figure accuracy. These results were typical of the findings derived by testing these schemes with the equations used in Table 7 and many others. Apart from the relatively poor results concerning

Table 7

Results for Gauss-Seidel direct iteration

eqn. no.	x_0		x_1		x_2		x_3		x_4	
	R	E	R	E	R	E	R	E	R	E
<u>$N=4, \epsilon=1.0(-4)$</u>										
1	1	0.0	8	7.0(-9)	8	6.9(-9)	8	4.1(-9)	7	8.2(-9)
2	26	1.2(-4)	26	1.3(-4)	26	1.4(-4)	25	1.3(-4)	25	1.2(-4)
3	5	5.5(-10)	5	3.1(-10)	5	7.0(-11)	5	3.9(-11)	5	4.9(-11)
5	1	0.0	12	5.6(-9)	11	6.2(-9)	12	3.0(-9)	11	5.6(-9)
6	1	0.0	14	1.3(-3)	13	1.3(-3)	13	1.3(-3)	13	1.3(-3)
7	12	0.0	12	7.0(-9)	12	6.9(-9)	11	4.1(-9)	11	8.2(-10)
<u>$N=8, \epsilon=1.0(-7)$</u>										
1	1	0.0	10	0.0	9	7.3(-11)	9	2.8(-11)	9	0.0
2	30	1.1(-10)	30	4.3(-11)	30	1.1(-10)	29	1.2(-10)	29	1.2(-11)
3	6	1.2(-11)	6	0.0	6	0.0	6	1.3(-11)	6	0.0
5	1	0.0	13	4.3(-11)	12	1.2(-10)	12	3.5(-11)	12	3.3(-11)
6	1	0.0	14	1.3(-4)	14	1.3(-4)	14	1.3(-4)	13	1.3(-3)
7	12	0.0	13	7.0(-9)	12	6.9(-9)	11	4.1(-9)	11	8.2(-10)
<u>$N=16, \epsilon=1.0(-9)$</u>										
1	1	0.0	9	2.1(-11)	9	1.1(-11)	9	0.0	8	2.4(-11)
2	28	6.1(-10)	28	5.5(-10)	28	4.3(-10)	27	6.2(-10)	27	4.9(-10)
3	6	1.2(-11)	6	0.0	6	0.0	6	0.0	6	0.0
5	1	0.0	11	2.7(-10)	12	1.5(-11)	11	2.4(-11)	11	2.7(-10)
6	1	0.0	13	1.6(-5)	13	1.6(-5)	13	1.6(-5)	12	1.6(-5)
7	12	0.0	12	7.0(-9)	11	6.9(-9)	11	4.1(-9)	11	8.2(-10)

N = Number of quadrature points.

ϵ = Relative error requested.

E = Relative error attained.

R = Minimum number of iterations required to satisfy $|f_N^{[R]}(x) - f_N^{[R-1]}(x)| < \epsilon |f_N^{[R]}(x)|$.

$x_i = ((a-b)\cos\frac{i\pi}{4} + a + b)/2$, $i=0(1)4$ where $[a, b] = [0, 1]$ for all equations except number (2) for which $[a, b] = [0, 7/2]$. Note that $f_0(x) = g(x)$.

equation (2) the only other equation to cause concern was equation (6). It is not surprising however that as many as 64 points were required to achieve full figure accuracy since the kernel contains a square root term which possesses a singular derivative.

2.5.3 SPLIT KERNEL INTEGRAL EQUATIONS

Many Fredholm integral equations take the form

$$f(x) = g(x) + \int_a^x K_1(x, y; f(y)) dy + \int_x^b K_2(x, y; f(y)) dy, \quad (1)$$

where K_1 and K_2 give the forms of the integral for $a \leq y \leq x$ and $x \leq y \leq b$ respectively. Equation (1) has close connections with boundary value problems. For example

$$f(x) = x \sin(1) + (1-x) \int_0^x y f(y) dy + x \int_x^1 (1-y) f(y) dy \quad (2)$$

is the counterpart of the differential equation

$$f''(x) + f(x) = 0, \quad f(0) = 0, \quad f(1) = \sin(1) \quad (3)$$

and the kernel

$$\begin{aligned} K(x, y) &= (1-x)y, & 0 \leq y \leq x \\ &= x(1-y), & x \leq y \leq 1 \end{aligned}$$

is the Green's function [16] of the parent differential equation (3).

2.5.3.1 A MODIFIED NUMERICAL SCHEME

Whenever the kernels $K_1(x,y)$ and $K_2(x,y)$ are defined for all x,y in $[a,b]$ it is possible to evaluate the integrals occurring in (2.5.3-1) in a manner very similar to that described in section 2.3.4 on the solution of integral equations for the slowing down of neutrons. In this instance both integrals are calculated over the same interval as x ranges through all the values between a and b . Assuming that the usual linear transformations will be carried out in order to map the interval $[-1,1]$ onto $[a,b]$, it is readily seen that the associated general split integral

$$I = \int_{-1}^1 \Psi_1(s) ds + \int_1^1 \Psi_2(s) ds \quad (1)$$

may be approximated by the sums

$$I \approx \sum_{j=0}^N w_j(t) \Psi_1(t_j) + \sum_{j=0}^N w_j(-t) \Psi_2(t_j) \quad (2)$$

where the $w_j(t)$ and t_j , $j=0(1)N$ are the weights and abscissae (3.4.1-2,3), (see El-gendi [63]).

2.5.3.2 RESULTS FOR A SELECTION OF EQUATIONS

The equations which were solved are to be found in Appendix 0.3. They are (A0.3-14,15,16,20). $N=8$ was sufficient to

ensure accuracy of all the quadratures to an accuracy of order 10^{-10} or better in all the equations. The maximum number of iterations required were $R=23$, 10 and 12 for equations (14), (15) and (16) respectively to attain relative errors $O(10^{-10})$ whereas the successive approximations were terminated after $R=30$ iterations in the case of equation (20). An order of accuracy 10^{-6} had been achieved up to this point. The rates of convergence are governed by the magnitude of the kernel norms but increased rates can be achieved by the application of an accelerator to the Neumann series (see Chapter 4). It will be seen that full figure accuracy can then be obtained even in the case of equation (20) after comparatively few iterations.

2.6 APPLICATION TO NONLINEAR FREDHOLM EQUATIONS

Many of the examples which have been tested are of the Hammerstein type

$$f(x) = \int_a^b K(x, y) F[y; f(y)] dy. \quad (1)$$

Tricomi [15] gives the following conditions which ensure that successive approximations converge to a solution of equation (1). They are:-

(i) the function

$$A^2(x) = \int_a^b K^2(x, y) dy \quad (2)$$

exists almost everywhere in $[a, b]$ and is summable there;

(ii) the function $F[y, u]$ uniformly satisfies the Lipschitz condition

$$|F[y; u_1] - F[y; u_2]| \leq C(y) |u_1 - u_2|; \quad (3)$$

(iii) the function $F[y, 0]$ belongs to the class L_2 .

Under these assumptions, the sequence of successive approximations

$$f_{r+1}(x) = \int_a^b K(x, y) F[y; f_r(y)] dy, \quad r=1, 2, 3, \dots \quad (4)$$

converges almost everywhere to a solution of (1) provided that

$$\int_a^b A^2(x) C^2(x) dx = M^2 < 1. \quad (5)$$

By making use of the Lipschitz condition (4) and Schwarz inequality (2.1-1) it is easy to obtain an estimate of the error involved in terminating the sequence at the r^{th} term. It

is

$$\begin{aligned} |f(x) - f_r(x)| &\leq cA(x) \sum_{k=r+1}^{\infty} M^k \\ &= \frac{cA(x) M^{r+1}}{(1-M)} \end{aligned} \quad (6)$$

where

$$c^2 = \int_a^b F^2[y; 0] dy. \quad (7)$$

2.6.1 RESULTS FOR A SELECTION OF NONLINEAR FREDHOLM EQUATIONS

Numerous nonlinear equations have been solved by means of successive approximations. Many of these equations arise from physical problems for which solutions are not available in a simple functional form. For these equations actual solution values will be presented for various fitting degrees NGRID. Wherever possible results from previous work will be included so that a comparison of the number of figures in agreement can be made.

Example 1

As a simple introductory example, equation (A0.4-7) was solved. Using the free term as initial guess convergence to the solution $f(x)=x$ occurred in 17 iterations using a value of NGRID as low as 4. The greatest relative error was $O(10^{-10})$. Gauss-Seidel iteration had the effect of reducing the number of iterations by 11.

Example 2

Equation (A0.4-8) is of the Hammerstein form (2.6-1) and it can be seen that condition (2.6-3) is not satisfied. Successive approximations should therefore not converge and this is indeed found to be the case. Nevertheless the use of Gauss-Seidel

iteration did induce convergence albeit to the null solution.

Example 3

Similar considerations apply to equations (A0.4-9a,9b) which are deriveable from the differential equation for a chemical reactor (equation (3.3.2-9)), [75,91]. As pointed out by Lapidus [75] serious numerical instabilities arise in its conventional solution by "shooting" methods, Fox [92]. As a result, the equation has recieved considerable attention with a view to overcoming these difficulties. For example, Pakes and Storey [91] proposed a variational method of solution and presented results in representative cases which are of interest for comparison purposes. In the more interesting case the integral equations are nonlinear and in fact several forms are available depending on how the original differential operator is decomposed in its conversion to integral form. The method of section 3.3.2 can be used and for example values of $a=-3$ and $b=1/2$ did produce a convergent Neumann series. However Fubini's variation of parameters technique (Tricomi [15]) described in section 3.3.4 is preferable.

Conventionally the original differential equation is solved by the "shooting" method [92], incorporating an initial value procedure such as Runge-Kutta. However, stability analysis, as described by Lambert [86], demonstrates the presence of serious numerical instability for large values of p . It is necessary to integrate the equation backwards from $x=1$ to $x=0$ to obtain an accurate solution. Even then, the step size has to be reduced to

$h=0.00625$ in combination with about 8 "shots" to obtain nine figure accuracy in $f(x)$. The results of this conventional solution of the differential equation are presented in Table 9 for reference purposes.

Table 9
Runge-Kutta solution of Eq. (3.3.2-9) starting at $x=1$

$f(x)$				
x	$p=1$	$p=10$	$p=50$	$p=100$
0.0	0.636784102	0.877469378	0.965175227	0.981426978
0.5	0.503903768	0.507475307	0.503092675	0.501721671
1.0	0.457588686	0.370512001	0.342481909	0.338054037

Step size $h = 0.00625$.

Number of shots (n) to achieve 9 figure accuracy is $n=8$ for $p=10, 50, 100$ and $n=3$ for $p=1$.

By way of comparison, calculations were carried out based on the alternative forms (A0.4-9a, 9b) for the p values specified in Table 9. As anticipated, serious convergence difficulties arise. For example, the form (A0.4-9b), due to the presence of the factor p in the kernel the sequence of successive approximations diverges for all values of p . However the use of Gauss-Seidel iteration does induce convergence at least in the case $p=1$.

Some improvement is realised by selecting form (A0.4-9a) with its more natural kernel. A measure of the convergence is provided by the norm which equals

$$k^2 = \frac{1}{2} + \frac{1}{2} p^{-1} - [1 - \exp(-2p)](4p^2)^{-1} \quad (1)$$

so that k varies between 1 as $p \rightarrow 0$ and $1/\sqrt{2}$ as $p \rightarrow \infty$. There is therefore the possibility of convergence for larger values of p here, especially since some assistance is provided by the presence of the term $2f^2(y)$ with $|f| < 1$. This prediction is born out by the results depicted in Table 10 which demonstrates the number of iterations required to achieve an accuracy of 10^{-6} in the $f(x)$ values for various values of p . The starting point in each case was $f^{[0]} = 1$. As predicted by equation (1), there is a slight decrease in the number of iterations required as p increases. However, the compensating disadvantage is that, because of the presence of the term $\exp(-py)$ in the kernel of (A0.4-9a), the number of quadrature points required has to be increased. Thus, at $p=1$, 8 Clenshaw-Curtis points are adequate for accurate numerical integration, but, at $p=100$, the number has to be increased to 32. The automatic scheme D8 of Chapter 3 was used.

Table 10
Number of iterations (M) required
to achieve relative error $\epsilon = 10^{-6}$ in
the $f(x)$ values.

p	1	10	50	100
M	-	48	16	15

For smaller values of p the iterative sequence diverges and, even at $p=10$, $M=48$ iterations were necessary to achieve the specified accuracy. Gauss-Seidel iteration did improve this situation slightly.

Example 4

This next example deals with a variant of the the H-equation (A0.4-1), (see Chandrasekhar [71], Rall [14], Stibbs and Weir [70] and Baker [11]). The H-equation occurs in the study of radiative transfer in planetary and stellar atmospheres and has as its solution the H-functions which were originally introduced by Ambartsumian and later developed by Chandrasekhar. The general form for this equation is

$$f(x) = 1 + xf(x) \int_0^1 \frac{\psi(y)f(y)dy}{x+y} \quad (2)$$

where $\psi(x)$ is usually a polynomial of even order in x satisfying the condition

$$\int_0^1 \psi(x)dx < \frac{1}{2} . \quad (3)$$

The particular form solved refers to isotropic scattering in a semi-infinite plane-parallel atmosphere where $\psi(x) = \lambda/2$ and λ is the particle albedo. In their paper Stibbs and Weir [70] give values of $f(x)$ for $\lambda=0(0.05)1$. Values for $\lambda=0.5$ will be given in this section and results for higher values of λ may be found in Chapter 5.

Values of the solution (or relative errors) will be given for the two endpoints and the mid-point of the range in each case since these points which occur in the Clenshaw-Curtis scheme are usually the only ones to correspond to points in other works.

Results for equation (A0.4-1) are presented in Table 11 for $N=4,8,16,32$. Results obtained from Stibbs and Weir's paper [70] are also included.

Table 11
Results for the H-equation (A0.4-1) by successive approximations.

N	x_1	R	x_2	R	x_3	R	x_4	R
4	1.09296377	6	1.18763202	7	1.23673213	7	1.25117995	8
8	1.09325016	10	1.18773189	10	1.23683921	10	1.25125563	10
16	1.09325656	13	1.18773493	13	1.23684280	14	1.25125933	13
32	1.09325670	13	1.18773511	13	1.23684300	13	1.25125955	14
*	—		1.187735		—		1.251259	

N is the number of grid and quadrature points.

R is the number of iterations to attain convergence.

$x_i = \cos(i\pi/4)$, $i=1(1)4$

* Results in bottom line are from [70].

It is clear that even a quadrature of order 8 is sufficient to produce an accuracy of six figures which represents a very inexpensive means of the obtaining the solution. Similar results were also obtained for the related equations (A0.4-2,3,4). The numbers for equation (A0.4-4) are the inverses of the actual numbers obtained. All values were obtained by using Gauss-Seidel iteration although this only had a small beneficial effect, producing a saving of at most three iterations. (See section 4.5.2 for quicker iterations on equation (A0.4-1)).

The final two examples demonstrate the use of successive approximations applied to integro-integral equations as obtained in Chapter 3.

Example 5

In this example equation (A0.4-11) is treated and results may be compared with those obtained upon using the related form

Table 12
Results for the solution of equation (A0.4-11)

x_0		x_1		x_2		x_3		x_4	
R	E	R	E	R	E	R	E	R	E
1	0.0	13	6.7(-8)	7	1.5(-8)	2	1.8(-8)	8	0.0

N = Number of quadrature points=32

E = Relative error requested=1(-7).

E = Relative error attained.

R = Minimum number of iterations required to satisfy $|f_N^{[R]}(x) - f_N^{[R-1]}(x)| < \epsilon |f_N^{[R]}(x)|$.

$x_l = ((a-b)\cos\frac{l\pi}{4} + a + b)/2$, $l=0(1)4$ where $[a,b]=[0,12]$.

(A0.2-8) which achieved equivalent accuracy with the same number of points but at the expense of a maximum of 25 iterations

Example 6

Results for the famous Blasius equation [25]

$$2f'''(x) + f(x)f''(x) = 0, \quad f(0)=f'(0)=f'(\infty)-1=0 \quad (4)$$

are presented in Table 13. They were obtained from the use of successive approximations applied to the integro-integral

Table 13
The Blasius equation solved as an integro-integral equation

N	x_1	R	x_2	R	x_3	R	x_4	R
16	0.226884125	13	2.30575782	13	5.10770541	12	6.27924001	12
32	0.226884117	13	2.30575779	12	5.10770541	12	6.27924019	14
*	—		2.30576		—		6.27923	

N is the number of grid and quadrature points.

R is the number of iterations to attain convergence.

$x_i = \cos(i\pi/4)$, $i=1(1)4$

* Results in bottom line are from Howarth [150].

equation (A0.4-12). It was found to be adequate to let infinity be $x=8$.

The results of the previous sections reveal that successive approximations, when implemented in the various forms described here is a very useful tool for the solution of many integral equations. However the convergence difficulties associated with some equations, especially those of Fredholm type has made the search for alternative ways of using the method necessary. The findings of the subsequent research are contained in Chapter 4. Prior to this, brief information relating to basic theory and numerical details of successive approximations and subsequent methods will be given in the next chapter.

CHAPTER 3

SOME BASIC THEORY

3.1 INTRODUCTION

In this chapter two main topics are considered. The first is the link between ordinary differential equations and integral equations which provides the source of many physical examples for later chapters. Secondly the way in which quadrature and interpolation formulae will be incorporated into numerical schemes will be explained more fully, since they form the basis from which schemes will be developed. Many of the equations which have been solved involve L_2 -functions for which an extensive theory exists. This avoids the more restrictive condition that functions be continuous (and hence bounded), although at least piecewise continuity is necessary if automatic routines which employ interpolation are used. It also enables one to employ the very useful Schwarz inequality.

$$\left[\int_a^b f(x)g(x)dx \right]^2 \leq \int_a^b f^2(x)dx \int_a^b g^2(x)dx. \quad (1)$$

If the function $f(x)$ is square integrable then its norm $\|f\|_2$ must satisfy

$$\|f\|_2^2 = \int_a^b f^2(x)dx < P < \infty \quad (2)$$

and similarly if the kernel $K(x,y)$ is to be such a function then its norm $\|K\|_2$ must satisfy

$$\|K\|_2^2 = \int_a^b \int_a^b |K(x,y)|^2 dx dy < M < \infty. \quad (3)$$

Because of the close connections of differential and integral equations it will be of interest to attempt the solution of problems in both differential and integral forms and to compare the relative merits of either means of solution.

The next section gives ways of transforming differential to integral equations.

3.2 CONVERSION OF DIFFERENTIAL TO INTEGRAL EQUATIONS

It is well known that differential equations may be expressed as integral or integro-differential equations [15,16,18,25]. Problems which are expressible as differential equations can therefore equally well be expressed in integral equation or integro-differential equation form. In the past, preference has frequently been given to the differential equation option since such a wealth of theory and methods of solving these equations exists. However the integral equation formulation often gives better insight into the nature of solutions and may also admit a wider class of solutions to the problem. For example, the differential equation for the position y of a particle at time t after leaving the origin of coordinates and travelling at a velocity $V(t)$ may be written as

$$\dot{y}(t) = V(t), \qquad y(0)=0. \qquad (1)$$

The equivalent integral equation is

$$y(t) = \int_a^x V(s) ds. \quad (2)$$

If V were the discontinuous function

$$\begin{aligned} V(t) &= 1 & 0 \leq t \leq t_1, \\ &= 0 & t_1 \leq t \leq T, \end{aligned} \quad (3)$$

then the solution to equation (2) would be

$$\begin{aligned} y(t) &= t & 0 \leq t \leq t_1, \\ &= t_1 & t_1 \leq t \leq T. \end{aligned} \quad (4)$$

This is a perfectly acceptable solution to the original problem but not to equation (1) since $y(t)$ is not differentiable at $t=t_1$. Additionally, the process of integration produces a beneficial smoothing effect which can be particularly important when badly behaved functions are involved. A further advantage is derived from the fact that boundary or initial conditions are built into the integral equation.

These points suggest that it could be advantageous to pose problems in integral equation form. Some of the work in the thesis will be devoted to discovering whether this form provides a competitive alternative to solving problems via the differential equation approach. Since many problems that

one encounters in the literature already appear as differential equations, it will be useful to possess techniques for converting differential to integral or integro-differential equations. Preferably one would like to obtain integral equations in order to avoid the restrictive properties of derivatives. Many methods exist for carrying out this process, but a brief word of caution is necessary. Careful analysis of the converted equation should always be made. The simple differential equation

$$f'(x) = f^2(x) + 1, \quad f(0) = 0 \quad (5)$$

gives rise to the integral equation

$$f(x) = x + \int_0^x f^2(y) dy. \quad (6)$$

The solution to these two equations is

$$f(x) = \tan(x) \quad (7)$$

which becomes undefined at $x = \pi/2$. If the range of definition includes $x = \pi/2$ and the method used involves sampling the integrand over the whole range then it will break down.

Differential equations may frequently be transformed to various different forms of integral equation. This can depend on the method used or the way in which the original differential equation is decomposed. In certain circumstances the decomposition may be arbitrary but in others, boundary

conditions may dictate which decompositions are appropriate, (see equations (14)-(24) and (31)-(34) in section 3.2.4 and the accompanying text). Some methods for transforming differential equations to integral equations together with examples will be outlined next.

3.2.1 FIRST ORDER EQUATIONS

Equations of this type can be immediately converted by integration of both sides of the equation. The equation

$$\frac{df(x)}{dx} = K(x;f(x)) \quad f(a)=A \quad (1)$$

is equivalent to the Volterra integral equation

$$f(x) = A + \int_0^x K(x;f(y))dy. \quad (2)$$

3.2.2 EQUATIONS OF ORDER N WITH CONSTANT COEFFICIENTS

The notation $F(D)f$ will be used to define the equivalence

$$F(D)f = (a_0D^n + a_1D^{n-1} + \dots + a_{n-1}D + a_n)f \quad (1)$$

$$= a_0 \frac{d^n f}{dx^n} + a_1 \frac{d^{n-1} f}{dx^{n-1}} + \dots + a_{n-1} \frac{df}{dx} + a_n. \quad (2)$$

The general n^{th} order equation will be written as

$$\begin{aligned}
 F(D)f &= Q(x; f, f', f'', \dots, f^n) \\
 &= Q(x; \underline{f}')
 \end{aligned}
 \tag{3}$$

where Q is a function, possibly nonlinear of x, f and its n derivatives. Without loss of generality a_0 may be assumed to be non-zero. The terms contained in equation (3) may be transferred from one side to the other in order to facilitate the transformation. The following two standard methods are applicable [22].

i) Denoting the inverse operator by $1/F(D)$ so that

$$\frac{1}{F(D)} \cdot F(D)f(x) = f(x) \tag{4}$$

and factorising the operator $F(D)$, yields

$$f(x) = \frac{1}{(D-\alpha_1)} \cdot \frac{1}{(D-\alpha_2)} \cdot \dots \cdot \frac{1}{(D-\alpha_n)} Q(x; \underline{f}'). \tag{5}$$

The operators $1/(D-\alpha_r)$, $r=n, n-1, \dots, 1$ are then applied in turn to yield

$$f(x) = e^{\alpha_1 x} \int_0^x e^{(\alpha_2 - \alpha_1)x} \dots \int_0^x e^{(\alpha_n - \alpha_{n-1})x} \int_0^x e^{-\alpha_n x} Q(x; \underline{f}') dx^n \tag{6}$$

It is possible to add the constants of integration to (6) but it proves to be far simpler to find the complementary function by inspection and so obtain the primitive.

ii) Alternatively $1/F(D)$ may be expressed as the sum of n partial

fractions

$$\frac{1}{F(D)} = \frac{N_1}{(D-\alpha_1)} + \frac{N_2}{(D-\alpha_2)} + \dots + \frac{N_n}{(D-\alpha_n)}. \quad (7)$$

Hence

$$f(x) = \sum_{j=1}^n N_j \int_0^x e^{\alpha_j(x-y)} Q(y; f') dy + \sum_{j=1}^n c_j \frac{x^{n-j}}{(n-j)!} \quad (8)$$

and the constants c_j are obtainable from the boundary conditions. Alternatively one can find the complementary function and then calculate the relevant constants.

As an illustration of the use of methods i) and ii) consider the equation, [65, 91],

$$\frac{1}{p} \frac{d^2 f}{dx^2} - \frac{df}{dx} - R[f] = 0 \quad (9)$$

with boundary conditions

$$f(0) - \left. \frac{1}{p} \frac{df}{dx} \right|_{x=0} = 1, \quad (10)$$

$$\left. \frac{df}{dx} \right|_{x=1} = 0 \quad (11)$$

and

$$R[f] = 2 f(x) \quad (12)$$

or

$$R[f] = 2 f^2(x). \quad (13)$$

Equation (9) describes the material balance equation for the fraction of reactant, $f(x)$, remaining in a tubular reactor with axial diffusion. p is known as the Peclet number and its size greatly influences the ease with which solutions may be obtained [65]. This equation will be studied in detail later in Chapter 5.

The integral equations are derived as follows. Rewriting (9) in the standard form

$$\begin{aligned} \frac{d^2 f}{dx^2} + a \frac{df}{dx} + bf &= (a+p) \frac{df}{dx} + pR[f] + bf \\ &= Q(x, f, f') \end{aligned} \quad (14)$$

where a, b are constants to be chosen. Letting the roots of the auxiliary equation be α and β the complementary function may be written as

$$f(x) = A e^{\alpha x} + B e^{\beta x} \quad (15)$$

where A and B will be determined later. Method (i) produces the primitive in the form

$$f(x) = A e^{\alpha x} + B e^{\beta x} + e^{\alpha x} \int_0^x e^{(\beta-\alpha)y} \int_0^y e^{-\beta z} Q(z; f, f') dz dy. \quad (16)$$

By changing the order of integration equation (16) becomes

$$f(x) = A e^{\alpha x} + B e^{\beta x} + \int_0^x \left[\frac{e^{\beta(x-y)} - e^{\alpha(x-y)}}{(\beta-\alpha)} \right] Q(y; f, f') dy. \quad (17)$$

which is exactly the equation that method (ii) would produce. Noting that Q contains the derivative of f one may integrate by parts to obtain

$$f(x) = Ae^{\alpha x} + Be^{\beta x} + \int_0^x [K(x,y)\{pR[f]+b\} + W(x,y)f(y)]dy, \quad (18)$$

where

$$K(x,y) = [e^{\beta(x-y)} - e^{\alpha(x-y)}]/(\beta-\alpha) \quad (19)$$

and

$$W(x,y) = [\beta e^{\beta(x-y)} - \alpha e^{\alpha(x-y)}]/(\beta-\alpha). \quad (20)$$

Imposing the boundary conditions (10) and (11) one finds that A and B satisfy the equations

$$A = \frac{p(\beta+p+a)e^{\beta} - (\beta+a)[C+(p+a)D]}{(\alpha+p+a)(\beta+a)e^{\alpha} - (\beta+p+a)(\alpha+a)e^{\beta}}, \quad (21)$$

$$B = \frac{-p(\alpha+p+a)e^{\alpha} - (\alpha+a)[C+(p+a)D]}{(\alpha+p+a)(\beta+a)e^{\alpha} - (\beta+p+a)(\alpha+a)e^{\beta}}, \quad (22)$$

where

$$C = \frac{1}{(\beta-\alpha)} \int_0^1 W(1,y)\{pR[f]+b\} + L(1,y)\{a+p\}f(y)dy, \quad (23)$$

$$D = \frac{1}{(\beta-\alpha)} \int_0^1 [K(x,y)\{pR[f]+b\} + W(1,y)\{a+p\}]f(y)dy \quad (24)$$

and

$$L(x, y) = [\beta^2 e^{\beta(x-y)} - \alpha^2 e^{\alpha(x-y)}]. \quad (25)$$

The roots α and β satisfy the equations

$$\alpha = (-a + \sqrt{a^2 - 4b})/2, \quad (26)$$

$$\beta = (-a - \sqrt{a^2 - 4b})/2 \quad (27)$$

and

$$\alpha + \beta = -a, \quad \alpha\beta = b. \quad (28)$$

Far simpler forms may be derived from equation (9), but the point of transforming (9) to equation (18) and its associated equations is that this exhibits a general decomposition and subsequent transformation of the equation. The constants a and b may be chosen to aid in the convergence properties of the method. (see section 2.6.1). Note that equation (18) is nonlinear in $f(x)$ even when $R[f] = 2f(x)$.

3.2.3 EQUATIONS OF ORDER N WITH VARIABLE COEFFICIENTS

The differential equation takes the form

$$\begin{aligned} F(D)f &= a_0(x)D^n f(x) + a_1(x)D^{n-1} f(x) + \cdots + a_n(x)f(x) \\ &= R(x) \end{aligned} \quad (1)$$

where R and a_i , ($i=0(1)n$) are functions of x alone. Lovitt [17] uses the substitution

$$\frac{d^n f}{dx^n} = \varphi(x) \quad (2)$$

and successively integrates to obtain the solution from the integral equation

$$a_0(x)\varphi(x) + \int_0^x K(x,y)\varphi(y)dy = \Phi(x) \quad (3)$$

where

$$K(x,y) = \sum_{j=0}^{n-1} \frac{(x-y)^j}{j!} a_{j+1}(y) \quad (4)$$

and

$$\Phi(x) = R(x) - \sum_{j=1}^n a_j(x) \sum_{i=j}^{n-1} c_{n-i} \frac{x^{j-1}}{(j-1)!} \quad (5)$$

The constants c_j are the constants of integration to be obtained from the boundary conditions.

The solution follows from the equation

$$f(x) = \frac{1}{(n-1)!} \int_0^x (x-y)^{n-1} \varphi(y) dy + \sum_{j=0}^{n-1} \frac{x^j}{j!} c_j \quad (6)$$

which is the result of integrating (2) n times. As an example consider the Lane-Emden equation [60,25,90] which describes the thermal behaviour of spherical gas clouds in gravitational equilibrium.

$$\frac{d^2 f}{dx^2} + \frac{2}{x} \frac{df}{dx} + f^\nu(x) = 0 \quad (7)$$

where

$$f(0) = 1, \quad \left. \frac{df}{dx} \right|_{x=0} = 0 \quad (8)$$

and the choice $\nu=1$ is made for illustrative purposes. Substituting $d^2f/dx^2=\varphi(x)$ produces the equations

$$\frac{df}{dx} = \int_0^x \varphi(y) dy + c_0, \quad (9)$$

$$f(x) = \int_0^x (x-y)\varphi(y) dy + c_0 x + c_1. \quad (10)$$

By using the boundary conditions and substituting terms into the original differential equation one obtains the equation

$$\varphi(x) = -1 - \frac{1}{x} \int_0^x (x^2 - xy + 2) \varphi(y) dy. \quad (11)$$

Limitations in this approach are at once apparent. Equation (11) still needs to be evaluated once equation (10) has been solved and if $\nu > 1$, then a non-standard form of nonlinear integral equation will result involving unnecessary extra computation. A far better technique is that attributed by Tricomi [15] to Fubini (see also [22] and Bellman[107]). This method creates integral equations using Lagrange's technique of variation of parameters.

3.2.4 A METHOD BASED ON VARIATION OF PARAMETERS

Given the differential equation

$$F(D)f(x) = \sum_{r=0}^n a_r(x) \frac{d^{n-r} f(x)}{dx^{n-r}} = R(x) \quad (1)$$

where R and $\{a_r(x)\}$, $(r=0(1)n)$ are functions of x , the first step is to rearrange the equation so that the complementary function for part of the equation can be found analytically. Suppose that the form

$$\sum_{r=0}^n A_r(x) \frac{d^{n-r} f(x)}{dx^{n-r}} = \sum_{r=0}^n B_r(x) \frac{d^{n-r} f(x)}{dx^{n-r}} + R(x) \quad (2)$$

$$= Q(x; \underline{f}'), \quad A_0(x), B_0(x) \neq 0.$$

has been obtained for which the general solution to the homogeneous equation

$$\sum_{r=0}^n A_r(x) \frac{d^{n-r} f(x)}{dx^{n-r}} = 0 \quad (3)$$

may be found explicitly in a simple manner. Let $\{\varphi_j(x)\}$, $(j=1(1)n)$ be n linearly independent solutions of (3). In order to find the solution of equation (2) one then assumes a solution of the form

$$f(x) = \sum_{j=1}^n c_j(x) \varphi_j(x) \quad (4)$$

where the functions $c_j(x)$, $(j=1(1)n)$ will be chosen in

accordance with certain conditions. On differentiating (4) n times and imposing the conditions

$$\sum_{j=1}^n c_j'(x) \varphi_j^r(x) = 0, \quad r=0(1)n-1, \quad (5)$$

substitution of the resulting expressions for $f^r(x)$, ($r=0(1)n$) into (2) produces the additional equation

$$\sum_{j=1}^n c_j'(x) \varphi_j^{n-1}(x) = Q(x; \underline{f}')/A_0(x). \quad (6)$$

Equations (5) and (6) may conveniently be expressed in matrix form as

$$\underline{\Phi}(x) \underline{c}'(x) = \underline{q}(x; \underline{f}') \quad (7)$$

where

$$\Phi_{ij}(x) = \varphi_j^{(i)}(x), \quad i=0(1)n-1, \quad (8)$$

$$j=1(1)n,$$

$$\underline{c}'(x) = [c_1'(x), c_2'(x), \dots, c_n'(x)]^T, \quad (9)$$

and

$$\underline{q}(x; \underline{f}') = [0, 0, \dots, 0, Q(x; \underline{f}')/A_0(x)]^T. \quad (10)$$

Since the functions $\varphi_j(x)$, ($j=1(1)n$) are linearly independent, their Wronskian is nonvanishing. The system (7) may therefore be solved uniquely for \underline{c}' . Thus

$$\underline{c}'(x) = \underline{\Phi}^{-1}(x) \underline{q}(x; \underline{f}'). \quad (11)$$

If $c_j(a)=a_j$, $j=1(1)n$, then the system of differential equations (11) may be converted to the system of integro-differential equations

$$\underline{c}(x) = \underline{a} + \int_a^x \underline{\Phi}^{-1}(y) \underline{g}(y; \underline{f}') dy \quad (12)$$

If $\varphi(x)$ is the vector of linearly independent solution values $[\varphi_1(x), \varphi_2(x), \dots, \varphi_n(x)]^T$, then the solution to equation (2) is obtained by taking the inner product with $\underline{c}(x)$ given by (12). The differential equation (1) is therefore equivalent to the integro-differential equation

$$\begin{aligned} f(x) &= \underline{c}(x) \cdot \varphi(x) \\ &= [\underline{a} + \int_a^x \underline{\Phi}^{-1}(y) \underline{g}(y; \underline{f}') dy] \cdot \varphi(x) \end{aligned} \quad (13)$$

In practice, it is frequently possible to transform (13) into a pure integral equation either via integration by parts or by solving a set of simultaneous integral equations for the functions $c_j(x)$, ($j=0(1)n$). The following simple example illustrates these points. Consider the equation

$$f''(x) + 2f(x)f'(x) = 0, \quad f(0)=0, f(12)=1. \quad (14)$$

If equation (14) is decomposed in the form

$$f''(x) = -2f(x)f'(x) \quad (15)$$

then two linearly independent solutions of the homogeneous equation

$$f''(x) = 0 \quad (16)$$

are $f_1(x)=x$ and $f_2(x)=1$. Upon trying a solution of the form

$$f(x) = c_1(x)x + c_2(x), \quad (17)$$

differentiating twice and imposing a condition equivalent to conditions (5) one obtains

$$f'(x) = c_1(x) \quad (18)$$

together with

$$c_1'(x)x + c_2'(x) = 0 \quad (19)$$

and

$$f''(x) = c_1'(x), \quad (20)$$

or equivalently

$$c_1'(x) = -2f(x)f'(x). \quad (21)$$

Integration of equations (19) and (21) and the appropriate substitutions in (17) produces the integro-differential equation

$$f(x) = \frac{x}{12} + 2 \int_0^{12} K(x,y)f(y)f'(y)dy, \quad (22)$$

where

$$\begin{aligned} K(x,y) &= (1-x/12)y & 0 \leq y \leq x, \\ &= (1-xy/12) & x \leq y \leq 12. \end{aligned} \quad (23)$$

For this particular problem a pure integral equation may be obtained using integration by parts. This may be done either directly to equation (22) or to equations (19) and (21) and repeating the above process. The resulting equation is

$$f(x) = \frac{x}{12} + \int_0^x \left[\frac{x-1}{12} \right] f^2(y) dy + \frac{x}{12} \int_x^{12} f^2(y) dy \quad (24)$$

Certain differential equations do not yield integro-differential equations which are amenable to integration by parts. The equation

$$f'''(x) + f(x)f''(x) = 0, \quad f(0) = f'(0) = f'(\infty) - 2 = 0 \quad (25)$$

which is a version of the Blasius equation [25,147] occurring in boundary layer theory provides such an example. A means of overcoming this difficulty is to solve the set of integro-differential equations corresponding to the equations (11) by substituting for f and its derivatives in terms of the functions $c_j(x)$, ($j=0(1)n$). Referring to the previous example, equations (17) and (18) can be substituted into equations (19) and (21) to produce the system

$$c_1(x) = \frac{1}{12} - \frac{1}{6} \int_0^x K(y; c_1, c_2) dy + 2 \int_x^{12} W(y; c_1, c_2) dy, \quad (26)$$

$$c_2(x) = 2 \int_0^x K(y; c_1, c_2) dy.$$

where

$$K(y; c_1, c_2) = y c_1(y) [y c_1(y) + c_2(y)] \quad (27)$$

and

$$W(y; c_1, c_2) = \left[\frac{1-y}{12} \right] c_1(y) [y c_1(y) + c_2(y)]. \quad (28)$$

Once values for $c_1(x)$ and $c_2(x)$ have been found they may be substituted into (17) to obtain the solution.

Yet another alternative in solving integro-differential equations involves the creation of a system of integral equations for the derivatives of the solution [119]. Take an m^{th} order integro-differential equation in the form

$$f(x) = g(x) + \int_a^b K(x, y; f(y), f'(y), \dots, f^{(m)}(y)) dy. \quad (29)$$

Then upon making the assumption that both g and K possess sufficient derivatives with respect to x in the region of definition, one may form the following set of $m+1$ integral equations

$$f^{(j)}(x) = g^{(j)}(x) + \int_a^b K^{(j)}(x, y; f'(y), \dots, f^{(m)}(y)) dy. \quad (30)$$

This system may then be solved for the unknowns $f^{(j)}(x)$, $j=0(1)m$, thus yielding the solution to the original problem $f^{(0)}(x)=f(x)$. Methods for the solution of systems of integral equations occur in Chapter 7 where comparisons are made between various means of solving problems.

The preceding paragraphs illustrate various options which may be employed in effecting the transformation of a differential equation. The particular form which is finally obtained may be dictated by the equation itself. The equation (14) is an approximate form of the equation

$$f''(x) + 2f(x)f'(x) = 0, \quad f(0)=0, f(\infty)=1 \quad (31)$$

whose solution is $f(x)=\tanh(x)$. Clearly the form of solution chosen in equation (17) would not fit the boundary condition $f(\infty)=1$. A different decomposition of the original equation therefore becomes necessary. The form

$$f''(x) + f'(x) = f'(x)(1 - 2f(x)) \quad (32)$$

provides a suitable starting point, since the complementary function to the left-hand side of (27) suggests a trial function given by

$$f(x) = A(x)e^{-x} + B(x). \quad (33)$$

The integral equation resulting from this approximation is,

$$f(x) = 1 - e^{-x} + \int_0^x e^{y-x} [f(y) - f^2(y)] dy \quad (34)$$

which fortuitously turns out to be a Volterra equation.

Many other differential equations are defined over an infinite range. If the boundary conditions at infinity prove difficult to build into an integral equation, then an alternative might be to assume that the limiting values are attained at some finite point $x = X$. An integral equation can then be created which contains the variable X . Solutions to the equation with increasing values of X are then found until successive values of X produce the same solutions. This approach will of course only work if the contribution to the integral over $[X, \infty)$ which has been neglected is indeed negligible.

Take for example the simple equation

$$f''(x) - f(x) = 0, \quad f(0)=1, f(\infty)=0. \quad (35)$$

On replacing the condition at infinity by a condition at some distant point b so that $f(b)=B$ the integral equation

$$f(x) = [1 + (B-1)\frac{x}{b}] + [\frac{x-1}{b}] \int_0^x f(y) dy + x \int_x^b [\frac{y-1}{b}] f(y) dy \quad (36)$$

may be obtained. If one then assumes that $f(x)=B=0$ for all $b > X$ one obtains the equation

$$f(x) = 1 - \frac{x}{X} + \left[\frac{x-1}{X} \right] \int_0^x y f(y) dy + \int_x^X \left[\frac{y-1}{X} \right] f(y) dy \quad (37)$$

which can be solved for increasing values of X .

In concluding the sections on the conversion of differential to integral equations, one further process will be mentioned. Reference is made to Ames [25] p.118.

3.2.5 THE CONSTRUCTION OF INTEGRO-INTEGRAL EQUATIONS

In the previous section reference was made to the Blasius equation where it was mentioned that unless one resorts to the solution of a system of integral equations, then only an integro-differential equation is obtainable from the original differential form. The beneficial smoothing effect of integration has already been alluded to and Ames presents a method which introduces extra integrals into the converted equation, rather than derivatives in order to maintain these advantages. The resulting equations may be regarded as integro-integral equations. This technique can be employed if a pair of dependent variables involving the solution and its derivatives satisfy a relationship of the form

$$\frac{d^p f(x)}{dx^p} = F\left[\frac{d^q f(x)}{dx^q}, x\right] \quad 0 \leq q < p. \quad (1)$$

Both the Blasius equation and equation (3.2.4-31) satisfy (1). In order to achieve the desired results one integrates equation (1) $p-q$ times. This yields

$$\frac{d^q f(x)}{dx^q} = \int_0^x \int_0^{x_1} \dots \int_0^{x_{p-q-1}} P\left[\frac{d^q f}{dy^q}, y\right] dy dx_{p-q-1} \dots dx +$$

$$\sum_{j=1}^{p-q} A_j \frac{x^{j-1}}{(j-1)!} \quad (2)$$

$$= \frac{1}{(p-q-1)!} \int_0^x (x-y)^{(p-q-1)} P\left[\frac{d^q f}{dy^q}, y\right] dy +$$

$$\sum_{j=1}^{p-q} A_j \frac{x^{j-1}}{(j-1)!} \quad (3)$$

The boundary conditions may be used to find the constants A_j . Once equation (3) has been solved for $d^q f/dx^q$, the solution $f(x)$ may be found by a q -times repeated integration. Consider equation (3.2.4-31) for example. This can be written as

$$\frac{df'}{dx} / f' = -2f. \quad (4)$$

Whence

$$f'(x) = A \exp(-2 \int_0^x f(y) dy). \quad (5)$$

The constant A cannot be obtained until the next step. Integrating once more from 0 to x gives

$$f(x) = A \int_0^x \exp(-2 \int_0^y f(z) dz) dy + B \quad (6)$$

and imposing the boundary conditions yields

$$f(x) = \frac{\int_0^x \exp(-2 \int_0^y f(z) dz) dy}{\int_0^\infty \exp(-2 \int_0^y f(z) dz) dy}. \quad (7)$$

The solution to equation (7) is most efficiently obtained if the infinite integral is evaluated in two parts over $[0, x]$ and $[x, \infty)$ since the first integral comprises the numerator.

The solution to the more interesting Blasius equation is derived in a similar way and can be shown to take the form

$$f(x) = \frac{2 \int_0^x (x-y) \exp[-\int_0^y f(z) dz] dy}{\int_0^\infty \exp[-\int_0^y f(z) dz] dy} . \quad (8)$$

These integral equations can be conveniently solved by the method of successive approximations. (See section 2.7.3).

These techniques for generating integral equations were used in the thesis for finding convenient examples with alternative methods of solution for comparison. Because of the advanced development of ordinary differential equation methods, it would not be envisaged to make such a conversion of an initial value problem. However, boundary value problems may well be more easily solved as integral equations than as differential equations.

Before concluding this chapter some of the underlying quadrature and interpolation for the numerical schemes is outlined. This work is used repeatedly in later chapters and results in the construction of suitable discretizations.

3.3 QUADRATURE AND INTERPOLATION STRATEGIES

The basic numerical devices required in solving integral equations will be quadrature rules, interpolation formulae and methods for solving sets of linear equations. The need for quadrature rules is obvious and methods which reduce integral equations to sets of linear equations, such as expansion techniques and the basis set approach clearly need a means of solving them. It is hoped to develop automatic schemes for the solution of integral equations in the sense that the method will take account of the behaviour of the various functions involved in the equations. It will be seen that interpolation is a useful tool in this connection.

All numerical schemes for the solution of integral equations involve discretization of the equation in some form and are therefore related to the original schemes proposed by Nyström [7]. The general Urysohn equation of Fredholm or Volterra type

$$f(x) = g(x) + \int_a^x K(x,y;f(y))dy \quad (1)$$

where the upper limit of integration takes the appropriate value may be approximated by

$$f_N(x_i) = g(x_i) + \sum_{j=0}^N w_j(x_i) K(x_i, x_j; f_N(x_j)), \quad i=0(1)N. \quad (2)$$

in its simplest form. The integral in equation (1) has been replaced by a quadrature rule of order N and possessing weights $w_j(x_i)$. The general quadrature depicted in (2)

depends on $N+1$ weights and abscissae since the subscripts run from 0 to N . The Clenshaw-Curtis rule is of this type but other rules such as the Patterson rules depend only on N weights and abscissae. In order to avoid complicated explanation both types of rule will either be said to be of order N or be N -point rules, it being assumed that the reader is aware of the differences. If equation (1) is of Fredholm type then only the weights $w_j(x_N)$, $(j=0(1)N)$ will be used. If it is of Volterra type then the subscript i need only run from 1 to N . The approximate value of the solution is denoted by the addition of the subscript N and the value of the true solution at any point x_i , $(i=0(1)N)$ is given by

$$f(x_i) = f_N(x_i) + E_Q(N, a, X, K_N). \quad (3)$$

Here K_N signifies that the kernel K contains approximate values of the solution and E_Q denotes the quadrature error which depends on the quadrature order N , limits of integration a and X (which might be the fixed limit b or the variable limit x , depending on the type of equation) and of course K_N .

3.3.1 CHOICE OF QUADRATURE RULE

Choice of quadrature rule depends on numerous factors. On the one hand simplicity might be a desirable property. Rall [3] for example uses the simple technique of Riemann sums in solving certain integral equations. Alternatively, the need for high accuracy might lead one to use Gaussian quadratures

[2], whereas the desire for a progressive routine might suggest the use of the Clenshaw-Curtis integrator [4] or the pseudo-Gaussian quadratures of Patterson [5]. Special properties of the integral equation will also govern the choice. The problems caused by singular kernels may be alleviated in some instances by the use of suitable weight functions. If a Volterra equation possesses a kernel which is not defined for $y > x$ as in the case $K(x,y) = \ln(x-y)$ then it is necessary to employ a rule which samples the integrand only in the range $[a,x]$. This is easily done of course by performing definite integrals over this range, but then quadrature abscissae required in any two intervals $[a,x_1]$ and $[a,x_2]$ in the evaluation of the solution at the points x_1 and x_2 will not generally correspond to points at which the solution has already been updated.

Step-by-step methods (see [11] and also comparisons in the next chapter) employ repeated rules to overcome this problem but the order of such rules is usually low, thus requiring large numbers of points to obtain reasonable accuracy.

The development of automatic routines which are adaptive in the sense that they will adjust to the behaviour of the functions involved makes the use of progressive rules such as those of Clenshaw and Curtis or Patterson advisable. Ideally such schemes would be able to evaluate integral equations more accurately and efficiently by being able to concentrate update points in areas where functions are badly behaved and use less points in areas where the functions are smooth. It certainly seems possible

that greater accuracy can be obtained, but due to the overheads involved in deciding whether and where complications exist, it seems unlikely that greater efficiency can easily be achieved. For the sake of high accuracy it is considered to be acceptable that efficiency should be sacrificed to a certain degree. Fortunately the two progressive rules which have been used extensively throughout the research possess a high order of accuracy [6].

This quantity is measured in terms of a quadrature rules' ability to integrate polynomials of a certain degree. The highest degree for which the quadrature error can be made zero is the order of accuracy. The Patterson rules [5] have an order of accuracy $(3N+1)/2$ which is comparable to the order attainable with true Gaussian formulae for which the order is $2N-1$ for an N -point rule. This is especially so for low values of N ($N=3, 7, 15$) which frequently prove to be adequate in evaluating accurate integrals. This fact certainly aids in producing efficient schemes.

The Clenshaw-Curtis rules [4] prove to be more accurate than their order would suggest and O'Hara and Smith [49] have shown through extensive tests that these rules are nearly as accurate as Gaussian quadratures.

The progressive property of these two sets of rules is also a very valuable asset since this minimises the number of new integrand evaluations necessary whenever the quadrature order needs to be increased. The points which have already been

used simply interlace with the points for subsequent values of N . Tests have indicated that Patterson's rules are usually marginally more accurate than the Clenshaw-Curtis quadratures for comparable numbers of points. They do however not possess good error estimates like the Clenshaw-Curtis schemes ([4], [49]) and more computation is often necessary to check accuracy than in the latter rules.

Equations (A2-1)-(A2-9) involve considerable computational expense. Fortunately the series involved in these equations may be summed in an alternative manner to produce a quadrature formula in the standard form,

$$I_1(x;f) = \int_{-1}^x f(y)dy = \sum_{j=0}^N w_j(x) f(x_j) \quad (1)$$

where $x_j = -\cos(j\pi/N)$, $j=0(1)N$, (see El-gendi [63]). Note that the Clenshaw-Curtis rules can be used to evaluate indefinite integrals in the sense that sample points are used which lie outside the range of integration $[a,x]$ but within the total range of definition of the equation $[a,b]$. This proves to be a very useful device in many schemes. Patterson's rules do not possess this facility.

The general form of the weights $w_j(x)$, $j=0(1)N$ of the Clenshaw-Curtis rules is

$$w_j(x) = \frac{1}{N} \left\{ \sum_{k=1}^{N-2} \frac{1}{k} [T_k(x) - (-1)^k] [T_{k-1}(x_j) - T_{k+1}(x_j)] + \frac{1}{(N-1)} [T_{N-1}(x) + 1] [T_{N-2}(x_j) - (-\frac{1}{2})^j] + \right. \quad (2)$$

$$\frac{1}{N} [T_N(x) - 1] T_{N-1}(x_j) + \frac{1}{2(N+1)} [T_{N+1}(x) + 1] (-1)^j \}$$

and

$$w_0(x) = w_N(x) = \quad (3)$$

$$\frac{1}{2N} \left\{ \left[\frac{T_{N-1}(x) + 1}{(N-1)} \right] + \left[\frac{T_{N+1}(x) + 1}{(N+1)} \right] + \left[\frac{T_N(x) - 1}{N} \right] \right\}.$$

The weights for definite integration take the much simpler form.

$$w_j(1) = \frac{4 \sum_{k=0}^{N/2} \frac{1}{1-4k^2} \cos \frac{2jk\pi}{N}}, \quad j=1(1)N-1 \quad (4)$$

with

$$w_0(1) = w_N(1) = \frac{1}{N^2-1}. \quad (5)$$

In the solution of Fredholm equations possessing Green's functions it becomes necessary to evaluate integrals of the form

$$I_2(x; f) = \int_x^1 f(y) dy. \quad (6)$$

By noting that the simple substitution $u=-y$, produces the equivalence

$$I_2(x; f) = \int_{-1}^{-x} f(u) du = \int_{-1}^{-x} F(y) dy = I_1(-x; f) \quad (7)$$

where $F(y) = f(-y)$, it can easily be verified that the weights $w_j^{(1)}$ associated with the integral I_1 and the weights $w_j^{(2)}$ used

in evaluating I_2 satisfy the following useful relationship

$$w_j^{(2)}(x) = w_{N-j}^{(1)}(-x), \quad j=0(1)N. \quad (8)$$

It is therefore not necessary to calculate a new set of weights. Since x will usually correspond to the quadrature points $x_i = -\cos i\pi/N$, ($i=0(1)N$) it is economical to store the $(N+1) \times N$ arrays of weights corresponding to these points thus eliminating much of the computational labour required in the original scheme [4]. The coefficients used in the error estimates do still need to be evaluated but this requires only a relatively small amount of work.

3.3.2 CHOICE OF INTERPOLATION FORMULAE

The Lagrange interpolation formulae (A3-2)-(A3-4) and the Fourier-Chebyshev series (A3-6)-(A3-9) will be used in programs. The Lagrange scheme is particularly useful if the data points are irregularly distributed within the range of definition, but it should be noted that the Lagrange and Chebyshev expansions are identical for the same set of data points. According to Elliott [32] the formula (A3-6) has a small theoretical advantage over (A3-8) when the Chebyshev series converges slowly, and sometimes if the interpoland is even or odd. It also has the practical advantage that the matching points for degree N are alternate points for the set using degree $2N$. Since the intermediate points corresponding to degree $2N$ are known it proves economical to adopt El-gendis' philosophy [63] of storing appropriate

coefficients for interpolation as well as for quadrature.

By combining (A3-6) and (A3-7) and noting that the points at which new function values must be found are the points $x_s = \cos(2s+1)\pi/N$, $s=0(1)(N-1)$ it is easily shown that

$$f(x_s) = 2 \sum_{j=0}^N f(y_j) \sum_{k=0}^N \frac{\cos jk\pi}{N} \cos \frac{(2s+1)k\pi}{2N}, \quad s=0(1)N-1 \quad (1)$$

where $y_j = \cos j\pi$, $(j=0(1)N)$. Equation (1) can be concisely written in the form

$$f(x_s) = \sum_{j=0}^N c_{s,j} f(y_j), \quad s=0(1)N-1 \quad (2)$$

where the coefficients $c_{s,j}$, $(j=0(1)N)$ take the appropriate form from (1). Use can be made of the identity

$$\sum_{j=0}^N \cos jA = \frac{1}{2} \sin NA \cot \frac{A}{2} \quad (3)$$

in order to evaluate the coefficients. Thus

$$\begin{aligned} c_{s,j} &= 2 \sum_{k=0}^N \frac{\cos jk\pi}{N} \cos \frac{(2s+1)k\pi}{2N}, \quad s=0(1)N-1, \\ &\quad j=0(1)N, \quad (4) \\ &= \left(\frac{-1}{2N}\right)^{j+1} \left[\cot \left\{ \frac{2(j+s)+1}{4N} \pi \right\} - \cot \left\{ \frac{2(j-s)-1}{4N} \pi \right\} \right]. \end{aligned}$$

The coefficients $c_{s,j}$ may be stored in order to make evaluation of (2) more economical. Identity (3) may also be used to reduce the computation involved in interpolating for f at a general point x . If $\theta = \cos^{-1}x$ then the formula (A3-6) is equivalent to

$$f(x) = \frac{\sin N\theta}{2N} \sum_{k=0}^N (-1)^k f(x_k) \left[\cot \frac{1}{2} (k\pi + \theta) - \cot \frac{1}{2} (k\pi - \theta) \right]. \quad (5)$$

Interpolation becomes necessary when insufficient values of the integrand have been evaluated in order to produce an accurate integral. Suppose for illustrative purposes that up until the r^{th} iteration N quadrature points have been sufficient to calculate accurate integrals, but at this stage the iterates become too complicated and a quadrature order $2N$ is deemed to be necessary. Four possibilities present themselves. i) Either one uses the integral equation with quadrature order N to find the intermediate values of the r^{th} iterate, or ii) one can evaluate the r^{th} iteration with quadrature order N and then interpolate the resulting values. iii) Alternatively one can interpolate the integrand containing the $(r-1)^{\text{th}}$ iterate and then reevaluate the integral or iv) one can interpolate the $(r-1)^{\text{th}}$ iterate and then calculate the corresponding intermediate integrand values prior to integration. The first two choices cannot bring about any increase in accuracy at the r^{th} iteration. This only occurs at the $(r+1)^{\text{th}}$ iteration. However they do not waste any of the integrals which have already been evaluated, as must be the case for the second two alternatives. As long as the values so produced remain within the region of accessibility of the solution, subsequent iterations will converge more accurately to the correct values. Of the two, the second choice is to be preferred since it is more economical. The third choice should never be used since it involves more work even than the first alternative and it also cannot produce increased

accuracy of the r^{th} iteration. To see this consider for example the Clenshaw-Curtis formulae (A2-2)-(A2-9) applied to the integral

$$I = \int_{-1}^1 f(y) dy. \quad (6)$$

Suppose that the set of integrand values $f(\cos i\pi/N), i=0(1)N$ has been augmented by the set of values $f(\cos(2s+1)\pi/2N), s=0(1)N-1$ using equations (2)-(4) of this section. Let the coefficients relating to orders N and $2N$ be $a_i^{[N]}, (i=0(1)N)$ and $a_s^{[2N]}, (s=0(1)2N)$, respectively. Similarly let the integrand values be $f_i^{[N]}$ and $f_s^{[2N]}$. It is clear that

$$f_i^{[N]} = f_s^{[2N]}, \quad i=0(1)N, \quad s=2i. \quad (7)$$

The remaining values of $f^{[2N]}$ are found from the interpolator. Thus

$$f_s^{[2N]} = \sum_{k=0}^N a_k^{[N]} \cos \frac{ks\pi}{2N}, \quad s=1, 3, \dots, N-1. \quad (8)$$

The coefficients $a_s^{[2N]}$ required in the quadrature of order $2N$ are calculated from the usual formula

$$a_s^{[2N]} = \frac{1}{N} \sum_{j=0}^{2N} f_j^{[2N]} \cos \frac{s j \pi}{2N}, \quad s=0(1)2N. \quad (9)$$

Their actual form may be expressed in terms of the $a_i^{[N]}, (i=0(1)N)$ as the following succession of equations shows. Equation (6) is equivalent to

$$a_s^{[2N]} = \frac{1}{N} \left[\sum_{j=0}^{2N} f_j^{[2N]} \cos \frac{sJ\pi}{2N} + \sum_{j=1}^{2N-1} f_j^{[2N]} \cos \frac{sJ\pi}{2N} \right] \quad (10)$$

$$= \frac{1}{N} \left[\sum_{j=0}^N f_j^{[N]} \cos \frac{sJ\pi}{N} + \sum_{j=1}^{2N-1} \left\{ \sum_{i=0}^N a_i^{[N]} \cos \frac{iJ\pi}{2N} \right\} \cos \frac{sJ\pi}{2N} \right] \quad (11)$$

$$= \frac{1}{N} \left[\sum_{j=0}^N f_j^{[N]} \cos \frac{sJ\pi}{N} + \right. \quad (12)$$

$$\left. \sum_{i=0}^N a_i^{[N]} \left\{ \sum_{j=0}^{2N} \cos \frac{sJ\pi}{2N} \cos \frac{iJ\pi}{2N} - \sum_{j=0}^N \cos \frac{iJ\pi}{N} \cos \frac{sJ\pi}{N} \right\} \right].$$

Let

$$S_{2N}(i, s) = \sum_{j=0}^{2N} \cos \frac{sJ\pi}{2N} \cos \frac{iJ\pi}{2N} \quad (13)$$

and

$$S_N(i, s) = \sum_{j=0}^N \cos \frac{iJ\pi}{N} \cos \frac{sJ\pi}{N}. \quad (14)$$

The orthogonality relationships for sums of cosines [4] indicate that

$$S_{2N}(i, s) = 0 \quad i \neq I, \quad (15)$$

$$= 2N \quad i=s=0 \text{ (or } 2N), \quad (16)$$

$$= N \quad i=s \neq 0 \text{ (or } 2N), \quad (17)$$

and

$$S_N(i, s) = 0 \quad i \neq s, \quad (18)$$

$$= N \quad i = s = 0 \text{ or } N, \quad (19)$$

$$= \frac{N}{2} \quad i = s \neq 0 \text{ or } N. \quad (20)$$

If $0 \leq s \leq N$, then substitution of equations (12)-(17) into (9) shows that

$$a_s^{[2N]} = a_s^{[N]} \quad 0 \leq s \leq N, \quad (21)$$

$$a_N^{[2N]} = \frac{a_N^{[N]}}{2}. \quad (22)$$

If $s > N$ then one may write $s = 2N - k$ where $0 < k \leq N$, so that equation (9) becomes

$$\begin{aligned} a_s^{[2N]} &= \frac{1}{N} \left[\sum_{j=0}^N f_j^{[N]} \cos \frac{j k \pi}{N} - \sum_{i=0}^N a_i^{[N]} S_N(i, k) \right] \\ &= 0. \end{aligned} \quad (23)$$

The resulting quadratures of orders N and $2N$ will therefore be identical. The choice therefore lies between options ii) and iv). Option ii) only has the effect of delaying the calculation of more accurate values until the next iteration and so it seems sensible to take choice iv) provided that less than half of the quadratures have been evaluated before an increase in order is deemed necessary. Otherwise a great deal of computation will have been wasted.

The policy to be adopted in automatic routines will be to attempt the initial calculations with a minimum number of grid points at which the solution will be updated. If necessary this value will be increased until all points can be evaluated to the desired accuracy. Any calculations which require fewer quadrature points can be carried out by selecting only intermediate points.

3.3.3 DISCRETIZATION STRATEGIES

This section describes the various configurations of points at which the functions involved in an integral equation will be evaluated within its domain of definition. Various schemes will be shown explicitly and their titles will be denoted by a D followed by a number. Some designs correspond to standard arrangements of points. The discretization D1 for example is of the same pattern as that used for the classical Nyström method [7,8], in which the solution is evaluated at the abscissae used in calculating the integral. The exact placing of the points depends on the quadrature rule but the general form is the same for all schemes. In many examples treated by successive approximations it will be seen that the use of Gauss-Seidel updating proves beneficial. The Gauss-Seidel implementation of D1 is given as an illustration. See scheme D1G. This modification of more complex schemes is cumbersome to display in algebraic terms and will therefore not be included in subsequent schemes. However, the computational implementation is just as easily carried out as in D1G. Wherever possible the configurations of points will be described in algebraic terms but later designs are intended for

automatic routines which become complicated to illustrate in this way. They will be given in the form of flow diagrams.

Since most methods of solution will be iterative, an iteration counter r will be placed on the approximations to the solution. $f^{[r]}$ denotes the r^{th} iterate of the solution f . In order to signify that values of the solution are approximations obtained from the use of a quadrature rule or introduction of an interpolator, subscripts will be used. In either case a summation of say $N+1$ terms will be involved. N will either signify the quadrature order or be equivalent to the degree of the interpolating polynomial. If the particular value of the solution is being or has been derived from a quadrature then N will be further subscripted by the letter Q and if from an interpolator then the subscript I will be used. A glance at schemes D1 and D3 makes the use of these subscripts clear. Thus in D1, N_Q signifies that a quadrature of order N is being used to calculate updated values of the solution f . In scheme D3 a quadrature of order $2N$ is used so that the subscript of the $r+1^{\text{th}}$ iterate is $2N_Q$. The integrand involves function values from the r^{th} iteration which are calculated using a quadrature of order N , hence the terms carrying the subscript N_Q . The intermediate values are obtained by interpolation on this set and therefore carry the subscript N_I . If the interval of definition is subdivided in order to find values of f then the notation $f_{\Sigma N}$ will be used and is meant to convey that a sum of quadratures is necessary to obtain the approximate value of f . Abscissae will be

denoted by x_i , ($i=0(1)N$). Points will be ordered from left to right so that for a closed quadrature rule used over the interval $[a,b]$, $x_0=a$ and $x_N=b$. Any remaining details will be elaborated as they occur.

D1

This is the most basic configuration that will be used in which the solution is evaluated at points corresponding to the abscissae of the quadrature rule. The approximate equation takes the form

$$f_{N_Q}^{[r+1]}(x_i) = g(x_i) + \sum_{j=0}^N w_j(x_i) K(x_i, x_j, f_{N_Q}^{[r]}(x_j)), \quad i=0(1)N. \quad (1)$$

where $\{w_j(x)\}$, ($j=0(1)N$) are the weights of the quadrature rule.

D1G

The Gauss-Seidel version of any iterated scheme involves substituting current approximations to the solution directly into the integrand as soon as they have been found and resembles the Gauss-Seidel method for the iterative solution of sets of algebraic equations [6]. The appropriate modification of the above scheme is

$$f_{N_Q}^{[r+1]}(x_i) = g(x_i) + \sum_{j=0}^{i-1} w_j(x_i) K(x_i, x_j, f_{N_Q}^{[r+1]}(x_j)) + \sum_{j=i}^N w_j(x_i) K(x_i, x_j, f_{N_Q}^{[r]}(x_j)), \quad i=0(1)N. \quad (2)$$

The following two schemes employ the progressive property of either the Clenshaw-Curtis or Patterson rules. They are used if the quadrature order needs to be increased.

D2

$$\begin{aligned} f_{N_Q}^{[r+1]}(x_s) &= g(x_s) + \sum_{j=0}^N w_j(x_s) K(x_s, x_j; f_{N_Q}^{[r]}(x_j)), \quad s=2i, \\ & \quad i=0(1)N, \\ f_{N_I}^{[r+1]}(x_t) &= \sum_{k=0}^N c_k(x_t) f_{N_Q}^{[r+1]}(x_k), \quad t=2i+1, \quad i=0(1)N. \end{aligned} \quad (3)$$

D3

$$\begin{aligned} f_{2N_Q}^{[r+1]}(x_s) &= g(x_s) + \sum_{j=0}^{2N} w_j(x_s) K(x_s, x_j; f_{N_Q}^{[r]}(x_j)) + \\ & \quad \sum_{k=1}^{2N-1} w_j(x_s) K(x_s, x_k; f_{N_I}^{[r]}(x_k)), \quad s=0(1)2N. \end{aligned} \quad (4)$$

where

$$f_{N_I}^{[r]}(x_k) = \sum_{h=0}^N c_h(x_k) f_{N_Q}^{[r]}(x_h). \quad (5)$$

Schemes D2 and D3 correspond to the choices ii) and iv) of section 3.3.2 respectively. In the above schemes the number of solution collocation points $N+1$ equals the number of quadrature points and the points are in fact the same. The schemes have been displayed in this way for notational convenience. The number of update grid points N_Q can be made independent of the quadrature order. Wherever possible it is advisable to employ solution values from updated points in the quadratures, since they will be more accurate than

interpolated values. However, this can require the manipulation of prohibitively large numbers of points if many quadrature abscissae become necessary and schemes such as D3 do then have their use (see the discussion in Example 2 section 2.7.1 on the reactor problem for large Peclet numbers).

Several schemes will require the use of interpolation in order to find function values at points which do not correspond to points of the update grid. The evaluation of particular singular integral equations using the schemes of Chapter 6 for example make this a necessity. Volterra equations for which the kernel $K(x,y;f(y))$ is undefined when $y > x$, or split kernel Fredholm equations involving similar properties can be treated by means of such schemes. A typical scheme for a Volterra equation is given in D4. The appropriate modifications may be made for singular and split kernel equations.

D4

$$f_{N_Q}^{[r+1]}(x_i) = g(x_i) + \sum_{j=0}^N w_j(x_i) K(x_i, x_j; f_{N_Q}^{[r]}(x_j)) \quad (6)$$

where $x_i, (i=0(1)N)$ lie in $[a,b]$ and $x_j, (j=0(1)N)$ lie in $[a, x_i]$ and

$$f_{N_Q}^{[r]}(x_j) = \sum_{k=0}^N c_k(x_j) f_{N_Q}^{[r]}(x_k), \quad j=0(1)N. \quad (7)$$

D5

In the step-by-step solution of Volterra equations most schemes seem to be based on the construction of combinations of various repeated rules such as the trapezoidal or Simpson's rules which are not of very high order. As an alternative it is proposed to construct a scheme which uses the definite integration form of Clenshaw-Curtis quadrature from order 1 up to some value N . The rules of order 1 and 2 are the trapezium and Simpsons rules respectively, but subsequently rules represent increasingly more powerful quadrature approximations. It is to be hoped that the solution values obtained at each step will provide a better interpolating polynomial with which to find values required at the next step since the mini-max properties of the Chebyshev polynomials may be used to advantage. In order to clarify ideas consider the interval $[a,b]$ and impose a maximum Chebyshev fit of $(N+1)$ points upon it. The abscissae are given by the usual linear transformation

$$x_i^{[N]} = ((b-a)\cos\frac{i\pi}{N} + a + b)/2, \quad i=0(1)N. \quad (8)$$

For the grid of order $N-1$ take the range $[a, x_1^{[N]}]$. The abscissae in this range are then given by

$$x_i^{[N]} = ((x_1^{[N]} - a)\cos\frac{i\pi}{(N-1)} + a + x_1^{[N]})/2, \quad i=0(1)N-1. \quad (9)$$

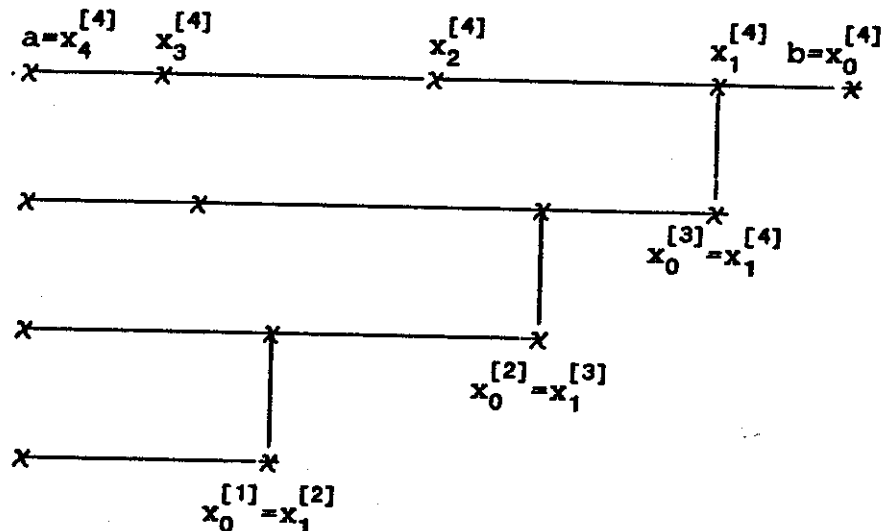
This process is carried on until the point $x_0^{[1]}$ has been obtained, the general form being given by

$$x_i^{[n-1]} = ((x_1^{[n]} - a) \cos \frac{i\pi}{(n-1)} + a + x_1^{[n]})/2, \quad n=N(-1)2, \quad (10)$$

$$i=0(1)n-1.$$

Figure 1 illustrates the process for $N=4$. Once the points

Fig.1



given by equation (10) have been found, evaluation of the solution at the points $x_0^{[n]}$, $n=1(1)N$ is carried out, beginning with $n=1$. For $n>2$ interpolation must be used to find the function values appropriate to that particular value of n . Since the values for order $(n-1)$ are situated at the Chebyshev points $x_i^{[n-1]} = \cos i\pi/(n-1)$, $i=0(1)(n-1)$, better approximations should normally be obtainable than if say an equidistant fit had been used. The scheme may be used equally well for iterative processes as for straightforward step-by-step methods.

As an introduction to the construction of adaptive schemes, the details for a method which relies on the summation of

integrals over preassigned subintervals will be given next. This scheme can be of use if it is known that either the solution or kernel are badly behaved in certain regions of the interval $[a,b]$. The equations for the slowing down of neutrons (3.3.4-2)-(3.3.4-14) (see Chapter 3) possess a discontinuous kernel for which this type of scheme is ideal. The scheme takes the following form.

D6

$$f_{\Sigma N}^{[r+1]}(x_{si}) = g(x_{si}) + \sum_{j=1}^{NI} \sum_{k=0}^{NI} w_{jk}(x_{si}) K(x_{si}, x_{jk}, f_{\Sigma N}^{[r]}(x_{jk})), \quad (11)$$

$$s=1(1)NI,$$

$$i=0(1)N_s,$$

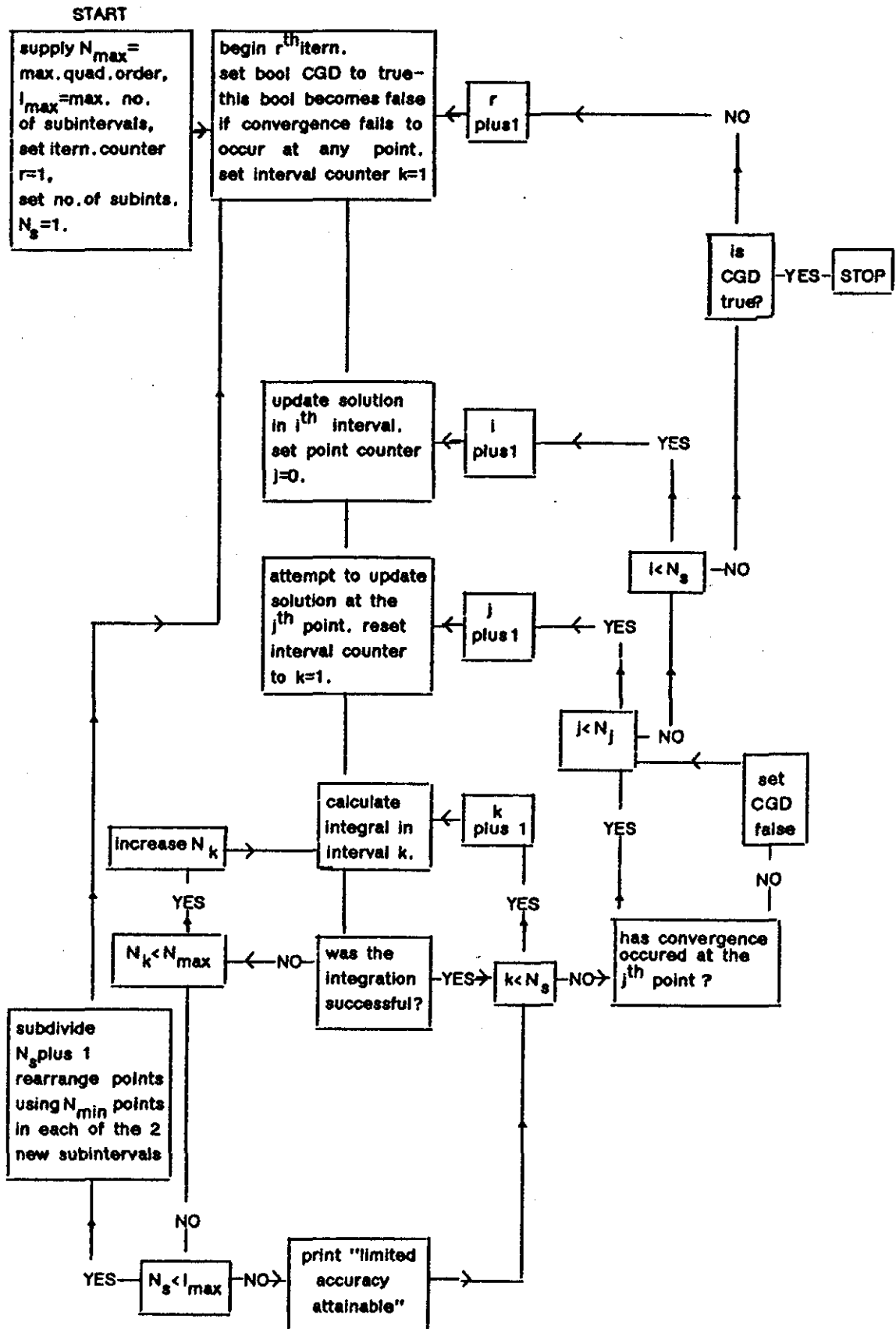
where NI is the number of subintervals. The first of the subscripts refers to the subinterval to which the weights or abscissae belong and the second subscript signifies which one it is. In an automatic routine, adaptivity is provided by manipulating the data points in an attempt to match the local behaviour of the integrand. In this type of scheme the relevant functions are evaluated proportionately more in regions where they fluctuate most. If the regions of greatest fluctuation remain stationary then it is possible to subdivide the interval around these regions once and for all and update the solution at suitable points within them. If these regions do not remain fixed as in the displacement kernel $K(x-y)$ then a different set of subintervals may be necessary for each point. Naturally this requirement will prove more expensive, but so long as the solution is updated

at sufficient points then accurate values will be obtainable. Flow diagrams for a many-interval update grid (one in which subintervals are kept once they are created) and single update grid (the update grid is kept constant but increase of quadrature points and subdivision can occur) are given in schemes D7 and D8 respectively. These schemes in effect use preceding schemes such as D3 and D6 but implement them automatically. The decision to invoke doubling up of points or subdivision can be triggered either by error tests within the integrator such as those used by Clenshaw and Curtis [4] or O'Hara and Smith [49] or by testing the derivatives of the iterates as they emerge [134]. Many embellishments to the schemes are possible, but they become increasingly more difficult to depict in concise form. In scheme D8 for example it can be advantageous to be able to increase the order of the update grid as the iteration proceeds. This modification is far more easily made to a computer program than the flow diagram. Further schemes will therefore not be given.

In summary, care has to be taken in order to keep the relative accuracies of quadrature and interpolation rules on a par, as obviously a low order interpolation formula would make a high order quadrature rule acceptable only when the unknown function $f(y)$ is simple, but the kernel $K(x,y)$ is poorly behaved.

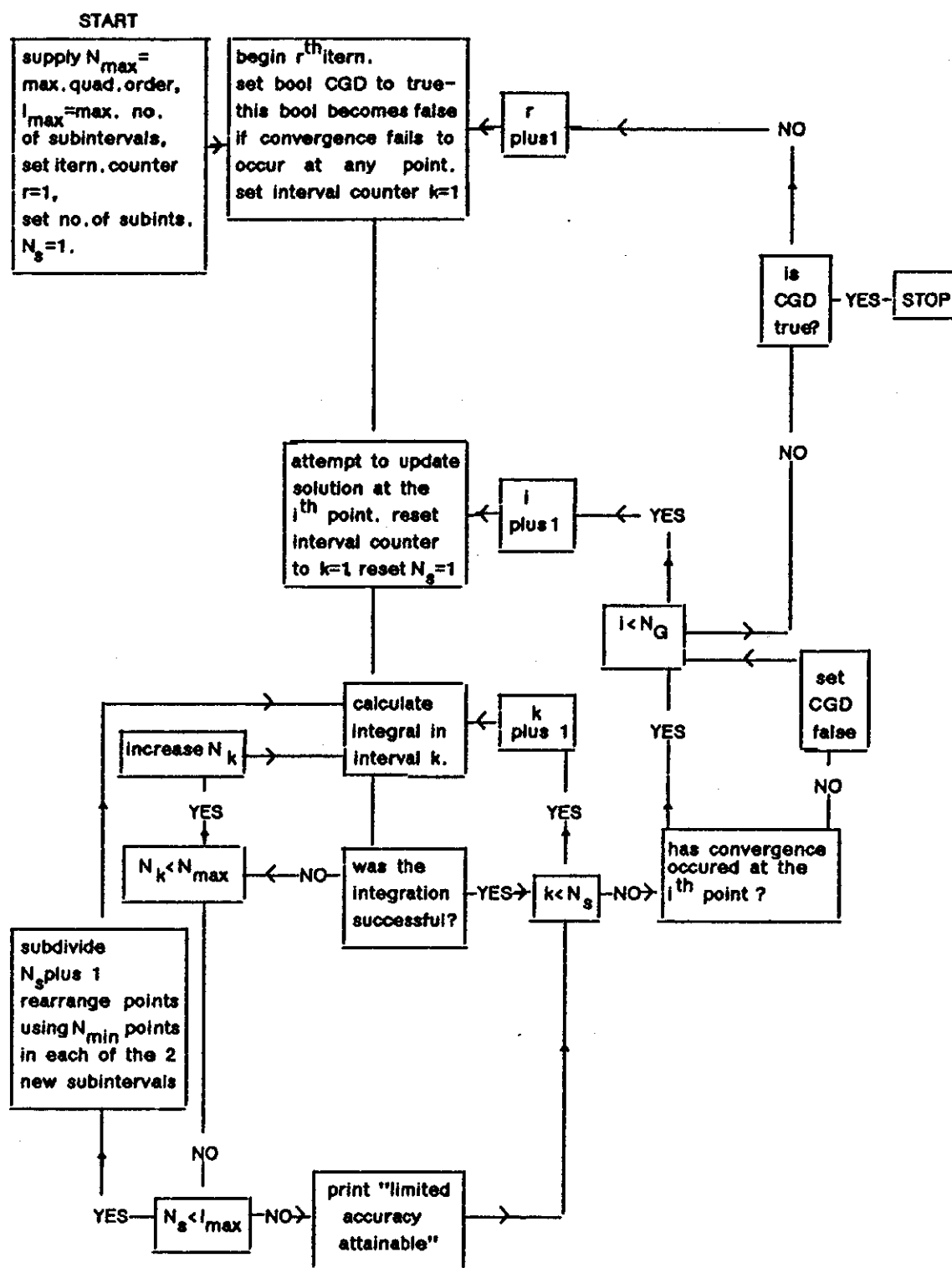
D7

Fig. 2



D8

Fig.3



The details contained in this chapter will now be put to use in creating some numerical schemes for the solution of integral equations.

CHAPTER 4

ALTERNATIVE WAYS OF IMPLEMENTING SUCCESSIVE APPROXIMATIONS

4.1 INTRODUCTION

The results of chapter 3 reveal that the method of successive approximations can be a very useful means of solving integral equations. The well known convergence conditions for both linear Volterra and Fredholm equations (see chapter 3) reveal that in theory at least, the method will provide the solution of Volterra equations for all values of the parameter λ , whereas this is not the case for Fredholm equations. This fact suggests the search for ways of regarding Fredholm as Volterra equations. Various Volterra and Fredholm integral equations may have the same solution. For example the equations

$$f(x) = x + \int_0^1 \min(x, y) f(y) dy \quad (1)$$

and

$$f(x) = x/\cos\sqrt{x} - \lambda \int_0^x (x-y) f(y) dy \quad (2)$$

both have the solution

$$f(x) = \sin(x\sqrt{\lambda})/[\sqrt{\lambda}\cos\sqrt{\lambda}], \quad (3)$$

Successive approximations applied to equation (2) will theoretically produce the solution to (2) and hence (1) for all values of λ , whereas application of the method to equation (1) will only succeed if $|\lambda| < \|K\|^{-1}$ where $K(x, y) = \min(x, y)$. It is also conceivable that a Fredholm

equation whose radius of convergence for successive approximations is R_1 , may be equivalent to another Fredholm equation whose radius of convergence is $R_2 > R_1$. Whether it is possible to associate different classes or types of equation with a particular solution is in general not clear. Any means of converting one to the other is also far from obvious. Some insight can be gleaned from the solution of boundary value problems by the shooting method [86,98]. This method can be regarded as an application of the methods for initial value problems. When converted to integral form, initial value problems become Volterra equations and boundary value problems generate Fredholm integral equations. It will be seen that translation of the shooting method to integral equations reduces the problem of solving Fredholm equations to that of solving equations of Volterra type.

4.2 THE SHOOTING METHOD FOR INTEGRAL EQUATIONS ARISING FROM BOUNDARY VALUE PROBLEMS

The advantages of converting differential to integral equations have already been stressed in chapter 2 and numerical evidence has been supplied in chapter 3. Solution of the resulting integral equations by successive approximations has proved to be successful except in the cases when the kernel norm was too large to allow convergence to occur. Since boundary value problems reduce to Fredholm equations this can present difficulties to the method. The following scheme avoids these short comings since it produces

Volterra instead of Fredholm equations from the original boundary value problem. As an illustration take the general equation

$$f_{xx}(x) = K(x; f, f_x), \quad f(a)=A, \quad f(b)=B, \quad (1)$$

$$f_x = d/dx, \quad f_{xx} = d^2/dx^2.$$

When converted to integral form it is the condition $f(b)=B$ which forces the equation to be of Fredholm type. The corresponding integral equation is

$$f(x) = A + \frac{(x-a)B}{(b-a)} + \int_a^b L(x,y)K(y; f(y), f_y(y))dy \quad (2)$$

where

$$\begin{aligned} L(x,y) &= \frac{x-y-(b-y)(x-a)}{(b-a)}, & a \leq y \leq x, \\ &= -\frac{(b-y)(x-a)}{(b-a)}, & x \leq y \leq b. \end{aligned} \quad (3)$$

In order to prevent this happening the condition at b is discarded and a guessed condition at a is introduced. The introduction of an initial guess $f_x(a)=\alpha$ and integration of (1) then produces

$$f_x(x) = \alpha + \int_a^x K(y; f(y), f_y(y))dy, \quad (4)$$

$$f(x) = A + \alpha(x-a) + \int_a^x (x-y)K(y; f(y), f_y(y))dy. \quad (5)$$

The solution of (3) and (4) will thus coincide with the original equation (1) if

$$R[\alpha] = A - B + \alpha(b-a) + \int_a^x (b-y)K(y; f(y, \alpha), f_y(y, \alpha)) dy \quad (6)$$

$$= 0.$$

Note that in equation (6) it is recognised that the solution f is a function of α , namely $f(y, \alpha)$. Assume that the particular value of α which forces $R[\alpha]$ to be identically equal to zero is $\alpha^* = \alpha + \delta\alpha$. Inserting α^* into (5) and expanding the integrand in (6) as a Taylor series in powers of $\delta\alpha$ one obtains

$$R[\alpha^*] = 0$$

$$= R[\alpha] + \delta\alpha(b-a) + \quad (7)$$

$$\int_a^x (b-y) [K_t(y; f, f_y) f_{\alpha} + K_{t_y}(y; f, f_y) f_{y\alpha}] \delta\alpha dy,$$

terms involving higher powers of $\delta\alpha$ than the first having been neglected. Since α does not depend on x , a Newton iteration for α may be set up in the usual way giving

$$\alpha_{r+1} = \alpha_r - R[\alpha_r]/R_{\alpha}[\alpha_r], \quad r=0, 1, \dots \quad (8)$$

where

$$R_{\alpha}[\alpha] = b-a + \int_a^x [K_t(y; f, f_y) f_{\alpha} + K_{t_y}(y; f, f_y) f_{y\alpha}] dy. \quad (9)$$

In order to proceed through the iteration (8), values of $f(x, \alpha)$, $f_x(x, \alpha)$, $f_\alpha(x, \alpha)$ and $f_{x\alpha}(x, \alpha)$ must be known. These may be found on differentiating equations (3) and (4) with respect to α . Thus

$$f_\alpha(x, \alpha) = x - a + \int_a^x (x - y) [K_f(y; f, f_y) f_\alpha + K_{fy}(y; f, f_y) f_{y\alpha}] dy \quad (10)$$

and

$$f_{x\alpha}(x, \alpha) = 1 + \int_a^x [K_f(y; f, f_y) f_\alpha + K_{fy}(y; f, f_y) f_{y\alpha}] dy \quad (11)$$

respectively. These equations may be regarded as a system of four Volterra equations which may be solved by successive approximations. The various ways of implementing the methods to be described in chapter 7 are applicable here. Alternatively, since values of f and f_α will be known over the complete range $[a, b]$ it is possible to iterate using only equations (5) and (10) and obtain the values for f_x and $f_{x\alpha}$ by means of a differentiator provided that $f_{x\alpha} = f_{\alpha x}$ in order to make this possible.

It is often possible to reduce the integro-differential equations (4), (5), (10) and (11) to pure integral equations by the methods described in Chapter 2 in which case only equations (5) and (10) in suitably modified form need be solved.

It should be emphasised that the method described above has strong connections with imbedding techniques (see [99, 100, 101, 102, 103, 107, 108, 109]). The basis of such methods

centres on the notion that the solution to a functional equation be regarded as a function of some parameter.

These ideas have received considerable attention over the last twenty years and are now well known in many areas such as variational problems, boundary-value problems, differential-difference equations and integral equations. In particular, they feature extensively in optimal filtering, radiative transfer and neutron transport. These methods characteristically lead to the formulation of systems of initial value problems which are generally easier to solve than the original problems. In contrast, the present method is used to create a system of Volterra integral equations which may be solved directly by means of successive approximations.

4.2.1 AN EXAMPLE SOLVED BY THE SHOOTING METHOD

Consider the split kernel equation (A0.3-14) possessing the equivalent differential form

$$f_{xx}(x) + \lambda f(x) = 0, \quad f(0) = 0, f_x(1) = 1. \quad (1)$$

The procedure outlined in section 4.2 gives rise to the iteration

$$\alpha^{[r+1]} = \alpha^{[r]} - R[\alpha^{[r]}] / R_\alpha[\alpha^{[r]}], \quad r = 0, 1, 2, \dots \quad (2)$$

where

$$R[\alpha^{[r]}] = \alpha^{[r]} - 1 - \lambda \int_0^1 f(y, \alpha^{[r]}) dy, \quad (3)$$

$$R_{\alpha}[\alpha^{[r]}] = 1 - \lambda \int_0^1 f_{\alpha}(y, \alpha^{[r]}) dy, \quad (4)$$

and the iterations for f and f_{α} are given by

$$f^{[s+1]}(x, \alpha^{[r]}) = \alpha^{[r]} x - \lambda \int_0^x (x-y) f^{[s]}(y, \alpha^{[r]}) dy \quad (5)$$

and

$$f_{\alpha}^{[s+1]}(x, \alpha^{[r]}) = 1 - \lambda \int_0^x (x-y) f_{\alpha}^{[s]}(y, \alpha^{[r]}) dy, \quad (6)$$

for each value of r .

The purpose of this section is both to illustrate the method and also emphasise that it can be used for values of λ greater than the spectral radius of the Fredholm kernel. Table 1 gives relevant information to the case $\lambda=10$ which is well outside the norm bounds. Results in Table 1 were obtained using a Clenshaw-Curtis indefinite integrator with $N=8$ and beginning the iteration with $\alpha^{[0]}=1$. Since successive approximations are used to iterate for f and its derivatives, nonlinear equations can also easily be solved. The nonlinear form of the reactor problem (A0.4-9a) was treated in the same manner.

Table 1.

r	S ₁	S ₂	E _α	E _f
0	3	4	2.0(-11)	7.0(-11)
1	4	4	1.5(-10)	8.1(-11)
2	5	4	3.2(-10)	1.4(-10)
3	4	5	2.4(-11)	3.5(-10)
4	5	5	3.6(-11)	2.8(-10)

r=iteration counter for α.

S₁, S₂=maximum number of iterations required to find f(x, α^[r]) and f_α(x, α^[r]) respectively.

E_α, E_f=maximum relative errors in α^[r] and f(x, α^[r]) respectively.

Note that the exact values are:-

α=1/cos√λ, f(x, α)=sin(x√λ)/[√λ cos√λ].

In that example with p=1 the iterates for α converged in 5 steps with the maximum number of iterations corresponding to S₁ and S₂ in table 1 being 12. The final relative error E_f was 5.3(-10) showing that once again the method can be used as a viable means of solving integral equations. The relevant equations corresponding to equations (3,4,5,6) in this instance are

$$R[\alpha^{[r]}] = \alpha^{[r]} + 2 \int_0^1 e^{-py} f^2(y, \alpha^{[r]}) dy, \quad (7)$$

$$R[\alpha^{[r]}] = 1 + 4 \int_0^1 e^{-py} f(y, \alpha^{[r]}) f_{\alpha}(y, \alpha^{[r]}) dy \quad (8)$$

together with

$$f(x, \alpha^{[r]}) = 1 + \alpha^{[r]} e^{py} + 2 \int_0^1 [-1 + e^{p(x-y)}] f^2(y, \alpha^{[r]}) dy \quad (9)$$

and

$$f_{\alpha}(x, \alpha^{[r]}) = e^{px} + 4 \int_0^1 [-1 + e^{p(x-y)}] f(y, \alpha^{[r]}) f(y, \alpha^{[r]}) dy. \quad (10)$$

4.3 USES OF THE SHOOTING METHOD IN CONNECTION WITH GENERAL FREDHOLM EQUATIONS

Section 4.2 described the use of the shooting method in solving boundary value problems by first converting them to Volterra rather than Fredholm equations. It would be very useful if this technique could be applied to general Fredholm equations. Unfortunately the present means of obtaining the Volterra from the Fredholm equation can only be applied to certain specific examples. Since the Fredholm equation has to be differentiated first it often proves impossible to remove all terms containing definite integrals. No attempt will therefore be made to extend the method beyond that which has been presented in the previous section.

4.4 EQUIVALENCE CLASSES OF INTEGRAL EQUATIONS

It has been pointed out in the introduction to this chapter that no obvious means appears to exist of converting Fredholm to Volterra equations with the same solution, or of associating different equations with different norms with the same solution.

Both operations would be desirable, especially in connection with the method of successive approximations. On the one hand successive approximations always converge for Volterra equations, theoretically at least. On the other hand it is

evident that the norm and solution of an integral equation are independent of each other in as much as equations possessing the same solution but different kernels and hence also norms do exist. It would be desirable that, given an integral equation for which the Neumann series does not converge, one could obtain a related equation with the same solution but a larger radius of convergence. A means of accomplishing this latter objective for linear Fredholm equations does exist and will be developed in the next section.

4.4.1 A METHOD FOR CREATING EQUIVALENCE CLASSES OF LINEAR FREDHOLM INTEGRAL EQUATIONS

The device to be described will produce equivalence classes in the sense that given the equation

$$f(x) = g(x) + \lambda \int_a^b K(x, y) f(y) dy \quad (1)$$

another equation

$$f(x) = G(x) + \lambda \int_a^b H(x, y) f(y) dy \quad (2)$$

possessing the same unique solution may be derived from it. Let

$$g^*(x) = \int_a^x g(t) dt, \quad (3)$$

$$K^*(x) = \int_a^x K(t, y) dt \quad (4)$$

and

$$({}_1)K(x, y) = \int_a^b K_z(x, z) K^*(z, y) dz \quad (5)$$

Equation (1) may be written as

$$f(x) = g(x) + \lambda \int_a^b K(x, y) \frac{d}{dy} \left[\int_a^y f(s) ds \right] dy \quad (6)$$

On integrating by parts in equation (6) one finds after some manipulation that

$$\begin{aligned} f(x) = & [g(x) - \lambda \int_a^b g^*(t) K(x, t) dt] + \\ & \lambda \int_a^b [K(x, b) - \lambda ({}_1)K(x, y)] f(y) dy. \end{aligned} \quad (7)$$

Putting

$$G^{[1]}(x) = g(x) - \lambda \int_a^b g^*(t) K(x, t) dt \quad (8)$$

and

$$H^{[1]}(x, y) = K(x, y) - \lambda ({}_1)K(x, y) \quad (9)$$

equation (7) can be more conveniently written in the form

$$f(x) = G^{[1]}(x) + \lambda \int_a^b H^{[1]}(x, y) f(y) dy. \quad (10)$$

It is easily seen that repeated use of the method leads to the set of equations

$$f(x) = G^{[i]}(x) + \lambda \int_a^b H^{[i]}(x, y) f(y) dy, \quad i=1, 2, 3, \dots \quad (11)$$

where

$$G^{[1]}(x) = G^{[i-1]}(x) - \lambda \int_a^b G^{[i-1]}(s) \cdot H^{[i-1]}(x, s) ds \quad (12)$$

and

$$H^{[1]}(x, y) = H^{[i-1]}(x, b) - \lambda {}_{(1)}H^{[i-1]}(x, y). \quad (13)$$

Note that it is convenient to set

$$G^{[0]}(x) = g(x) \quad (14)$$

and

$$H^{[0]}(x, y) = K(x, y) \quad (15)$$

in order to conform with the above notation. The convergence criteria for equations (11) with respect to λ may or may not differ from the conditions applying to the original equation. Equations (11) must satisfy

$$|\lambda| \|H^{[k]}\|_2 < 1, \quad k=0, 1, 2, \dots \quad (16)$$

For simplicity take the functions $H^{[k]}$ to be real valued but allow λ to be complex, $\lambda = \lambda_1 + i\lambda_2$. Inequality (16) is equivalent to

$$[\lambda_1^2 + \lambda_2^2]^{\frac{1}{2}} |L_1 - 2\lambda_1 L_2 + \{\lambda_1^2 + \lambda_2^2\} L_3| < 1 \quad (17)$$

where

$$L_1 = (b-a) \int_a^b H^{[k]}(x,b)^2 dx, \quad (18)$$

$$L_2 = \int_a^b \int_a^b H^{[k]}(x,b) {}_{(1)}H^{[k]}(x,y) dx dy \quad (19)$$

and

$$L_3 = \int_a^b \int_a^b {}_{(1)}H(x,y)^2 dx dy. \quad (20)$$

Inequality (17) defines a new λ -region within which successive approximations may be used. It may be modified to take the form

$$[\lambda_1^2 + \lambda_2^2] |L_3| + \left| \left(\lambda_1 - \frac{L_2}{L_3} \right)^2 + \lambda_2^2 - \left(\frac{L_2}{L_3} \right)^2 + \frac{L_1}{L_3} \right| < 1. \quad (21)$$

Condition (21) shows that convergence of the Neumann series depends on the two interrelated circles centred on (0,0) and $(L_2/L_3, 0)$. Condition (21) is difficult to analyse, but an example will show that it is possible to increase the radius of convergence. Consider the equation

$$f(x) = g(x) + \lambda \int_0^1 e^{(x-y)} f(y) dy. \quad (22)$$

Successive approximations will converge if

$$|\lambda| \|K\|_2 = |\lambda| \left[\int_0^1 \int_0^1 e^{2(x-y)} dx dy \right]^{\frac{1}{2}} < 1. \quad (23)$$

That is if

$$|\lambda| < 1/\sinh(1) = 0.851. \quad (24)$$

Applying the method one obtains

$$K^*(x, y) = [e^x - 1]e^{-y}, \quad (25)$$

$${}_{(1)}K(x, y) = -e^{-1}e^{x-y} \quad (26)$$

and hence

$$H^{[1]}(x, y) = e^{x-1}[1 + \lambda e^{-y}]. \quad (27)$$

The norm condition (23) becomes

$$\left[\frac{\lambda_1^2 + \lambda_2^2}{2} \right] \left[\frac{1 - e^{-2}}{2} \right] |1 + 2\lambda_1(1 - e^{-1}) + \frac{(\lambda_1^2 + \lambda_2^2)}{2}(1 - e^{-2})| < 1. \quad (28)$$

Taking λ to be real for simplicity, (28) can be written as

$$\lambda^2(\lambda + 0.527)(\lambda + 2.091) < 1. \quad (29)$$

Inspection of (29) reveals that it is approximately satisfied by values of λ within the range

$$-2.2 < \lambda < 0.9 \quad (30)$$

which is a considerable increase on the original bounds.

4.4.2 THE k^{th} TRANSFORMED KERNEL AS A SERIES IN λ

For completeness sake, the k^{th} member of the sequence of equations (4.4.1-11) is expressed explicitly as a series in λ . Let

$${}_{(k)}K(x, y) = \int_a^b {}_{(k-1)}K_z(x, z) {}_{(k-1)}K^*(z, y) dz, \quad k=1, 2, \dots \quad (1)$$

where

$${}_{(0)}K(x, y) = K(x, y) \quad (2)$$

and

$$J_{i_1, i_2}(x, b) = \int_a^b {}_{(i_1)}K_z(x, z) {}_{(i_2)}K^*(z, b) dz, \quad \begin{aligned} i_1 &= 1(1)k-1, \\ i_2 &= 0(1)i_1-1, \end{aligned} \quad (3)$$

$$J_{i_1, i_2, i_3}(x, b) = \int_a^b {}_{(i_1)}K_z(x, z) J_{i_2, i_3}^*(z, b) dz, \quad \begin{aligned} i_1 &= 2(1)k-1, \\ i_2 &= 1(1)i_1-1, \\ i_3 &= 0(1)i_2-1, \end{aligned} \quad (4)$$

$$J_{i_1, \dots, i_s}(x, b) = \int_a^b {}_{(i_1)}K_z(x, z) J_{i_2, \dots, i_s}^*(z, b) dz, \quad \begin{aligned} i_1 &= s-1(1)k-1, \\ i_2 &= s-2(1)i_1-1, \\ &\vdots \\ i_s &= 0(1)i_{s-1}-1. \end{aligned} \quad (5)$$

Then the k^{th} transformed kernel takes the form

$$H^{[k]}(x, y) = H^{[k-1]}(x, b) - \lambda {}_{(1)}H^{[k-1]}(x, y)$$

$$\begin{aligned}
&= K(x, b) - \sum_{i=0}^{k-1} \lambda^{2^{i-1}} {}_{(i)}K(x, b) + \\
&\quad \sum_{i_1=1}^{k-1} \sum_{i_2=0}^{i_1-1} (-\lambda)^{2^{i_1+2^{i_2-1}}} J_{i_1, i_2}(x, b) + \\
&\quad \sum_{i_1=2}^{k-1} \sum_{i_2=1}^{i_1-1} \sum_{i_3=0}^{i_2-1} (-\lambda)^{2^{i_1+2^{i_2+2^{i_3-1}}}} J_{i_1, i_2, i_3}(x, b) + \dots + \\
&\quad \lambda^{2^{k-1} + 2^{k-2} + \dots + 2^{-1}} J_{k-1, k-2, \dots, 0}(x, b) - \lambda^{2^k-1} {}_{(k)}K(x, y).
\end{aligned} \tag{6}$$

Abbreviating this complicated expression to

$$H^{[k]}(x, y) = \sum_{i=0}^{2^{k-2}} (-\lambda)^i J_i(x, b) - \lambda^{2^{k-1}} {}_{(k)}K(x, y) \tag{7}$$

where the $J_i(x, b)$ take the relevant forms from (3), the k^{th} transformed equation can be written as

$$\begin{aligned}
f(x) &= G^{[k]}(x) + \lambda \sum_{i=0}^{2^{k-2}} (-\lambda)^i J_i(x, b) \int_a^b f(y) dy - \\
&\quad \lambda^{2^k} \int_a^b {}_{(k)}K(x, y) f(y) dy.
\end{aligned} \tag{8}$$

It can be seen that the norm condition for this equation will involve a polynomial in λ of degree $2k+2$. This allows a maximum of k new λ regions in which successive approximations will work. There is no guarantee, however that this maximum number will be attained since the polynomial may have repeated roots and some regions may coincide with each other. In the special case when

$$\lim_{k \rightarrow \infty} \lambda^{2^k} \int_a^b {}_{(k)}K(x, y) f(y) dy = 0 \tag{9}$$

then a solution can be found in closed form. For all $n > N$ where N is such that

$$|\lambda|^{2^N} \int_a^b K(x, y) f(y) dy| < \epsilon \quad (10)$$

and ϵ is some positive value of sufficiently small magnitude, this solution is given by

$$f(x) = G^{[n]}(x) + \frac{\int_a^b G^{[n]}(x) dx}{\left[1 - \lambda \sum_{i=0}^{2^n-2} (-\lambda)^i \int_a^b J_i(x, b) dx\right]} \quad (11)$$

4.5 THE SOLUTION OF INTEGRAL EQUATIONS BY MEANS OF ACCELERATED NEUMANN SERIES

The Neumann series for the linear second kind Fredholm equation

$$f(x) = g(x) + \lambda \int_a^b K(x, y) f(y) dy \quad (1)$$

can, at least formally be written down no matter what the comparative size of the kernel norm and the parameter λ . It is well known, however (see Chapters 2 and 3 that this series is divergent if $|\lambda| > \|K\|_2^{-1}$. The peculiar nature of the problem is highlighted if one assumes that $f(x)$ in (1) is the exact solution to the equation and then substitutes r times for $f(y)$ using the right hand side of (1). One obtains

$$f(x) = g(x) + \lambda \sum_{j=0}^r \lambda^j \int_a^b K_j(x, y) g(y) dy + \lambda^{r+1} \int_a^b K_{r+1}(x, y) f(y) dy. \quad (2)$$

Equation (2) defines a particular equivalence class of equations

for equation (1) with kernel

$$K_{r+1}(x, y) \quad (3)$$

and free function

$$g(x) + \lambda \sum_{j=0}^r \lambda^j \int_a^b K_j(x, y) g(y) dy. \quad (4)$$

The series in (2) is precisely that which diverges in the use of successive approximations using $f_0(x) = g(x)$ if $|\lambda| > \|K\|_2^{-1}$ and yet, since the exact solution has been substituted at each step to produce (2), the last term on the right hand side must be large enough to cancel out the excess. Obviously, in practical terms it is not possible to compute r terms of the series and then remove the spurious information by cancellation with the last term in equation (2) since $f(y)$ is the unknown. However it will be seen that the Neumann series, even when divergent does contain the necessary information about the true solution and that this can often be extracted by means of an accelerator before numerical overflow occurs.

The procedure to be proposed will be based on the ϵ -algorithm ([96, 76, 95]). Both Shanks [94] and Wynne [76] suggest the use of accelerators on integral equations. They only present one example each however. A study of the use of the ϵ -algorithm/successive approximations approach on a selection of Volterra as well as Fredholm equations, including both the nonlinear and linear variety will be made in section 4.5.2.

4.5.1 IMPLEMENTATION OF THE METHOD

Combining the ϵ -algorithm with successive approximations can be achieved in numerous ways. Only one version will be tested here. Since it is of interest to see how the successive approximations progress in comparison to the accelerated values, the following scheme will be used. The successive approximations will be evaluated at the $N+1$ quadrature points as in Chapter 3 and values of successive iterates

$$f^{[r]}(x_i), \quad r=0,1,2,\dots, i=0(1)N \quad (1)$$

will all be stored. For each $r \geq 3$ the ϵ -algorithm will be applied to all $r+1$ iterates at the points $x_i, i=0(1)N$, and upon using the initial conditions

$$f_{-1}^{[r]} = 0, \quad (r=1,2,\dots), \quad f_0^{[r]} = f^{[r]}(x_i), \quad r=0,1,\dots \quad (2)$$

further quantities $f_s^{[r]}$ may be constructed by means of the relationship [76]

$$f_{s+1}^{[r]} = f_{s-1}^{[r+1]} + \{f_s^{[r+1]} - f_s^{[r]}\}^{-1} \quad r,s=0,1,\dots \quad (3)$$

In certain cases the sequences $f_{2s}^{[r]}$ ($s=0,1,\dots$) for fixed r , converge far more rapidly than the original sequence $f^{[r]}$, ($r=0,1,\dots$), [77,94]. Wynne [76] maintains that the sequences $f_{2s}^{[0]}$ and $f_{2s}^{[1]}$, ($s=0,1,\dots$) converge most rapidly and iteration therefore ceases if either

$$|f_m^{[0]} - f_{m-2}^{[2]}| < \epsilon |f_m^{[0]}| \quad (4)$$

if m is even, or

$$|f_{m-1}^{[1]} - f_{m-3}^{[3]}| < \epsilon |f_{m-1}^{[1]}| \quad (5)$$

if m is odd where ϵ is the accuracy.

4.5.2 RESULTS OF AN ϵ -ALGORITHM/NEUMANN SERIES SCHEME APPLIED TO SOME INTEGRAL EQUATIONS

Tables 1-4 contain comparisons of the maximum number of iterations required to attain seven-figure accuracy using ordinary successive approximations, Gauss-Seidel iteration and the ϵ -algorithm modification to both of these schemes.

Table 1
Results for linear Volterra equations

Equation Number	Successive Approxns.	Gauss-Seidel Itern.	ϵ -Algorithm	ϵ -Algorithm +Gauss-Seidel
A0.1-1	10	7	8	6
A0.1-2	6	3	5	3
A0.1-3	5	3	5	3
A0.1-4	5	3	5	3
A0.1-5	12	9	11	7
A0.1-6	10	7	8	6
A0.1-7	9	6	8	5

The table gives the maximum number of iterations to achieve 7 figure accuracy.

Inspection of Tables 1 and 2 reveals that the ϵ -algorithm invariably produces a considerable saving in the number of iterations for Volterra equations especially when coupled with the Gauss-Seidel scheme. These results contrast interestingly

Table 2
Results for nonlinear Volterra equations

Equation Number	Successive Approxns.	Gauss-Seidel Itern.	ϵ -Algorithm	ϵ -Algorithm + Gauss-Seidel
A0.2-1	5	4	5	4
A0.2-2	14	10	12	8
A0.2-3	15	11	13	8
A0.2-4	9	7	8	5
A0.2-6	6	4	6	4

The table gives the maximum number of iterations to achieve 7 figure accuracy.

with those in Table 3. Both Gauss-Seidel iteration and the ϵ -algorithm had a beneficial effect, but the combination of ϵ - and Gauss-Seidel iteration proves to be inferior in nearly all the cases considered. The final table on nonlinear Fredholm equations contains results which resemble those in Tables 1 and 2 more closely.

The reader should be aware that the implementation of the ϵ -algorithm does not guarantee convergence in all cases. For example, the method works for the linear Fredholm equation (A0.3-2) even when λ is well outside the norm bounds on the spectral radius. However the same equation with free term $(-2/\pi)\cos(x)$ cannot be solved by this method if $\lambda=2/\pi$, since the

denominator in equation (4.5.1-3) becomes zero for $x=\pi/2$.

Table 3.
Results for linear Fredholm equations

Equation Number	$\ K\ _2^{-1}$	Successive Approxns.	Gauss-Seidel Iteration	ϵ -Algorithm	ϵ -Algorithm + Gauss-Seidel
A0.3-1	3.000	8 —*	7 —*	4 12*	6 —*
A0.3-2	0.765	7 —*	5 —*	5 6*	5 25*
A0.3-13	$\sqrt{6}$	15	11	6	7
A0.3-15	9.487	7	6	5	5
A0.3-17	$2/\sqrt{3}\pi$	8 ^{&} —	5 ^{&} 10	6 ^{&} 8	5 ^{&} 9
A0.3-18	$2.7(-2)$	—	—	15	—
A0.3-20	1.225	—	20	8	7

The table gives the maximum number of iterations to achieve 7 figure accuracy .
Note that $\lambda=1/2$ in all examples, except those marked * for which $\lambda=10$ and & where $\lambda=0.1$.
— denotes that convergence did not occur.

Having developed various extensions of successive approximations in this chapter, an alternative method which does not suffer from the same convergence difficulties will be presented in the next chapter.

Table 4
Results for nonlinear Fredholm equations

Equation Number	Successive Approxns.	Gauss- Seidel Itern.	ϵ - Algorithm	ϵ -Algorithm + Gauss-Seidel
A0.4-1	11	10	7	6
A0.4-2	19	18	8	8
A0.4-3	11	10	8	8
A0.4-4	8	6	5	5

The table gives the maximum number of iterations to achieve 7 figure accuracy.

CHAPTER 5

SOME IMPLEMENTATIONS OF NEWTON'S METHOD

5.1 INTRODUCTION

The results of Chapter 3 reveal that the direct solution of second kind equations, especially Volterra integral equations by means of successive approximations and its variants is to be highly recommended. However, Fredholm equations are not always amenable to these devices due to the well-known convergence difficulties associated with that type of equation. The modification using the ϵ -algorithm works well for many equations, but, as was pointed out in the previous chapter it does not work in every case.

In the present chapter an attempt will be made to extend the iterative approach involving direct numerical integration to the solution of both Fredholm and Volterra equations. It will be seen that the method does not suffer from the same kind of convergence difficulties as the Neumann series and is also equally applicable to nonlinear as to linear equations

5.2 DEVELOPMENT OF THE METHOD

The method will be developed in terms of the general Urysohn integral equation of the type (1.2-1).

$$f(x) = g(x) + \int_C K(x, y; f(x), f(y)) dy. \quad (1)$$

The range of integration is depicted as a general contour C . This may be a complex contour a definite range of real integration

$[a,b]$ or even the indefinite form $[a,x]$. Although the method will initially be constructed in order to solve Fredholm integral equations there is no reason why it should not be applied to Volterra equations.

Equation (1) may be regarded as a nonlinear operator equation applied to the function $f(x)$ in the form

$$R[x;f] = f(x) - g(x) - \int_C K(x,y;f(x),f(y))dy. \quad (2)$$

Solving equation (1) is then equivalent to finding that particular function $f^*(x)$ which satisfies

$$R[x;f^*] = 0. \quad (3)$$

R represents the residual. The well known method of linearization (Rall [14], Saaty [66]) may be applied to equation (2) in the following way. If $f(x)$ is perturbed to $f(x) + \delta f(x)$ then

$$R[x;f + \delta f] = f(x) + \delta f(x) - g(x) - \int_C K(x,y;f(x) + \delta f(x), f(y) + \delta f(y))dy. \quad (4)$$

On expanding the kernel and neglecting second order terms in $\delta f(x)$ one obtains a form of Newton's method

$$R[x;f] = -\delta f(x) + \int_C [K_{f(x)} \delta f(x) + K_{f(y)} \delta f(y)]dy \quad (5)$$

where

$$K_{f(z)} = \frac{d}{df(z)} K(x, y; f(x), f(y)), \quad f(z) = f(x) \text{ or } f(y). \quad (6)$$

If K does not depend on $f(x)$ the obvious omission is made. Equation (5) forms the basis for an iterative scheme

$$R[x; f^{[r]}] = -\delta f^{[r]} + \int_C [K_{f(x)}^{[r]} \delta f^{[r]}(x) + K_{f(y)}^{[r]} \delta f^{[r]}(y)] dy, \quad (7)$$

$r=0, 1, \dots,$

which may be regarded as an integral equation for the perturbation $\delta f^{[r]}(x)$. This perturbation may then be used to update the solution via the equation

$$f^{[r+1]}(x) = f^{[r]}(x) + \delta f^{[r]}(x). \quad (8)$$

In the linear case, equation (5) is of course exact, whereas in the nonlinear case, the result is an approximate linearised equation for $\delta f(x)$. This suggests an attempt to solve (7) by algebraic means. A basis set approach has been adopted and the perturbation approximated by

$$\delta f^{[r]}(x) = \sum_{j=0}^M a_j^{[r]} \varphi_j(x), \quad r=0, 1, 2, \dots \quad (9)$$

where $\varphi_j(x), j=0(1)M$ are a set of linearly independent functions. This leads to the set of linear algebraic equations

$$R[x; f^{[r]}] = \sum_{j=0}^M a_j^{[r]} [-\varphi_j(x) + \psi_j^{[r]}] \quad r=0, 1, 2, \dots \quad (10)$$

where

$$\varphi_j^{[r]} = \int_C [K_{f(x)}^{[r]} \varphi_j(x) + K_{f(y)}^{[r]} \varphi_j(y)] dy. \quad (11)$$

Once the a_j , $j=0(1)M$ have been found, the perturbation $\delta f^{[r]}(x)$ may be calculated via (9) and hence $f^{[r+1]}(x)$ from equation (8). Equations (8,10,11) are familiar as standard equations arising from the application of Newton's method to an integral equation, ([11], p.689). Convergence criteria for Newton's method may be found in numerous works among them being Kantorovich and Krylov [67], Todd [68] p.513, Anselone [69] (in particular the article by Moore), Rall [14] and Baker [11]. In contrast, equation (8) provides the means of evaluating the iterates in an entirely new way. The point which is often stressed about the standard Newton iteration is the computational expense involved in evaluating the Jacobian of the system. For linear equations this may not be significant since it can be evaluated once and for all. It is possible to save work on nonlinear integral equations by using only the initial value of the Jacobian or at least only updating it when convergence of several steps to some set of function values occurs and then continuing this process until a fixed point of the iteration is found. If one resorts to this device however, the usual quadratic convergence of Newton's method is lost.

Equation (9) gives rise to a Jacobian which is of dimension $(M+1)$. Since M is independent of both the quadrature order N and the number of solution grid points $NGRID$, it will be seen that the dimension of the Jacobian can be significantly reduced, especially in the early stages of the iteration.

5.3 NUMERICAL IMPLEMENTATION OF THE METHOD

A choice of basis functions $\varphi_j(x)$, $j=0(1)M$ has yet to be made. Experiments have been carried out with the simple polynomial fit

$$\delta f(x) = \sum_{j=0}^M a_j x^j. \quad (1)$$

Results were favourable when only a low degree polynomial was required ($M \leq 4$), but higher values of M produced large alternating coefficients with the associated loss in accuracy of the iterates. See [65] for example. Baker [11], p.704 also encounters this problem in his formulation of Newton's method for the H-equation (A0.4-1) which has appeared in the previous chapter. See [70,69,71,14] for further information on this equation and its solution.

In order to remedy this defect the polynomial fit will be replaced by the Chebyshev series

$$\delta f(x) = \sum_{j=0}^M a_j T_j(x). \quad (2)$$

By letting M and $NGRID$ (the number of points less one at which the solution is actually updated) be multiples of 2 with M less than or equal to $NGRID$, approximation (2) fits naturally into the numerical integration scheme employed in the previous chapter. The residual may conveniently be evaluated at the points $x_j = \cos i[NGRID/M]\pi/NGRID$ which coincide with the points at which the iterates $f(x)$ are already known where [2]

denotes the greatest integer less than or equal to z . The decision to let M be less than or equal to $NGRID$ does not appear unreasonable, since if Newton's method converges then the residual will tend to zero and one would not expect its behaviour to be any more complicated than that of the solution itself. This hypothesis has been substantiated by the examples studied. The resulting numerical scheme involves integrals of the form

$$I_j(x) = \int_{-1}^1 K_{j(z)}(x, y; f(x), f(y)) T_j(y) dy, \quad j=0(1)M, \quad (3)$$

$z=x \text{ or } y.$

Elliott [30] and Elliott and Warne [72] suggest a scheme for linear equations in which the kernel K is also expanded as a Chebyshev series and the resulting integrals

$$c_{jk} = \int_{-1}^1 T_j(y) T_k(y) dy \quad (4)$$

evaluated analytically. This form is computationally expensive [63] and also restrictive in the sense that the kernel must be defined over the whole square. (See section 5.6). It is considered to be more economical and to be of more general application to treat the integral in (3) in exactly the same way as in Chapter 3 using El-gendi type quadrature, or alternatively of course the Patterson form. As a consequence of making M and $NGRID$ independent of one another only a minimal number of the equations (5.2-9,10,11) need to be evaluated at each iteration and the labour involved in calculating values of the Jacobian is greatly reduced.

Indeed, it has been observed in practice that very low degree fits for $\delta f(x)$ - linear or quadratic - are adequate in the early stages of the iteration when the initial values of $f(x)$ are far from the true solution and only rough estimates of the perturbation are required. As convergence is approached M can be increased in order to approximate $\delta f(x)$ to the full desired accuracy. Once the coefficients $a_j^{[r]}, j=0(1)M$ have been found equation (5.2-11) may be used as an interpolation formula to obtain

$$f^{[r+1]}(x) = f^{[r]}(x) + \sum_{j=0}^M a_j^{[r]} T_j(x) \quad (5)$$

for any value of x at which $f(x)$, is previously known. Alternatively $f(x)$ may be obtained at the points $x = \cos(i\pi/M)$, $i=0(1)M$ using (5) and the remaining values at points $x_i = \cos(I\pi/NGRID)$, $(I=0(1)NGRID, I \neq i[NGRID/M], i=0(1)M)$ be obtained by means of an interpolation formula such as (3.4.2-2).

5.4 SOME LINEAR EXAMPLES

A pilot study will be carried out on a selection of the linear equations which have already been solved by successive approximations and its modifications. Volterra, ordinary and split-kernel Fredholm equations will be treated. The main purpose in exhibiting this preliminary set of results will be to give an indication of the order of accuracy which can be obtained using a range of values for the degree of the fitting polynomial on the residual $\delta f(x)$. In all examples

given the grid order will be $N=16$ since this number was sufficient to calculate accurate quadratures in all examples except for the equations containing square root terms.

The maximum relative error over the complete range for fitting degrees $M=2,4,8,16$ will be given together with the maximum number of iterations. Theoretically only one iteration should be necessary to obtain the solution to the highest attainable accuracy for any particular value of M . However at least two iterations are necessary in order to check convergence of the results.

5.4.1 RESULTS FOR THE LINEAR EXAMPLES

It is evident from the table that a fit of $M=4$ provides reasonably good accuracy in most cases. Increasing this value to $M=8$ has the effect of producing full figure accuracy in several examples and in the remaining examples full accuracy can be obtained by pushing M up to 16 except for those equations containing badly behaved functions. In solving nonlinear equations, for which the iteration will usually take more than 2 steps the use of $M=2$ during the initial iterations and increasing the value (up to the maximum $M=NGRID$ if necessary) will certainly enable considerable computational expense to be saved.

Table 1.
Results for Newton's Method on linear second kind equations

M	2		4		8		16	
Eqn. No.	E	R	E	R	E	R	E	R
A0.1-1	1.3(-2)	2	3.9(-5)	2	3.2(-8)	2	2.8(-11)	2
A0.1-2	1.9(-1)	2	9.1(-4)	2	2.7(-6)	2	1.7(-10)	2
A0.1-3	9.4(-4)	2	2.4(-6)	2	4.4(-11)	2	1.0(-11)	2
A0.1-4	6.1(-2)	3	9.8(-4)	3	2.8(-6)	3	2.3(-9)	3
A0.1-5	8.2(-3)	3	2.1(-5)	3	1.5(-10)	3	1.2(-11)	3
A0.1-6	2.6(-2)	3	1.5(-2)	3	1.5(-2)	2	1.8(-2)	2
A0.1-7	5.4(-3)	2	6.8(-6)	2	9.0(-11)	2	1.1(-11)	2
A0.2-1	3.4(-1)	3	3.6(-1)	3	3.6(-1)	3	1.3(-11)	2
A0.2-2	3.4(-1)	3	3.6(-1)	3	3.6(-1)	3	1.5(-5)	2
A0.2-3	6.9(-3)	2	5.6(-5)	2	3.1(-10)	2	1.2(-11)	2
A0.3-1	7.2(-2))	2	4.2(-4)	2	9.9(-10)	2	1.6(-10)	2
A0.3-2	4.4(-1)	2	7.5(-4)	2	5.4(-9)	2	1.0(-11)	2
A0.3-3	2.9(-2)	2	5.5(-6)	2	3.5(-8)	2	3.1(-11)	2

M=Degree of fit for $df(x)$.

E=Relative error achieved.

R=Number of iterations required.

Note that the quadrature order $N=16$ for all examples.

5.5 NONLINEAR EQUATIONS

The numerical implementation of Newton's method described in the previous sections was originally created in order to solve a particular nonlinear problem. The method was developed in the context of the material balance equations for tubular reactors with axial diffusion [75,91] in an

attempt to overcome convergence difficulties of successive approximations when applied to the integral formulations (A0.4-9a,9b), (see chapters 2 and 4). These equations were obtained from the parent differential equation (3.3.2-9)) by use of the methods in chapter 2. The use of successive approximations is limited by conditions (2.6-2)-(2.6-6) (and (2.5-3) in the linear case) and should theoretically not work in some instances although Gauss-Seidel and accelerated forms of iteration do achieve a moderate amount of success. In contrast, the Newton method is effective in solving both forms of equation (A0.4-9a,9b) over a wide range of p values [65]. Initially experiments were carried out with a polynomial fit to the perturbation $\delta f(x)$ for the case $p=1$.

Table 2
Solution of equation (A0.4-9a) using a polynomial fit of order $m=4$.

r	$f(0)$	$f(0.5)$	$f(1.0)$
0	1.0	1.0	1.0
1	0.6983	0.5069	0.5663
2	0.6397	0.5086	0.4633
3	0.636792	0.503917	0.457588
4	0.6367843	0.5039042	0.4575891
*	0.636784102	0.503903768	0.457588686

Order of quadrature formula is 8.

$p=1$.

* denotes values from Table 11 of section 3.7.1.

The convergence towards the true solution is shown in Table 2 at the representative points $x=0, 0.5$ and 1.0 , the initial function being $f^{[0]}(x)=1$. It may be observed that the sequence $\{f^{[r]}(x)\}$ is converging rapidly towards the correct values for this size of

the parameter p where it is possible to fit $\delta f(x)$ with a low degree polynomial. Similar results were obtained for equation (A0.4-9b). For large values of p ($p > 10$), convergence is still rapid but the instability observed in the original differential equation approach manifests itself in the need for higher degree polynomials with which to fit $\delta f(x)$. Inevitably, large alternating coefficients appear with an associated loss of accuracy in the solution values $f(x)$. The introduction of the form (5.3-2) remedies this defect since the Chebyshev coefficients a_j decrease in size rapidly as j increases. Since the perturbation fit M , solution fit NGRID and quadrature order N are all independent of one another it is possible to obtain high accuracy results using relatively low values of M or equivalently a Jacobian of small dimensions. Table 3 shows the accuracy obtainable for the most difficult case tested, $p=100$, using a range of values for $M=2,4,8,16$ and NGRID=16. The number of iterations required are also included

Table 3
Relative errors in $f^{[r]}(x)$ values using a Chebyshev fit to the $\delta f^{[r]}(x)$ values.

Eq. (A0.4-9b)			Eq. (A0.4-9a)	
M	R	ϵ	R	ϵ
2	3	$1.0(-2)$	3	$1.0(-3)$
4	3	$1.0(-3)$	3	$1.0(-4)$
8	3	$1.0(-7)$	3	$1.0(-9)$
16	4	$1.0(-10)$	3	$1.0(-11)$

M =degree of Chebyshev fit to $\delta f^{[r]}(x)$.

R =Number of iterations for a given M .

ϵ =Relative error achieved in $f^{[r]}(x)$ values.

Maximum order of quadrature formula is 64.

$p=100$.

It will be observed that, as the order of the fitting polynomial is increased resulting in accurate perturbations to the solution being obtainable, rapid convergence is achieved even in the extreme case $p=100$. Here, where the numerical instabilities associated with the conventional differential approach are at their greatest, it is necessary to increase the order of the quadrature procedure to 64 in the case of the exponential kernel associated with equation (A0.4-9a). Nonetheless, when the numerical integrations associated with the iterative process are performed accurately, the convergence rate is extremely rapid. It appears therefore, that the numerical difficulties associated with the parent differential equation may be transferred to the numerical integration involved in the iterative solution of the resulting integral equations. The powerful automatic quadrature techniques which are currently available are then capable of resolving this difficulty and rapidly converging solutions are obtained.

The main advantage of the present iterative solution is its ability to tackle nonlinear integral equations where the conventional algebraic approaches are difficult. An illustration of these difficulties is provided by the results obtained in attempting to solve equation (5.1-1) by the standard discretization

$$f(x_i) = g(x_i) + \sum_{j=0}^N w_j K(x_i, x_j; f(x_j)), \quad i=0(1)N. \quad (1)$$

The solution of these $N+1$ nonlinear equations for $f(x_i)$ by say, a

conventional form of Newton's method is a considerable task, especially if a high order quadrature rule is necessary to perform the integration accurately. For low values of the parameter p , accurate results are obtainable for N values as low as 8, but, even at $p=10$, it is necessary to use values up to $N=32$ as evidenced by the following Table.

Table 4
Relative errors obtained in $f(x)$ values from equation (1)

N	p=1	p=10
4	1.0(-4)	—
8	1.0(-8)	1.0(-2)
16	1.0(-10)	1.0(-7)
32	1.0(-10)	1.0(-10)

The cases $p=50$ and $p=100$ proved to be quite intractable by this algebraic method, since the number of quadrature grid points required for an accurate solution was prohibitive.

Before going on to the next section, brief details of the solution to Chandrasekhar's equation (A0.4-1), [70,71] by the present method will be given. Comparison may be made with the results in section 2.6.1 Table 11. Table 5 of this section contains the results for $\lambda=1/2$. The number of iterations are now much fewer than in the use of successive approximations, a fact which becomes more and more important as λ is increased. The case $\lambda=1$ for example could not be solved by the Neumann series whereas 15 iterations were sufficient to obtain the solution using the Newton scheme.

Table 5
Results for the H-equation (A0.4-1) by Newton's Method

M	x_1	R	x_2	R	x_3	R	x_4	R
4	1.09318919	4	1.18767610	4	1.23678739	4	1.25120439	4
8	1.09325082	4	1.18773375	4	1.23684157	4	1.25125809	4
16	1.09325667	4	1.18773510	5	1.23684298	5	1.25125949	5
32	1.09325670	5	1.18773511	6	1.23684300	5	1.25125952	7
*	—		1.187735		—		1.251259	

M is the number of fitting points for the perturbation $\delta f(x)$.
 NGRID=32 and the quadrature order N_Q is allowed to increase automatically up to this value.
 R is the number of iterations to attain convergence.
 $x_i = \cos(i\pi/4)$, $i=1(1)4$
 * Results in bottom line are from [70].

5.6 A MODIFIED SCHEME FOR VOLTERRA EQUATIONS

The scheme which has been described in the previous sections of this chapter may be used for both Volterra and Fredholm equations provided that the kernel $K(x,y;u)$ is defined over the whole square $a \leq x, y \leq b$ (and all $u \in C[a,b]$ or $L_2[a,b]$ say). Many Volterra equations such as (A0.1-13,15), contain kernels which are defined only over $[a,x]$. A modified scheme is therefore desirable for such equations.

Assume as before that the range $[a,b]$ has been normalized to $[-1,1]$ and that the Chebyshev fit (5.3-2) is used to approximate the perturbation $\delta f(x)$. The resulting set of linear equations for the coefficients a_j for such a Volterra equation is

$$R[x_i; f] = \sum_{j=0}^M a_j [-T_j(x_i) + \int_{-1}^{x_i} K_j(x_i, y; f(y)) T_j(y) dy], \quad i=1(1)M. \quad (1)$$

Note that the fit of δf corresponds to a fit over the whole range $[a, b]$. The integral in (1) can nonetheless be evaluated by the usual definite integration form (3.4.1-4,5) over the range $[-1, x_i]$ if it is written in the form

$$I(x_i) = \frac{(x_i+1)}{2} \int_{-1}^1 K(x_i, y; f(y)) T_j(y) ds \quad (2)$$

where

$$y = ((x_i+1)s-1+x_i)/2, \quad i=1(1)M. \quad (3)$$

If the equation is singular then the methods in the next chapter for the calculation of singular integrals may easily be applied to equation 2. Values of $f(y)$ required in the quadratures can be stored for each $i=1(1)M$ and then updated via equation (5.3-5). Alternatively they need not be stored at all and can be obtained whenever necessary by interpolation formula (3.4.2-5) on the set of grid values $f(x_i)$, $i=0(1)NGRID$.

CHAPTER 6

THE SOLUTION OF SOME SINGULAR INTEGRAL EQUATIONS

6.1 INTRODUCTION

The importance and wide range of applications of singular integral equations is well known. Singular equations often arise from potential problems, for example in electrostatics, hydrodynamics and elasticity theory, (see [40,126] for some basic references). Currently much work is being carried out in this field and numerous methods now exist for the solution of particular types of singular equations, [135,137,138,139,142]. The methods do have their limitations however. Most are only applicable to linear equations and in some cases such as Delves [137] special knowledge of the nature of the singularities is necessary. The schemes to be developed in this chapter will not suffer from these shortcomings. By building a singular integrator [12] into the numerical schemes which have already been discussed in the previous chapters it is possible to create a more general direct method which is capable of treating both linear and nonlinear equations without the need to analyse the singularities in advance.

Singular integral equations have been defined in Chapter 2. In this chapter only finite range singular equations will be treated. Equations which are defined over infinite ranges may be converted to a finite range by means of transformations such as

$$x = t/(1+t)$$

(1)

and

$$x = \exp(-t) \quad (2)$$

which map $t \in [0, \infty)$ onto $x \in [0, 1]$ or

$$x = \tanh(t) \quad (3)$$

which maps $t \in (-\infty, \infty)$ onto $x \in [-1, 1]$. The methods to be proposed can be used not only to solve singular equations but those possessing badly behaved kernels (or driving terms). This includes kernels with peaks and those with singular derivatives.

6.2 DESCRIPTION OF AN ACCELERATOR TECHNIQUE

The method to be described is in the spirit of the work carried out by Chisholm et al.[13] in which sequences of quadrature approximations are generated and then subjected to acceleration routines. In that work the sequences were created by a systematic increase in the number of quadrature points. In contrast, the present technique utilises the process described by Rabinowitz as "creeping up on the singularity". It involves integration over increasingly small subintervals as the singularity is approached. The resulting sequence of approximations is then accelerated by means of the ϵ -algorithm [76,77,94,95], (See also brief details on its use in section 4.5.1). Motivation for the method may be found in [12].

For convenience assume that $[0,1]$ is the range of integration and that the integral

$$S(x) = \int_0^1 I(x,y) dy \quad (1)$$

involves an integrand $I(x,y)$ which is singular at $y=0$. If the integrand is singular at $y=1$ suitable modifications to the method which follows can be made, or alternatively the integral can be submitted to the program in the form

$$S(x) = -\int_1^0 I(x,y) dy. \quad (2)$$

If the singularity occurs at $y=x$ (a common form is $k|x-y|$ which arises in potential theory) then the integral must be treated as one containing a split kernel and will therefore involve two integrals. They may both be treated as lower limit singular integrals by writing

$$S(x) = \int_x^1 I(x,y) dy - \int_x^0 I(x,y) dy. \quad (3)$$

Doubly singular integrals may be conveniently evaluated as one integral with singular lower end point from the equation

$$S(x) = \frac{1}{2} \int_0^1 [I(x, \frac{y}{2}) + I(x, 1-\frac{y}{2})] dy \quad (4)$$

although experience has shown that it is better to treat each of the integrals separately. All the other forms of singular integral such as those singular at x and one end point or singular at other points within the range may be dealt with by combinations of equations (1), (2) and (3). Returning now

to equation (1), this will be evaluated by constructing the sequence of integrals

$$S_n(x) = \int_{x_n}^1 I(x,y) dy \quad (5)$$

where $x_0=1$ and $x_n \rightarrow 0$ as $n \rightarrow \infty$. The successive terms of the sequence $\{S_n\}$ will be generated by the sums

$$S_n(x) = \sum_{k=0}^n J_k(x) \quad (6)$$

where

$$J_k(x) = \int_{x_j}^{x_{j-1}} I(x,y) dy, \quad j \geq 1. \quad (7)$$

Clearly

$$\lim_{n \rightarrow \infty} S_n(x) = S(x) \quad (8)$$

In the present method, the sequence $\{S_n\}$ will be subjected to the ϵ -algorithm, (see Chapter 4 or [12,76]) in order to obtain its limit and hence the integral $S(x)$. Various possibilities for the monotonic decreasing sequence $\{x_n\}$ exist. It was found [12] that the choice

$$x_n = \theta^n, \quad 0 < \theta < 1 \quad (9)$$

was most effective in practice. Large values of θ sometimes resulted in large numbers of terms ($n=20$) being necessary for the ϵ -algorithm to produce convergence to a specified

accuracy. Small values of θ occasionally induced x_n to approach the singularity too closely before convergence had occurred although rapid convergence of the accelerated sequences did occur in many cases. The value of $\theta=0.2$ was found to be a suitable compromise and will be used in solving integral equations. The numerical scheme is similar to discretization D8 in which the update grid is fixed but the points at which the integrand is evaluated are allowed to vary automatically. Subdivision occurs and the relevant points correspond to those in (9). A Clenshaw-Curtis or Patterson fit is then imposed upon each of the subintervals in order to calculate the integrals. This form of singular integral equation solver will be denoted by S1. The interpolation formulae (3.4.2-2) and (3.4.2-5) are used in obtaining the solution values required in the integrand from the basic update grid.

6.3 RESULTS FOR SOME EQUATIONS

The results which are presented in the following tables were obtained using a Chebyshev based update grid and Patterson quadrature. At the time of writing most results were only available from the Honeywell computer, working to 8 figures of accuracy. A realistic comparison with other work is therefore not possible, but the results do serve to indicate how the scheme functions. Equations (A0.1-13,15) are solved by the above scheme S1 and details are contained in Table 1. Due to the simple nature of the solutions to these equations it is possible to use an update grid of only three points. A maximum allowable

quadrature order, N_Q , for each integral over the subintervals $[x_{i-1}, x_i]$ is given together with the maximum number of quadrature points, N_P , used in evaluating the complete singular integral. The number of accelerator terms T is also supplied. The underlying method used in obtaining these results was the ϵ -algorithm + Gauss-Seidel form of successive approximations.

Table 1
Results obtained by the use of ϵ + Gauss-Seidel iteration and the scheme S1

Equation Number	N_Q	N_P	T	R	E
(A0.1-13)	3	15	5	5	2.4(-4)
	7	42	6	6	8.9(-8)
(A0.1-15)	3	18	6	7	4.4(-4)
	7	63	9	7	3.7(-8)

N_Q = max. no. of points allowed in any subinterval.

N_P = max. no. of points used over the whole range of integration $[a, x_i]$.

T = max. no. of terms used in the acceleration process.

R = min. no. of iterations required to achieve convergence.

E = relative error attained.

NGRID=2 for a Chebyshev based update grid.

Table 2 contains similar information concerning the two equations (A0.1-16,17) which may be found in [135] in which the fast Galerkin Algorithm of Delves et al. [137] is used. Comparison is not straight forward since the configuration and numbers of grid points are not the same. Nevertheless the table does show that reasonably few points are adequate to obtain accurate values to within the limits of the computer used. Due to the more complicated nature of the solutions it is found to be necessary to set NGRID equal to 8.

Table 2
Results obtained by the use of ϵ + Gauss-Seidel iteration and the scheme S1

Equation Number	N_Q	N_P	T	R	E
	3	15	5	7	9.5(-4)
(A0.1-16)	7	42	6	9	2.2(-8)
	15	105	7	10	1.1(-8)
	3	24	8	11	1.2(-1)
(A0.1-17)	7	77	11	15	2.9(-5)
	15	187	13	16	7.8(-6)

N_Q = max. no. of points allowed in any subinterval.

N_P = max. no. of points used over the whole range of integration $[a, x_1]$.

T = max. no. of terms used in the acceleration process.

R = min. no. of iterations required to achieve convergence.

E = relative error attained.

NGRID=8 for a Chebyshev based update grid.

By way of variation, results for equation (A0.3-9) given in Table 3 were obtained by the use of Newton's method. Successive approximations did not converge although the Gauss-Seidel iteration and ϵ -algorithm enhancement thereof did produce the solution. Again accurate results are obtained for a low expenditure in work since the update grid only requires three points, although a comparatively large number of quadrature points is necessary to attain full figure accuracy.

The main advantage of the S1 scheme over other methods is seen as its adaptability. It can as easily be implemented in a successive approximations setting as in the Newton method and can consequently be applied to nonlinear as well as linear

singular integral equations. The need to interpolate the solution can pose problems if the solution is in some sense singular.

Table 3
Results obtained by Newton's method and the scheme S1

Equation Number	N_Q	N_P	T	R	E
	3	21	7	2	4.4(-5)
(A0.3-9)	7	70	10	2	1.4(-7)
	15	150	10	2	6.0(-8)

N_Q = max. no. of points allowed in any subinterval.

N_P = max. no. of points used over the whole range of integration $[a, x_1]$.

T = max. no. of terms used in the acceleration process.

R = min. no. of iterations required to achieve convergence.

E = relative error attained.

NGRID=2 for a Chebyshev based update grid.

M = 2 for the fitting degree of the perturbation δf .

As an alternative it is possible to employ a scheme which is similar to D7 in that the subdivisions are kept and the solution updated at all quadrature points within each subdivision. Naturally, this implementation relies on the quadratures being of relatively low order. Use of the progressive Patterson or Clenshaw-Curtis rules greatly aids in this respect. In some instances a smoothing transformation can help in reducing the severity of a singularity (Morrow [134]). Equation (A0.3-6) for example which possesses a singular derivative of both the solution and the kernel can be transformed to

$$f(t^2) = t - 2t \int_0^1 s f(s^2) ds \quad (10)$$

by means of the transformation $x=t^2$. (More generally one may set

$x=t^n, n>2$). Investigations are currently being carried out on both of these alternatives

CHAPTER 7

APPLICATION TO SYSTEMS OF EQUATIONS

7.1 INTRODUCTION

In practice, one is often faced with systems of integral equations. Frequently they will have arisen from the conversion of systems of differential equations. Such systems occur in a wide field of applications. These include electric circuit theory, chemical engineering, control theory, population growth [56] and potential theory [18]. Systems of integral equations can also occur in the conversion of a single differential equation to integral form (see equations (3.2.4-12,13) for example). The shooting method developed in chapter 4 also results in systems of integral equations as can the conversion of the m^{th} order initial value problem.

$$f^{(m)}(x) = K(x; f, f', \dots, f^{(m-1)}), \quad f^{(r)}(a) = \eta_r, \quad (1)$$

$$r=0(1)m-1.$$

By defining the variables ${}^s f(x)$, $s=1(1)m$ as

$${}^1 f(x) = f^{(0)}(x) = f(x), \quad (2)$$

$${}^s f(x) = \frac{d}{dx} ({}^{s-1} f(x)) = f^{(s-1)}(x), \quad s=2(1)m, \quad (3)$$

the following system of integral equations may be created

$${}^s f(x) = \eta_{s-1} + \int_a^{x^{s+1}} f(y) dy, \quad s=1(1)m-1, \quad (4)$$

$${}^m f(x) = \eta_{m-1} + \int_a^x K(y; {}^1 f, {}^2 f, \dots, {}^m f) dy. \quad (5)$$

The Van der Pol equation [144, 56]

$$f^{(2)}(x) = -af(x) + \epsilon(1-f^2(x))f'(x), \quad f(0)=2, \quad (6)$$

$$f'(0)=0$$

for example is equivalent to the system of integral equations

$$^1f(x) = 2 + \int_0^x {}^2f(y)dy, \quad (7)$$

$$^2f(x) = \int_0^x [\epsilon(1-({}^1f(y))^2) {}^2f(y) - a {}^1f(y)]dy. \quad (8)$$

It is of course possible to carry out the reverse process and produce a single integral equation from a system [15,19], but use of this approach will not be considered.

In concluding this work it will be of interest to investigate the extension of the methods described so far to their use in connection with systems of equations. In initiating a description of the schemes to be used it will be assumed that the system of m integral equations

$${}^rf(x) = {}^rg(x) + \int_C {}^rK(x,y; {}^1f, {}^2f, \dots, {}^mf)dy, \quad r=1(1)m. \quad (9)$$

have already been obtained. Equation (9) may more concisely be written in vector form as

$$\underline{f}(x) = \underline{g}(x) + \int_C \underline{K}(x,y;\underline{f})dy \quad (10)$$

which is exactly the notation for a single integral equation with the obvious modifications for vectors.

7.2 SUCCESSIVE APPROXIMATIONS

Successive approximations for a system of equations forms a natural extension of the single equation case. Existence, uniqueness and convergence theory (in particular that for successive approximations) generally only requires trivial modifications to the details relating to single equations and involves the use of norm conditions for vectors and matrices rather than the scalar norms employed on single equations [56, 119]. The method of successive approximations has frequently been used in solving systems of integral equations [112, 113, 115, 116, 117, 118].

Successive approximations can be implemented in several ways. Firstly, normal iteration may be carried out in the sense that evaluation of ${}^sf^{[r+1]}(x_i)$, ($i=0(1)N, s=1(1)m, r=0, 1, 2, \dots$) only uses values from the r^{th} iteration. Secondly, Gauss-Seidel iteration may be employed in a number of ways. They are:- (i) G.S.1, all the most recent updates are used in evaluating the integrals, (ii) G.S.2, a complete set of solution values for ${}^sf(x_i)$, ($i=0(1)N, s$ and r held constant) is found and then substituted into the integrals for evaluation of ${}^{s+1}f(x_i)$. Alternatively one can update each solution at the same point before moving to the next. Then either, (iii) G.S.3, most recent values are used to evaluate the integrals or, (iv) G.S.4, the values ${}^sf(x_i)^{[r+1]}$, (i held constant, $s=1(1)m, r$ held constant) are all found before being used to find similar values at x_{i+1} . Finally, the Neumann series may be accelerated in the manner described in Chapter 4 with the

aid of the ϵ -algorithm [76]. Alternatively the Gauss-Seidel schemes may be accelerated.

7.2.1 COMPARISON OF FOUR ITERATIVE SCHEMES

In this section only the choices of ordinary direct iteration, Gauss-Seidel iteration G.S.1, the ϵ -algorithm modification and a combination of G.S.1 and the ϵ -algorithm will be studied. These choices outlined above are compared in the solution of the systems contained in appendices (A0.6)-(A0.9). A quadrature order $N=8$ is used in all cases.

Table 1
The number of iterations required to obtain convergence
in various implementations of successive approximations.

Problem Number	Ordinary Iteration	G.S.1	ϵ -Algorithm	G.S.1 + ϵ -Algorithm
A0.6-P1	14	8	14	8
A0.6-P2	—	—	16	15
A0.6-P3	8	7	8	7
A0.6-P4	12	14	10	9
A0.7-P1	8	7	5	4
A0.7-P2	28	15	12	11

G.S.1 stands for Gauss-Seidel modification of successive approximations.

— implies that convergence has not occurred.

The table displays the same characteristics for systems as were found for single equations and it is apparent that Gauss-Seidel + ϵ -algorithm iteration once again should form the preferred scheme.

7.2.2 SOLUTION OF (A0.9-P1)

This system is related to the differential equation

$$f''(x) + 2f(x)f'(x) = 0, \quad f(0)=0, f(12)=1. \quad (1)$$

and hence also to the single integral equations (A0.2-8), (A0.3-21) and (A0.4-11). Table 2 demonstrates the effectiveness of using successive approximations on the system of equations and the worth of possessing different methods of decomposing differential equations also becomes apparent. The ability to avoid the need to evaluate derivatives explicitly in calculations is one of the major advantages of the systems approach.

Table 2
Equation (1) solved as a system of integral equations

		x_0		x_1		x_2	
Problem Number		R	E	R	E	R	E
(A0.9-P1)	f_1	2	1.2(-11)	4	4.5(-10)	6	7.4(-10)
	f_2	2	3.9(-11)	4	6.5(-11)	7	8.8(-10)

NGRID = 8: - this is the update grid.

$N_Q = 8$: - this is the quadrature order used.

$x_i = 0, 6, 12, i=0, 1, 2$ respectively.

A final brief investigation will now be made into the viability of the Newton method schemes in connection with systems of equations.

7.3 NEWTON'S METHOD FOR SYSTEMS OF EQUATIONS

Newton's method as described in chapter 5 can be extended to the solution of systems of equations. It involves construction of the m residual equations

$${}^rR[x; \underline{f}] = {}^r f(x) - {}^r g(x) - \int_C {}^r K(x, y; \underline{f}(y)) dy, \quad r=1(1)m. \quad (1)$$

On introducing the m perturbations ${}^r \delta f(x)$, $r=1(1)m$ so that the exact solutions ${}^r f^*(x)$, take the forms ${}^r f^*(x) = {}^r f(x) + {}^r \delta f(x)$, $r=1(1)m$, similar arguments to those used in chapter 5 produce the equations

$$\begin{aligned} {}^r R[x; \underline{f}] = & -{}^r \delta f(x) + \int_C {}^r K_{r_f}(x, y; \underline{f}(y)) {}^r \delta f(y) dy + \\ & \sum_{\substack{s=1 \\ s \neq r}}^m \int_C {}^r K_{s_f}(x, y; \underline{f}(y)) {}^s \delta f(y) dy, \quad r=1(1)m. \end{aligned} \quad (2)$$

Use will again be made of low degree Chebyshev fits to the functions ${}^r \delta f(x)$ so that

$${}^r \delta f(x) = \sum_{j=0}^{M_r} a_j^{[r]} T_j(x), \quad r=1(1)m. \quad (3)$$

In order to accommodate this form, the usual linear transformation (2.1-3) will be used to transform the range of integration to $[-1, 1]$. The $M_r, r=1(1)m$ will be set equal to multiples of 2 and satisfy $M_r < N$ where $N+1$ is the number of update points. Substitution of equations (3) into (2) gives rise to

$$\begin{aligned}
 {}^rR[x;f] = & \sum_{j=0}^{M_r[r]} a_j^{[r]} \{ -T_j(x) + \int_{-1}^1 {}^rK_{r_f}(x,y;f(y)) T_j(y) dy \} + \\
 & \sum_{\substack{s=1 \\ s \neq r}}^m \sum_{j=0}^{M_s[s]} a_j^{[s]} \int_{-1}^1 {}^rK_{s_f}(x,y;f(y)) T_j(y) dy, \\
 & r=1(1)m.
 \end{aligned} \tag{4}$$

The $\sum_{r=1}^m M_r + m$ unknowns $a_j^{[r]}, j=0(1)M_r, r=1(1)m$ may be found by evaluating (4) at the Chebyshev points $x_i = \cos(i\pi/M_r), i=0(1)M_r, r=1(1)m$. The successive iterates to the solution are then updated in a manner analagous to that described in chapter 5.

7.3.1 RESULTS FOR SOME EXAMPLES

Table 3 contains results for a selection of the examples to be found in appendices A0.6 to A0.9. Absolute errors attained using the values $M_r = M = 4, 8, 16, r=1(1)m$, are depicted and show how few points are neccessary to attain very accurate results, $M=8$ giving full accuracy in most cases. The selection of examples is limited, and further work is needed to establish the full merits of the scheme.

7.4 APPLICATION OF THE SCHEMES TO STIFF SYSTEMS

Many systems of differential equations display the property of stiffness (Lambert [86]). The example (A0.6-P5) which was derived from equations (A0.6-28,29) occurs in Lambert and will be studied here. When translated to a system of integral equations it is clear that the problem of stiffness in the original differential equation is equivalent to the

Table 3
Newton's method applied to some systems of integral equations

Problem Number	x_0		x_1	x_2
	R	E	E	E
<u>M=4, $\epsilon=1.0(-4)$</u>				
(A0.6-P1)	2	2.9(-11)	4.2(-4)	2.1(-4)
(A0.6-P2)	3	2.4(-13)	7.7(-5)	2.7(-4)
(A0.6-P3)	4	5.6(-3)	1.0(-2)	5.6(-3)
(A0.6-P4)	2	7.3(-12)	2.6(-5)	1.6(-5)
(A0.9-P1)	3	3.9(-11)	6.5(-4)	8.8(-3)
<u>M=8, $\epsilon=1.0(-7)$</u>				
(A0.6-P1)	4	1.5(-11)	3.8(-9)	2.0(-10)
(A0.6-P2)	3	4.9(-13)	1.1(-5)	7.3(-7)
(A0.6-P3)	3	7.5(-11)	1.3(-7)	6.9(-7)
(A0.6-P4)	2	7.3(-12)	1.6(-10)	1.1(-11)
(A0.9-P1)	4	2.9(-11)	5.5(-4)	3.8(-5)
<u>M=16, $\epsilon=1.0(-9)$</u>				
(A0.6-P1)	3	0.0	0.0	7.3(-12)
(A0.6-P2)	3	5.9(-13)	1.1(-11)	6.8(-13)
(A0.6-P3)	3	8.3(-11)	1.5(-11)	7.3(-12)
(A0.6-P4)	2	1.1(-11)	1.8(-12)	3.6(-12)
(A0.9-P1)	4	2.1(-11)	3.7(-10)	9.8(-10)

R = max. number of iterations required to achieve convergence.

E = greatest absolute error attained in any of the solution values.

$x_1, i=0(1)2$ are the end and mid points of the respective ranges.

requirement that rapidly fluctuating functions in part of the region of definition need to be integrated. Because of this problem Lambert calculates that if a Runge-Kutta method were to be used on the differential system then a step-size of $h=0.00125$ would be necessary. In contrast it was found that by subdividing the range of integration using $h=0.1$ and employing quadratures of order $N=8$ it was possible to obtain full figure accuracy over the complete range. Even though the integration difficulties arising from the solutions are confined to the range $[0,0.1]$ this represents a large saving in the number of update points from 80 to 8 in this first interval. It was in fact possible to achieve full accuracy over the rest of the range $[0.1,1]$ using $N=8$ also. However, better comparative results would be obtained by a method such as Gear's [151] which is directly applicable to stiff systems of ordinary differential equations.

Possibilities exist for both successive approximations and Newton's method which can be implemented using several of the discretizations appearing in chapter 2. Either the automatic version of D7 or the hand tuned option which was used in the above example seem to be favourable alternatives. It is hoped that future work will involve the incorporation of a singular integrator into the scheme and it is envisaged that this might well provide the ideal tool to solve this type of problem.

CHAPTER 8

CONCLUSIONS

CONCLUSIONS

The work in this thesis must now come to an end. It is to be hoped that the numerical schemes which have been created so far for the solution of integral equations will serve a useful purpose not only because they are successful in solving many such equations as they stand, but also because they can form the basis of much more sophisticated schemes in the future. The discretizations which have been presented in Chapter 2 pave the way for much more complex numerical schemes which will be able to solve equations more efficiently and accurately. The schemes are largely independent of the particular numerical devices used in them and although they have mainly been based on integrators and interpolators using Chebyshev polynomials any other forms can easily be employed.

The use of successive approximations as described in Chapters 3 and 4 has proved to be very powerful, especially in connection with nonlinear equations. It is well known that convergence difficulties do arise, more so for Fredholm equations than for equations of the Volterra type but the contents of Chapter 4 reveal that there are ways of modifying the method to overcome these difficulties. The creation of equivalence classes of integral equations by means of integration by parts and the application of accelerators have achieved moderate success. The adaptation of the shooting method from a differential to integral equation setting allows certain Fredholm equations to be treated as Volterra equations thus overcoming some of the inherent convergence difficulties. Other methods

applicable to the solution of Volterra equations could also be easily substituted.

In situations where successive approximations fails, new routines based on Newton's method have been shown to be capable of delivering solutions in a more efficient manner than conventional implementations of that method. These schemes have also been used to treat nonlinear equations very successfully. First-kind equations can be solved by this method although very little research has been carried out on this possibility so far.

Many integral equations are singular and there has long been a need for methods for solving such equations. A new singular integral package has been created which performs very well. The great advantage of this tool is that it is independent of the method used to solve the integral equation. It can therefore easily be inserted into schemes based on successive approximations, Newton's method or any other routine for that matter.

The numerical implementations of successive approximations and Newton's method have also been extended to deal with systems of integral equations and this completes the armoury of devices for the solution of problems which can be reduced to integral equation form.

Since so many physical problems can be reduced to integral equations of one form or another, the value of the techniques

described in this thesis is at once obvious. One often finds a physical problem described in differential equation form. Chapter 2 contains many useful methods of transforming differential equations to their integral counterparts. The techniques, especially those derived from variation of parameters can be used to obtain numerous different but related integral equations from just one differential equation. This can be valuable in checking accuracy of results especially for nonlinear equations for which analytic solutions are rarely available. Sometimes an integro-differential equation will arise from the differential equation in which case it is often preferable to produce a system of integral equations. The value of extending the methods to cope with systems of equations is therefore assured.

As well as seeing the developement of more sophisticated schemes in the future it is hoped to be able to produce rigorous error bounds on the solution values obtained from any particular method. At present error analysis has been confined to the study of values obtained from the integrators. This aspect clearly needs to be extended and studied much more deeply if the schemes are to be of use in solving physical problems for which error bounds are often necessary.

APPENDICES

APPENDIX O

EXAMPLES SOLVED

0.1 LINEAR VOLTERRA EQUATIONS OF THE SECOND KIND

$$f(x) = 1 + \lambda \int_0^x f(y) dy, \quad 0 \leq x \leq 2, \quad (1)$$

$$= e^{\lambda x} \quad ([1] \text{ p.155, [144], [16] pp.22,25}).$$

The following six equations are contained in Baker [11] p.785.

$$f(x) = \cos x + \int_0^x (y-x) \cos(x-y) f(y) dy, \quad 0 \leq x \leq 2, \quad (2)$$

$$= (2\cos\sqrt{3}x + 1)/3.$$

$$f(x) = x + \int_0^x \sin(x-y) f(y) dy, \quad 0 \leq x \leq 2, \quad (3)$$

$$= x + \frac{x^3}{6}.$$

$$f(x) = \sin x + \int_0^x (x-y) f(y) dy, \quad 0 \leq x \leq 2, \quad (4)$$

$$= (\sin x + \sinh x)/2.$$

$$f(x) = e^x + 2 \int_0^x \cos(x-y) f(y) dy, \quad 0 \leq x \leq 2, \quad (5)$$

$$= e^x (1+x)^2.$$

$$f(x) = \cos x + \int_0^x f(y) dy, \quad 0 \leq x \leq 2, \quad (6)$$

$$= (e^x + \cos x + \sin x)/2.$$

$$f(x) = \sinh x - \int_0^x \cosh(x-y) f(y) dy, \quad 0 \leq x \leq 2, \quad (7)$$

$$= \frac{2 \sinh(\frac{\sqrt{5}x}{2})}{\sqrt{5}} e^{-x/2}.$$

$$f(x) = e^x + \int_0^x (x-y) f(y) dy, \quad 0 \leq x \leq 1, \quad (8)$$

$$= \left(\frac{3}{4} + \frac{x}{2}\right) e^x + \frac{e^{-x}}{4}.$$

$$f(x) = e^x + \lambda \int_0^x (x-y) f(y) dy, \quad 0 \leq x \leq 1, \quad (9)$$

$$= [\sqrt{\lambda} e^{\sqrt{\lambda}x} + (\lambda - \sqrt{\lambda}) e^{-\sqrt{\lambda}x} + e^x] / (\lambda - 1), \quad \lambda \neq 1,$$

$$f(x) = 1 + \int_0^x \frac{y}{x} (y-x) f(y) dy, \quad 0 \leq x \leq 10, \quad (10)$$

$$= \sin x / x.$$

$$f(x) = 13e^{-x} - 12 + 12 \int_0^x f(y) dy \quad (11)$$

$$= e^{-x}, \quad [11], \text{ p. xxx.}$$

$$f(x) = 13e^x - 12 - 12 \int_0^x f(y) dy \quad (12)$$

$$= e^x, \quad [11], \text{ p. xxx.}$$

$$f(x) = \frac{15}{4}x - x^{5/2} + \int_0^x (x-y)^{1/2} f(y) dy, \quad 0 \leq x \leq 1.5, \quad (13)$$

$$= \frac{15}{4} x, \quad [11], \text{ p. 876.}$$

$$f(x) = \frac{1}{2} \int_0^x \frac{f(y)}{y} dy. \quad (14)$$

Equation (14) is a Volterra form of (A0.3-5).

$$f(x) = x + x (3/4 - \ln(x)/2) + \int_0^x \ln(x-y) f(y) dy \quad (15)$$

$$= x.$$

$$f(x) = e^x + (x+1)^{3/2} e^{-1} + \int_{-1}^x (x-y-3/2)(x-y)^{1/2} f(y) dy \quad (16)$$

$$= e^x, \quad -1 \leq x \leq 1, \quad [135].$$

$$f(x) = e^{-2e(x+1)\ln(x+1)} +$$

$$2 \int_{-1}^x \ln(x-y) (x-2y+2xy^2-2y^3) f(y) dy \quad (17)$$

$$= e^{x^2}, \quad -1 \leq x \leq 1, \quad [135].$$

$$f(x) = ((x+1)^{1/2} + 1 + (x+1)\ln(x+1))/e +$$

$$\int_{-1}^x [(x-y-1)\ln(x-y) + (x-y-0.5)(x-y)^{1/2} f(y)] dy \quad (18)$$

$$= e^x, \quad -1 \leq x \leq 1, \quad [135].$$

0.2 NONLINEAR VOLTERRA EQUATIONS OF THE SECOND KIND

The following five equations are taken from Baker [11].

$$f(x) = e^{x^2} - x + x e^{(x^2)/2} - \int_0^x x y \sqrt{f(y)} dy \quad (1)$$

$$= e^{x^2}, \quad 0 \leq x \leq 4, \quad (p.832).$$

$$f(x) = 2 - e^x + \int_0^x e^{x-y} \{f(y)\}^2 dy \quad (2)$$

$$= 1, \quad 0 \leq x \leq 1, \quad (p.832).$$

$$f(x) = 1 - x + \int_0^x \{x \exp(y[x-2y]) + \exp(-2y^2)\} \{f(y)\}^2 dy \quad (3)$$

$$= \exp(x^2), \quad 0 \leq x \leq 1, \text{ (p.848).}$$

$$f(x) = \frac{1}{2} e^{-x} (1 + e^{-2x}) + \int_0^x e^{-x} \{f(y)\}^2 dy \quad (4)$$

$$= e^{-x}, \quad 0 \leq x \leq 16, \text{ (p.882).}$$

$$f(x) = 1 + x + \frac{x^2}{2} + \int_0^x (1+x-y) \{f(y)\}^2 dy \quad (5)$$

$$= 1, \quad 0 \leq x \leq 16, \text{ (p.882).}$$

$$f(x) = 1 + \int_0^x y(y-x) \{f(y)\}^\nu dy \quad \nu > 1, \quad 0 \leq x \leq 1. \quad (6)$$

Equation (6) is a nonlinear form for the Lane-Emden equation of index ν .

$$f(x) = \int_0^x \frac{K[f(y)] dy}{[\pi(x-y)]^{1/2}} \quad ([18], \text{ p.223}), \quad (7)$$

where K may be chosen to be any suitable function.

$$f(x) = 1 - e^{-x} + \int_0^x e^{y-x} [f(y) - f^2(y)] dy. \quad (8)$$

This is a Volterra equation of the boundary value problem

$$f''(x) + 2f(x)f'(x) = 0, \quad f(0)=0, \quad f(\infty)=1.$$

0.3 LINEAR FREDHOLM EQUATIONS OF THE SECOND KIND

$$f(x) = x + \lambda x \int_0^1 y f(y) dy \quad (1)$$

$$= \frac{3x}{3-\lambda},$$

$$f_r(x) = x \sum_{j=0}^r (\lambda/3)^j,$$

$$\|K\|_2^{-1} = 3.$$

$$f(x) = 1 + \lambda \int_0^{\pi/2} \cos(x-y) f(y) dy \quad (2)$$

$$= 1 + \frac{4\lambda}{[4-\lambda(\pi+2)]} (\cos x + \sin x),$$

$$f_r(x) = (\cos x + \sin x) \sum_{j=0}^r \lambda^j \left[\frac{2+\pi}{4} \right]^j, \quad \|K\|_2^{-1} = 0.765.$$

$$f(x) = 2 + \lambda \int_0^1 (x-y)^2 f(y) dy \quad (3)$$

$$= \alpha_0 + \alpha_1 x + \alpha_2 x^2,$$

$$\|K\|_2^{-1} = 3.873,$$

where

$$\alpha_0 = 60(6+\lambda)/(180-30\lambda-\lambda^2),$$

$$\alpha_1 = 360\lambda/(\lambda^2+30\lambda-180),$$

$$\alpha_2 = -\alpha_1.$$

$$f_r(x) = \sum_{j=0}^r \lambda^j f^{(j)}(x)$$

$$a_{j+1} = \frac{a_j + b_j + c_j}{3},$$

$$b_{j+1} = -(\frac{a_j}{2} + \frac{2}{3} b_j + c_j),$$

$$c_{j+1} = \frac{a_j + b_j + c_j}{5} - \frac{b_j}{4} - \frac{c_j}{3}$$

with

$$a_0 = b_0 = 0, c_0 = 2.$$

$$f(x) = 1 - \frac{(e-1)}{x} + \int_0^1 e^{xy} f(y) dy \quad (4)$$

$$= 1, \quad ([11], p.370),$$

$$\|K\|_2^{-1} = \left[\sum_{r=1}^{\infty} \frac{2^{r-1}}{r \cdot r!} \right]^{-1/2} \approx 0.768.$$

$$f(x) = \sqrt{x} - \int_0^1 \sqrt{xy} f(y) dy \quad (5)$$

$$= \frac{2}{3} \sqrt{x}, \quad \|K\|_2^{-1} \approx 2.$$

$$f(x) = \sqrt{x} - \sqrt{x} \int_0^1 f(y) dy \quad (6)$$

$$= \frac{3}{5} \sqrt{x}, \quad \|K\|_2^{-1} = \sqrt{2}.$$

$$f(x) = x^{1/2} \left(1 - \frac{\pi^2}{24} x\right) + \int_0^{\pi/2} (xy)^{3/2} f(y) dy \quad (7)$$

$$= x^{1/2}, \quad [136].$$

$$f(x) = (1-x^2)^{3/4} - \pi \sqrt{2} (2-x^2)/4 + \frac{2}{3} \int_{-1}^1 |x-y|^{-1/2} f(y) dy \quad (8)$$

$$= (1-x^2)^{3/4}, \quad ([135], [11] \text{ p.541}).$$

$$f(x) = g(x) + \int_{-1}^1 \ln|x-y| f(y) dy \quad (9)$$

$$= x, \quad ([11] \text{ p.536}),$$

where

$$g(x) = x - 0.5 \{x^2 \ln x + (1-x^2) \ln(1-x) - x - 0.5\}$$

$$f(x) = 1 + \int_0^1 \ln|x-y| f(y) dy, \quad [97]. \quad (10)$$

$$f(x) = g(x) + \frac{1}{\pi} \int_{-1}^1 \frac{d}{d^2 + (x-y)^2} f(y) dy, \quad \|K\|_2^{-1} = 0.672. \quad (11)$$

Equation (11) is Love's equation - see [11] pp. 358, 365, 372, 376, 381, 404, [63] p.391. $g(x)=1$ and $d=1$ in the text.

$$f(x) = 2 \tan^{-1} \left\{ \frac{k^{-1} \sin x}{[1 - \cos(x)][k_3 \cos(x) - k^{-2}]} \right\} + \quad (12)$$

$$\frac{k k_1}{\pi} \int_0^{\pi} \left\{ \frac{f(y)}{[1 - k_2 \cos(x+y)]} - \frac{f(\pi - y)}{[1 + k_2 \cos(x+y)]} \right\} dy$$

where

$$k_1 = (1 + k^2)^{-1}, \quad k_2 = k_1(k^2 - 1), \quad k_3 = 1 - k^{-2}.$$

This is the Lichtenstein-Gershgorin equation [76].

$$f(x) = 1 + x e^{-\int_0^1 K(x,y) f(y) dy}, \quad ([11] \text{ p.382}), \quad (13)$$

$$K(x,y) = \min(x,y), \quad \|K\|_2^{-1} = \sqrt{6}.$$

$$f(x) = x + \lambda \int_0^1 K(x,y) f(y) dy \quad [46], \quad (14)$$

$$= \sin(x\sqrt{\lambda}) / [\sqrt{\lambda} \cos \sqrt{\lambda}],$$

$$K(x,y) = \min(x,y), \quad \|K\|_2^{-1} = \sqrt{6}.$$

$$f(x) = x \sin(1) + \int_0^1 K(x,y) f(y) dy \quad (15)$$

$$= \sin x, \quad ([11] \text{ pp.357,402,415}),$$

$$K(x,y) = y(1-x) \quad y < x,$$

$$= x(1-y) \quad x \leq y.$$

$$f(x) = x^2 - \int_0^1 K(x,y)f(y)dy, \quad \|K\|_2^{-1} = \sqrt{90}, \quad (16)$$

$$= [(3e-2)e^x + e(2e-3)e^{-x}]/[e^2-1]-2$$

$$K(x,y) = y(1-x) \quad y < x,$$

$$= x(1-y) \quad x \leq y, \quad [xxx].$$

$$f(x) = e^x + \lambda \int_0^\pi K(x,y)f(y)dy, \quad \|K\|_2^{-1} = \frac{2}{\sqrt{3\pi}}. \quad (17)$$

$$= [\lambda \cos x(\lambda+1)]^{1/2} +$$

$$\frac{\lambda}{\Lambda \cos \pi \Lambda} \{e^{\pi} + \Lambda \sin \pi \Lambda\} \sin x \Lambda + 2e^x / (2+\lambda)$$

where

$$\Lambda = (\lambda+1)^{1/2}.$$

$$K(x,y) = \cos x \sin y \quad y < x,$$

$$= \sin x \cos y \quad x \leq y.$$

$$f(x) = \frac{\sin x}{2} + 2\cos \frac{x}{2} - 4(x+\pi) - 2 + \quad (18)$$

$$\int_0^x (1+x+y)f(y)dy + \int_x^{2\pi} (x+y)f(y)dy$$

$$= \frac{\sin x}{2}, \quad [11].$$

$$f(x) = \frac{5e^{2x}}{4} - \frac{1}{2}xe^2 - \frac{1}{4} + \lambda \int_0^x ye^y f(y)dy + \lambda \int_x^1 xe^y f(y)dy \quad (19)$$

$$= e^x, \quad ([11], \text{ p.385}).$$

$$f(x) = e^x + \frac{1}{2} \int_{-1}^x (x-y)f(y)dy + \frac{1}{2} \int_x^1 (y-x)f(y)dy \quad (20)$$

$$= \frac{1}{2}xe^x + Ae^x + Be^{-x}, \quad ([11], \text{ p.394}),$$

where

$$A = \frac{1}{8} \frac{(e^2+5)(e+3e^{-1})}{(e-e^{-1})}, \quad B = \frac{1}{4}(e^2+7)-A.$$

Equation (20) is an integral equation of the differential equation $f''(x)-f(x)=e^x$.

$$f(x) = \frac{x}{12} + \int_0^x [\frac{x}{12} - 1]f(y)dy + \frac{x}{12} \int_x^{12} f(y)dy \quad (21)$$

Equation (21) is related to the parent differential equation $f''(x)+2f(x)f'(x)=0$, $f(0)=0$, $f(12)=1$. See (A0.2-8) and (A0.2-10).

0.4 NONLINEAR FREDHOLM EQUATIONS

$$f(x) = 1 + \frac{\lambda f(x)}{2} \int_0^1 \frac{x}{x+y} f(y) dy \quad (1)$$

Equation (1) is the Chandrasekhar equation studied by Stibbs and Wier [70]. Alternative forms are:-

$$(1-\lambda)^{1/2} f(x) = 1 - \frac{\lambda f(x)}{2} \int_0^1 \frac{y}{x+y} f(y) dy. \quad (2)$$

$$f(x) = 1 + \lambda f(x) \int_0^1 \frac{x^2 y f(y)}{x^2 + y^2} dy. \quad (3)$$

$$\varphi(x) = 1 - \frac{\lambda}{2} \int_0^1 \frac{x}{(x+y)\varphi(y)} dy \quad (4)$$

Note that $\varphi(x)$ in (4) is the inverse of $f(x)$ in equations (1)-(3). The solution to the above equations (1)-(3) and the inverse of (4) is given by

$$f(x) = \exp\left\{-\frac{x}{\pi} \int_0^{\pi/2} \frac{\ln(1-\lambda t \cot t)}{\cos^2 t + x^2 \sin^2 t} dt\right\}. \quad (5)$$

When $\lambda=1$,

$$f(x) = \left\{ \frac{1}{2} \int_0^1 \frac{y f(y)}{x+y} dy \right\}^{-1}. \quad (6)$$

$$f(x) = x^2 + 2x + \frac{1}{3} - \int_0^1 \{x+f(y)\}^2 dy \quad (7)$$

$$= x \text{ and } 5x-4, \quad ([11], \text{ p.692}).$$

$$f(x) = \int_0^1 \sqrt{\{x+y\}} f^2(y) dy, \quad ([11], [73]). \quad (8)$$

$$f(x) = 1 - 2 \int_0^1 F[f(y)] dy - 2 \int_x^1 \exp p(x-y) F[f(y)] dy. \quad (9a)$$

$$f(x) = p(\gamma-x) + 2 \int_0^x p(x-\gamma) F[f(y)] dy + \quad (9b)$$

$$p \int_x^1 \{2(\gamma-\gamma) F[f(y)] - f(y)\} dy,$$

where

$$F[f(y)] = f(y) \quad \text{linear case,}$$

$$= f^2(y) \quad \text{nonlinear case.}$$

and $\gamma=1+1/p$, p being the Peclet number, [91]. The solution in the linear case is

$$f(x) = \{2e^{(3-x)} + e^{2x}\} / \{4e^3 - 1\}. \quad (10)$$

Equation (11) is related to the equation $f''(x) + 2f(x)f'(x) = 0$, $f(0)=0$, $f(\infty)=1$, (see (A0.2-8)).

$$f(x) = \frac{\int_0^x \exp\{-2\int_0^y f(z)dz\}dy}{\int_0^\infty \exp\{-2\int_0^y f(z)dz\}dy} \quad (11)$$

$$f(x) = 2 \frac{\int_0^x (x-y) \exp\{-\int_0^y f(z)dz\}dy}{\int_0^\infty \exp\{-\int_0^y f(z)dz\}} \quad (12)$$

Equation (12) is a form of the Blasius equation [25].

$$f(x) = 1 - \frac{\int_0^x \exp\{-\int_0^y (y-z)^2 f(z)dz\}dy}{\int_0^\infty \exp\{-\int_0^y (y-z)^2 f(z)dz\}dy} \quad (13)$$

The differential form of equation (13) is

$$f'''(x) + 2f(x)f''(x) + \{1 - f'^2(x)\} = 0, \quad (14)$$

$$f(0) = f'(0) = 0, \quad f'(\infty) = 1$$

which occurs in the analysis of the flow near the stagnation point of a body of revolution (Weyl [149], Siekmann [28]).

$$f(x) = \int_0^b K(x,y)F[y;f(y)]dy, \quad ([16], p.23), \quad (15)$$

$$K(x,y) = -(\omega \sin \omega b)^{-1} \sin \omega \sin \omega (b-y) \quad x < y,$$

$$= -(\omega \sin \omega b)^{-1} \sin \omega (b-x) \sin \omega y \quad y < x,$$

is equivalent to

$$\frac{d^2 f(x)}{dx^2} + \omega^2 f(x) = F[x; f(x)], \quad f(0) = f(b) = 0.$$

A0.6 SYSTEMS OF LINEAR VOLTERRA EQUATIONS

Problem 1

$$f_1(x) = 1 + 2 \int_0^x f_2(y) dy \quad (1)$$

$$f_2(x) = 1 - 2 \int_0^x f_1(y) dy, \quad 0 \leq x \leq 1. \quad (2)$$

Solution:-

$$f_1(x) = \cos 2x + \sin 2x \quad (3)$$

$$f_2(x) = \cos 2x - \sin 2x. \quad (4)$$

Differential form:-

$$f_1'(x) - 2f_2(x) = 0 \quad (5)$$

$$f_2'(x) + 2f_1(x) = 0 \quad \text{with } f_1(x) = f_2(x) = 1. \quad (6)$$

Problem 2

$$f_1(x) = \frac{1}{27} \int_0^x [5E + 80f_2(y) - 250f_1(y)] dy \quad (7)$$

$$f_2(x) = \frac{2}{27} \int_0^x [-E - 70f_2(y) + 50f_1(y)] dy, \quad 0 \leq x \leq 1. \quad (8)$$

Solution:-

$$f_1(x) = E[441 - 105e^{-10x/3} - 336e^{-100x/9}]/22050 \quad (9)$$

$$f_2(x) = E[e^{-100x/9} - e^{-10x/3}]/105. \quad (10)$$

Differential form:-

$$7f_1'(x) + 4f_2'(x) + 50f_1(x) = E \quad (11)$$

$$2f_1'(x) + 5f_2'(x) + 20f_2(x) = 0, \quad f_1(0) = f_2(0) = 0. \quad (12)$$

Problem 3

$$f_1(x) = \frac{3}{2}(1 - \cos 4x) + \int_0^x (y-x)[8f_1(y) + 2f_2(y)] dy \quad (13)$$

$$f_2(x) = \int_0^x (y-x)[2f_1(y) + 5f_2(y)] dy, \quad 0 \leq x \leq 1. \quad (14)$$

Solution:-

$$f_1(x) = -\frac{22}{7} \cos 4x + \frac{2}{5} \cos 2x + \frac{96}{35} \cos 3x \quad (15)$$

$$f_2(x) = -\frac{4}{7} \cos 4x - \frac{4}{5} \cos 2x + \frac{48}{35} \cos 3x. \quad (16)$$

Differential form:-

$$f_1''(x) + 8f_1(x) + 2f_2(x) = 24 \cos 4x \quad (17)$$

$$f_2''(x) + 2f_1(x) + 5f_2(x) = 0 \quad f_1 - f_2 - f_1' - f_2' = 0, \quad x=0. \quad (18)$$

Problem 4

$$f_i(x) = \frac{1}{2} \exp(-0.2x) - 2 \int_0^x f_i(y) dy \quad i=1, 2. \quad (19)$$

$$f_3(x) = 0, \quad [86], \quad 0 \leq x \leq 1. \quad (20)$$

Solution:-

$$\underline{f}(x) = \left[\frac{1}{2} e^{-2x}, \frac{1}{2} e^{-2x}, 0 \right]^T. \quad (21)$$

Problem 5

$$f_1(x) = 1 + \int_0^x [-21f_1(y) + 19f_2(y) - 20f_3(y)] dy \quad (22)$$

$$f_2(x) = \int_0^x [19f_1(y) - 21f_2(y) + 20f_3(y)] dy \quad (23)$$

$$f_3(x) = -1 + \int_0^x [40f_1(y) - 40f_2(y) - 40f_3(y)] dy, \quad (24)$$

$$0 \leq x \leq 1, \quad [86].$$

Solution:-

$$f_1(x) = \frac{1}{2}e^{-2x} + \frac{1}{2}e^{-40x}(\cos 40x + \sin 40x) \quad (25)$$

$$f_2(x) = \frac{1}{2}e^{-2x} - \frac{1}{2}e^{-40x}(\cos 40x + \sin 40x) \quad (26)$$

$$f_3(x) = -e^{-40x}(\cos 40x - \sin 40x) . \quad (27)$$

Differential form:-

$$\underline{f}'(x) = \underline{A} \underline{f}(x), \quad \underline{f}(0) = [1, 0, -1]^T \quad (28)$$

where

$$\underline{A} = \begin{bmatrix} -21 & 19 & -20 \\ 19 & -21 & 20 \\ 40 & -40 & -40 \end{bmatrix} . \quad (29)$$

Problem 6

The same as problem 5 except that the range of integration is now $0.1 \leq x \leq 1$ and the free terms in equations (25)-(26) are given by

$$f_1(0.1) = 3.96448765 \times 10^{-1} \quad (30)$$

$$f_2(0.1) = 4.22281987 \times 10^{-1} \quad (31)$$

$$f(0.1) = -1.88942069 \times 10^{-3} . \quad (32)$$

A0.7 SYSTEMS OF NONLINEAR VOLTERRA EQUATIONS

Problem 1

$$f_1(x) = 75 \cos\left(\frac{7\pi}{30}\right) - c \int_0^x f_1(y) [f_1^2(y) + f_2^2(y)]^{1/2} dy \quad (1)$$

$$f_2(x) = 75 \sin\left(\frac{7\pi}{30}\right) - gx - c \int_0^x f_2(y) [f_1^2(y) + f_2^2(y)]^{1/2} dy \quad (2)$$

where $g=32 \text{ ft/s}$ and $c=2.291 \times 10^{-3}$.

Equations (1,2) give the horizontal and vertical components of velocity of a spherical non-spinning ball from which it is possible to compute its trajectory by straightforward integration.

Differential form:-

$$f_1'(x) = -cf_1(x) [f_1^2(x) + f_2^2(x)]^{1/2} \quad (3)$$

$$f_2'(x) = -cf_2(x) [f_1^2(x) + f_2^2(x)]^{1/2} - g \quad (4)$$

with

$$f_1(0) = 75 \cos\left(\frac{7\pi}{30}\right), \quad f_2(0) = 75 \sin\left(\frac{7\pi}{30}\right).$$

Problem 2

$$f_1(x) = 2 + \int_0^x f_2(y) dy \quad (5)$$

$$f_2(x) = \int_0^x [\epsilon \{1 - f_1^2(y)\} f_2(y) - a f_1(y)] dy \quad (6)$$

These equations are equivalent to Van der Pol's differential

$$f''(x) = -a f(x) + \epsilon [1 - f^2(x)] f'(x), \quad (7)$$

$$f(0) = 2, f'(0) = 0.$$

A0.9 SYSTEMS OF NONLINEAR FREDHOLM EQUATIONSProblem 1

$${}^1f(x) = \frac{1}{12} + \int_0^{12} {}^1K(x, y; {}^1f(y), {}^2f(y)) dy \quad (1)$$

$${}^1K(x, y; u, v) = -\frac{1}{6} yu[yu+v] \quad 0 \leq y \leq x,$$

$$= 2 \left[\frac{1-y}{12} \right] u[yu+v] \quad x \leq y \leq 12,$$

$${}^2f(x) = 2 \int_0^x y {}^1f(y) [y {}^1f(y) + {}^2f(y)] dy \quad (2)$$

This system is derived from the parent differential equation (A0.3-16).

APENDIX 1Summation abbreviations

The following notation is used to abbreviate summations:-

$$i) \quad \sum_{j=0}^N f_j = \frac{f_0}{2} + f_1 + f_2 + \dots + f_{N-1} + f_N.$$

$$ii) \quad \sum_{j=0}^N f_j = \frac{f_0}{2} + f_1 + f_2 + \dots + f_{N-1} + \frac{f_N}{2}.$$

$$iii) \quad \sum_{j=1}^{N+1} f_j = f_1 + f_3 + f_5 + \dots + f_{N-1} + f_{N+1}.$$

$$iv) \quad \sum_{j=0}^{N_{IV}} f_j = \frac{f_0}{2} + f_2 + f_4 + \dots + f_{N-2} + \frac{f_N}{2}.$$

$$v) \quad \sum_{j=2}^{N-2} f_j = f_2 + f_4 + f_6 + \dots + f_{N-4} + f_{N-2}.$$

APPENDIX 2

CLENSHAW-CURTIS INTEGRATION

These integration rules are defined over the range $[-1,1]$, $[4,36]$. Both definite and indefinite integrals may be approximated. Upon using the notation $Q(a,b,N,f)$ to denote the quadrature approximation

$$Q(a,b,N,f) \approx \int_a^b f(y) dy \quad (1)$$

these forms may be expressed in the following way:-

Definite Integration

$$Q(-1,1,N,f) \approx \sum_{i=1}^{N+1} b_i \cdot \quad (2)$$

Indefinite Integration

$$Q(-1,x,N,f) \approx \sum_{i=1}^{N+1} b_i [T_i(x) - (-1)^i], \quad (3)$$

$$Q(x,1,N,f) \approx \sum_{i=1}^{N+1} b_i [1 - T_i(x)] \quad (4)$$

where

$$b_i = (a_{i-1} - a_{i+1})/2i, \quad i=1,2,\dots,n-2, \quad (5)$$

$$b_{N-1} = (a_{N-2} - a_N/2)/2(N-1) \quad (6)$$

$$b_N = a_{N-1}/2N \quad (7)$$

$$b_{N+1} = a_N/4(N+1) \quad (8)$$

and

$$\begin{aligned} a_i &= \frac{2}{N} \sum_{j=0}^N f\left(\cos \frac{j\pi}{N}\right) T_i\left(\cos \frac{j\pi}{N}\right) \\ &= \frac{2}{N} \sum_{j=0}^N f\left(\cos \frac{j\pi}{N}\right) \cos \frac{ij\pi}{N} . \end{aligned} \quad (9)$$

Clenshaw and Curtis give the following error estimates if the Chebyshev series for the integrand $f(x)$ converges slowly:-

$$E_{ND}^{(1)} = \max\left[\frac{|a_N|}{4(N+1)}, \frac{|2a_{N-2}-a_N|}{32(N-1)}, \frac{|a_{N-4}-a_{N-2}|}{128(N-3)}\right] \quad (10)$$

for definite integrals, and

$$E_{N1}^{(1)} = \max\left[\frac{|a_N|}{4(N+1)}, \frac{|a_{N-1}|}{16N}, \frac{|2a_{N-2}-a_N|}{256(N-1)}\right] \quad (11)$$

in the indefinite case. If the series is slowly converging then

$$E_{ND}^{(2)} = \max[|a_N|, 2|a_{N-2}|, 2|a_{N-4}|] \quad (12)$$

and

$$E_N^{(2)} = \max[|a_N|, 2|a_{N-1}|, 2|a_{N-2}|] \quad (13)$$

are more likely to be bounds on the actual error but in the majority of cases not such close bounds. For further error estimates see O'Hara and Smith [29] and Elliott [30,32].

APPENDIX 3

INTERPOLATION FORMULAE

If $f(x)$ is a function whose value is known at the points x_i , $i=0(1)N$ on an interval $[a,b]$ of the real line, then an approximation, $P_N(x)$ to the function at any point x within $[a,b]$ may be defined by

$$f(x) = P_N(x) + E_N(x). \quad (1)$$

Here, N is the degree of the approximation and $E_N(x)$ is the error incurred by using the approximation.

LAGRANGE INTERPOLATION

The terms $P_N(x)$ and $E_N(x)$ for the general Lagrange interpolation formula using unequally spaced abscissae take the form

$$P_N(x) = \sum_{k=0}^N m_k(x) f(x_k) \quad (2)$$

where, using the notation

$$\Pi(x) = (x-x_0)(x-x_1)\dots(x-x_N), \quad (3)$$

$$m_k(x) = \Pi'(x) / \Pi'(x_k). \quad (4)$$

The primes indicate that the terms $(x-x_k)$ and (x_k-x_k) in the numerator and denominator respectively are omitted. The error is

given by

$$E_N(x) = \Pi(x) f^{(N+1)}(\zeta) / (N+1)! \quad (5)$$

where ζ is an unknown point in $[a, b]$ and it is assumed that the $N+1^{\text{th}}$ derivative of f exists at this point.

Chebyshev Interpolation

Two forms of Chebyshev interpolator may be used [36]. They are:-

$$P_N(x) = \sum_{r=0}^N a_r T_r(x), \quad (6)$$

where

$$a_r = \frac{2}{N} \sum_{k=0}^N f(x_k) T_r(x_k), \quad x_k = \cos \frac{k\pi}{N} \quad (7)$$

or

$$P_N(x) = \sum_{r=0}^N a_r T_r(x), \quad (8)$$

with

$$a_r = \frac{2}{N+1} \sum_{k=0}^N f(x_k) T_r(x_k), \quad x_k = \cos \left(\frac{2k+1}{N+1} \frac{\pi}{2} \right). \quad (9)$$

x_k , $k=0(1)N$ in (7) represent the maxima of the Chebyshev polynomial $T_N(x)$, whereas x_k , $k=0(1)N$ are the zeros of the polynomial $T_{N+1}(x)$. The errors induced by the two expansions (6)

and (8) satisfy discrete least-squares criteria. For equation (6), $E_N(x)$ satisfies

$$E = \sum_{k=0}^N E_N^2(x_k) = \text{minimum}$$

with

$$E_{\min} = \sum_{k=0}^N [f^2(x_k) - \sum_{r=0}^N a_r^2 T_r^2(x_k)]. \quad (10)$$

Equation (8) has a similar error satisfying

$$E_{\min} = \sum_{k=0}^N [f^2(x_k) - \sum_{r=0}^N a_r^2 T_r^2(x_k)]. \quad (11)$$

REFERENCES

1. L.M.Delves, J. Walsh, Numerical Solution of Integral Equations, Clarendon Press, Oxford (1974).
2. P.J.Davis, P.Rabinowitz, Numerical Integration, Blaisdell Pub. Co. (1967).
3. L.B.Rall, Numerical Integration and the Solution of Integral Equations by the Use of Riemann Sums, SIAM Rev. 7, no.1 (1965).
4. C.W.Clenshaw, A.R.Curtis, A method for numerical integration on an automatic computer, Numer. Math. 2 (1960), 197.
5. T.N.L. Patterson, The optimum addition of points to quadrature formulae, Math. Comp. 22 (1968), 213.
6. A. Ralston, A first course in numerical analysis, McGraw-Hill Book Co. (1965).
7. E.J. Nyström, Über die praktische Auflösung von Integralgleichungen mit Anwendungen auf Randwertaufgaben, Acta. Math. 54 (1930), 185.
8. H. Brakhage, Über die numerische Behandlung von Integralgleichungen nach der Quadraturformelmethode, Numer. Math. 2 (1960), 183.
9. K.E. Atkinson, Iterative variants of the Nyström Method for the numerical solution of integral equations, Numer. Math. 22 (1973), 15.
10. T.Marsh, M.Wadsworth, An iterative method for the solution of Fredholm integral equations of the second kind, J. Inst. Maths. Applics. 18 (1976), 57.
11. C.T.H. Baker, The numerical treatment of integral equations. Clarendon (1977).

12. G.A. Evans, J. Hyslop, A.P.G. Morgan, An extrapolation procedure for the evaluation of singular integrals, Intern. J. Computer Math., Section B 12(1982),1.
13. J.S.R. Chisholm, A. Genz, G.E. Rowlands, Accelerated Convergence of Sequences of Quadrature Approximations, J. Comp. Phys. 10 (1973), 284.
14. L.B. Rall, Computational Solution of Nonlinear Operator Equations, Wiley and Sons, Inc. (1969).
15. F.G. Tricomi, Integral Equations, Interscience (1965).
16. B.L. Moiseiwitsch, Integral Equations, Longman, London, (1977).
17. W.V. Lovitt, Linear integral equations, Dover Pub. Inc. (1924).
18. C.D. Green, Integral Equation Methods, Nelson (1969).
19. G. Hoheisel, Integral Equations, Nelson (1967).
20. Anderson et al, The application and numerical solution of integral equations, Sijthoff & Noordhoff (1980).
21. S.G. Mikhlin, K.L. Smolitsky, Approximate Methods for the Solution of Differential and Integral Equations, American Elsevier Publishing Company, Inc. (1967).
22. F. Ayres, Differential Equations, Schaum Outline Series, McGraw-Hill Book Co. (1963).
23. G.M. Murphy, Ordinary Differential Equations and their Solutions, Van Nostrand (1960).
24. E.L. Ince, Ordinary Differential Equations, Dover (1926).
25. W.F. Ames, Nonlinear Differential Equations In Transport Processes, Mathematics in Science and Engineering, A.P. (1967).
26. J.C. Burkhill, The Theory Of Ordinary Differential

- Equations, Oliver and Boyd Ltd (1968).
27. G.S.H. Lock, Approximate Closed Solutions to Certain Boundary Layer Problems, Bull Mech. Eng. Educ., 5, (1966), 71.
 28. J. Siekmann, Note on an Integral Equation Occurring - Stagnation Point Flow. ZAMP, 13 (1962), 182.
 29. H. O'Hara and F.H. Smith, Computer J, 11 (1968), 213.
 30. D.Elliott, A Chebyshev series method for the numerical solution of Fredholm integral equations. Computer J., 6 (1963), 102.
 31. D. Elliott, The Evaluation and Estimation of the Coefficients in the Chebyshev Series Expressions of a Function. Math. Comp. 18 (1964), 274.
 32. D. Elliott, Truncation Errors in two Chebyshev Series Approximations, Math. Comp. 19 (1965), 234.
 33. W. Fraser, M.W. Wilson, Remarks on the Clenshaw-Curtis quadrature scheme SIAM Review, 8 (1967), 322.
 34. A.H. Stroud, D. Secrest, Gaussian Quadrature Formulae, Englewood Cliffs N.J.:Prentice Hall (1966).
 35. P.J Davis, P.R. Rabinowitz, On the estimation of quadrature errors for analytical functions. Math. Comp. 8 (1954), 193, MR 16, 404.
 36. L. Fox, I.B. Parker, Chebyshev Polynomials in Numerical Analysis, Oxford University Press (1968).
 37. P. Linz, A Method for the approximate solution of linear Integro-differential Equations, SIAM, J. Numer. Anal. 11 (1974), 137.
 38. P.Henrici, Elements of Numerical Analysis, Wiley (1964).

39. H.Hochstadt, Integral Equations, J. Wiley & Sons (1973).
40. S.G. Mikhlin, Integral Equations, Pergamon Press (1957).
41. R.K. Miller, Nonlinear Volterra Integral Equations, Benjamin (1971).
42. P. Linz, The numerical solution of Volterra integral equations by finite difference methods. M.R.C. Tech. Rept., 825 (1968).
43. M. Kobayasi, On numerical solution of the Volterra integral equations of the second kind by linear multistep methods. Rep. Stat. Appl. Res. JUSE 13 (1966), 1.
44. D.F. Mayers, In "Numerical solution of ordinary and partial diferential equations", Ed. L. Fox, Pergamon (1962).
45. B. Noble, In "Conference on the numerical solutions of Differential equations". Springer lecture Notes 109 (1969).
46. A.P.G. Morgan, A comparison of Certain numerical methods used in the solution of integral equations, M.Sc. dissertation (September 1977), Loughborough University.
47. L. Fox, E.T. Goodman, The numerical solutions of nonsingular linear integral equations, Ph.D. Trans. R. Soc. 245 (1953), 501.
48. F.C. Freidlander, The reflexion of sound pulse by cover parabolic reflector. Proc. Camb. Phil. Soc. 37 (1949), 134.
49. H. O'Hara, F. H. Smith, Error Estimation in the Clenshaw-Curtis quadrature formula. Comp. J. 11 (1968), 213.

50. W.M. Gentleman, Commun. Assoc. Comput. Math. 15 (1972), 337, 343.
51. A.M. Weinberg, E.P. Wigner, The Physical Theory of Neutron Chain Reactors. Univ. of Chicago Press, Chicago (1958).
52. G. Placzek, Phys. Rev. 69 (1960), 423.
53. T. Teichman, Nucl. Sci. and Eng. 7 (1960), 292.
54. D. Eidelman, J. Nucl. Energy 18 (1964), 45.
55. S. Glasstone, M.C Edlund, The elements of nuclear reactor theory, Van Nostrand Princeton N.J. (1952).
56. H.T. Davis, Introduction to Nonlinear Differential and Integral Equations, Dover, New York (1962).
57. L. Garey, Predictor-corrector methods for nonlinear Volterra integral equations of the second kind. Nord. Tidskr. Inf-behandl. 12 (1972), 325.
58. L. Garey, Solving nonlinear second kind Volterra equations by modified increment methods. SIAM J. numer. Anal. 12 (1975), 501.
59. J.P. Imhof, On the Method for Numerical Integration of Clenshaw and Curtis, Numer. Math. 5 (1963), 138.
60. Emden functions, Mathematical Tables, Brit. Assoc. for Advnce. Sci. Vol.2 (1932).
61. W.F. Ames, Nonlinear Ordinary Differential Equations in Transport Processes, Academic Press, New York (1968).
62. F.G. Tricomi, Differential Equations, Blackie, London (1961).
63. S.E. El-gendi, Chebyshev solution of differential, integral and integro-differential equations, Comp. Journ. (1969), 282-289.

64. H. Williams, D.Phil Thesis on solution of Nonlinear Volterra equations at Nat. Phys. Lab. under Dr. G. Miller.
65. G.A. Evans, J. Hyslop, A.P.G. Morgan, The Iterative Solution of a Nonlinear Fredholm Integral Equation Arising from a Chemical Reactor Problem, Intern. J. Computer Math. 11 (1982), 73.
66. T.L. Saaty, J. Bram, Nonlinear Mathematics. McGraw-Hill, New York (1964).
67. L.V. Kantorovich, V.I. Krylov, Approximate methods of higher analysis, New York, Noordhoff Groningen (1964).
68. J. Todd, Survey of numerical analysis, McGraw-Hill, New York (1962).
69. P.M. Anselone, Nonlinear integral equations. University of Wisconsin Press, Madison (1964).
70. D.W.N. Stibbs, R.E. Weir, On the H-functions for isotropic scattering. Mon. Not. R. astr. Soc. 119 (1959), 512.
71. S. Chandrasekhar, F.H. Breen, Ap. F. 106 (1947), 106, 143.
72. D. Elliott, W.G. Warne, An algorithm for the numerical solution of linear integral equations.
73. L. Collatz, Functional analysis and numerical mathematics. (transl. from German by H. Oser), Academic Press, New York (1966).
74. A. Young, The application of product-integration to the numerical solution of integral equations. Proc. R. Soc. A., 244 (1954), 561.
75. L. Lapidus, Digital Computation for Chemical Engineers. McGraw-Hill, New York (1962).
76. P. Wynn, General Purpose Vector Epsilon Algorithm Algol Procedures, Numer. Math. 6 (1964), 22.

77. P. Wynn, Acceleration Techniques in Numerical Analysis with particular reference to Problems in one independent variable - Proceedings of the IFIP Congress 1962. North Holland Publishing Co. pp. 149-156.
78. K. Topfar, Z. Math. Phys. 60 (1912), 397.
79. R. Butler, E. Kerr, An introduction to numerical methods, Pitman (1962).
80. Hartree, Numerical Analysis, 2nd ed., p. 156. O.U.P. (1958).
81. G. Bratu, Sur Les Equations Integrales Non Lineaires. Bull. Soc. Math. de France, 42 (1914), 113.
82. J.M. Ortega, W.C. Rheinboldt, Studies in Numerical Analysis, Vol. 2. Numerical solutions of Non-Linear Problems, SIAM, Philadelphia (1970).
83. T.L. Saaty, Modern Nonlinear equations, McGraw-Hill, New York (1967).
84. A. Hammerstein, 'Nichtlineare Integralgleichungen nebst Anwendungen, Acta. Math. 54 (1930), 'Ivar Fredholm in memoriam', pp. 117-76.
85. M. Golomb, Zur Theorie der nichtlinearen Integralgleichungen, Integralgleichungssysteme und allgemeinen Funtionalgleichungen, Math. Zeitschrift 39 (1934), 45.
86. J.D. Lambert, Computational Methods in Ordinary Differential Equations, Wiley (1973).
87. H.T. Davis, The Theory of the Volterra Integral Equation of the Second Kind. Indiana Univ. Studies, 17 (1930).
88. T. Lalesco, Theorie des Equations Integrales, Hermann et Fils, Paris (1912).

89. T. Hasegawa et. al., Generalised Chebyshev Interpolation and Its Application to Automatic Quadrature, Maths. of Comp. 41(1983), 537.
90. G.A. Evans, J. Hyslop, A.P.G. Morgan, Iterative Solution of Volterra Integral Equations using Clenshaw-Curtis Quadrature, J. Comp. Phys. 40 (1981).
91. H.W. Pakes, C. Storey, Solution of the Equations for a Tubular Reactor with Axial Diffusion by a Variational Technique, The Chemical Engineer No. 208 (1967).
92. L. Fox, The Numerical Solution of Two Point Boundary Value Problems in Ordinary Differential Equations, Clarendon Press, Oxford (1957).
93. V. Kourganoff, I.W. Busbridge, Basic Methods in Transfer Problems, Clarendon Press, Oxford (1952).
94. D. Shanks, Non-linear transformations of divergent and slowly convergent sequences, J. Math. Phys. 34 (1955), 1.
95. P. Wynn, On a device for computing the $e_m(S_n)$ Transformations, Math. Comp. 10 (1956), 91.
96. D. Levin, Development of non-linear transformations for improving convergence of sequences, Intern. J. Computer Math. 3 (1973), 371.
97. A.M. Cohen, Cautious Romberg extrapolation, Intern. J. Computer Math. 8 (1980), 137.
98. H.B. Keller, Numerical Methods for Two-Point Boundary Value Problems, Blaisdell (1968).
99. V.V. Sobolev, A New Method in the Theory of the Diffusion of Light, Russ. Astr. J. 28(1951), 355.
100. B. Noble, A Bibliography of Methods for Solving Integral Equations, University of Wisconsin, Mathematics Research

Centre, Report No. 73 (1971).

101. H. Kagiwada, R. Kalaba, Integral Equations Via Imbedding Methods, Addison-Wesley.
102. H. Kagiwada, R. Kalaba, Imbedding Methods for Integral Equations with Applications, JOTA 24 No. 1 (1978).
103. M.R. Scott, A Bibliography on Invariant Imbedding and Related Topics, Sandia Laboratories, Report No.SC-71-0886 (1971).
104. V.A. Ambarzumian, Diffuse Reflection of Light by a Foggy Medium, Doklady Akademii Nauk SSSR, 38 (1943), 229.
105. R.C. Allen, G.M. Wing, A Method for Accelerating the Iterative Solution of a class of Fredholm Integral Equations. JOTA 24 No. 1 (1978),
106. H.J. Kopp, Synthetic Method Solution of the Transport Equation, Nuclear Science and Engineering, 17 (1963), 65.
107. R. Bellman, Perturbation Techniques in Mathematics, Physics and Engineering, Holt, Rinehart and Winston, Inc. (1966).
108. C. Temple, "Linearization and Delinearization" Proc-Inter. Cong. of Math. (1958).
109. T. Von Karman, "The Engineer Grapples with Nonlinear Problems", Bull. Am. Math. Soc. 46 (1940), 615.
110. A. Wouk, Direct Iteration, Existence and Uniqueness, Nonlinear Integral Equations, ed. P.M. Anselone, Wisconsin (1964).
111. T.W. Mullikin, Nonlinear Integral Equations of Radiative Transfer, Nonlinear Integral Equations, ed. P.M. Anselone, Wisconsin (1964).

112. V.A. Ambarstsumian, "Diffusion of Light by Planetary Atmospheres", *Astron. Zhur.* 19 (1942), 30.
113. B. Carlson, Numerical Solution of Neutron Transport Problems, *Amer. Math. Soc. Proceedings of Symposia in Applied Mathematics* 10 (1961).
114. S. Chandrasekhar, D. Elgert, The illumination and Polarization of the Sunlit Sky on Rayleigh Scattering, *Trans. Amer. Philos. Soc. (N.S.)* 44, 643-728.
115. S. Chadrasekhar, D. Elbert, A. Franklin, The X and Y functions for Isotropic Scattering, I, *Astr. J.* 115 (1952), 244.
116. S. Churchill, C. Chee, L. Evans, L. Ten, S. Parg, Exact Solutions for Anisotropic, Multiple Scattering by Parallel Plane Dispersions, The University of Michigan, DASA-1257, 03675-1-F (1961).
117. K.I. Gross, Discussion of an Iterative Solution to an Equation of Radiative Transfer, *J. of Math. and Phys.* XLI, No. 1 (1962), 53-61.
118. L.B. Rall, Quadratic Equations in Banach Spaces, *Rendiconti del Circolo Matematico di Palermo, Serie II, Tomo X* (1961), 1.
119. W. Pogorzelski, Integral Equations and their Applications Vol. 1, Pergamon Press (1966).
120. D. Hilbert, Grundzüge einer allgemeinen Theorie der linearen Integralgleichungen. Leipzig (1912).
121. H. Poincare, Les methodes nouvelles de la mecanique celeste, Vol.1 Paris (1893).
122. G. Bertrand, La Theorie des marees et les equations integrales, *Ann. de l'Ec. Norm.* 40 (1923), 151.

123. F. Noether, Über eine Klasse singulärer Integralgleichungen, Math. Ann. 82 (1921), 42.
124. T. Carleman, Sur la resolution des marees et les equations integrales, Arkiv för matematik, astr. o phys (1922).
126. N.I. Muskhelishvili, Singular Integral Equations (Transl. from Russia by J.R.M. Radock), Noordhoff, Gröningen (1953).
127. I. Vekua, On linear singular integral equations containing integrals in the sense of the Cauchy principal value, Dokl. Acad. Nauk. S.S.S.R. 26 (1940), 335.
128. M. Blakemore, Numerical Application of a Variational Technique in Quantum Mechanics. Ph.D. Thesis, Loughborough University (1977).
129. F. Stenger, Integration formulae based on the trapezoidal formula, J. Inst. Math. Applic. 12 (1973), 102.
130. W. Squire, A quadrature method for finite intervals, Int. J. Num. Math. Engng. 10 (1976), 708.
131. W. Squire, A comment on quadrature in the presence of endpoint singularities Intern. J. Computer. Math, 7 (1979), 239.
132. C.B. Harris, W.A.B. Evans, Extension of numerical quadrature formulae to cater for end point singular behaviour over finite intervals, Intern. J. Computer Maths. 6 (1977), 219.
133. P.M. Morse, H. Feshbach, Methods of Theoretical Physics, McGraw Hill, New York (1953).
134. C.R. Morrow, A Study of Some Numerical Quadrature and

- Cubature Processes, Ph.D. Thesis, The Queen's University Belfast (1977).
135. L.M. Delves, L.F. Abd-Elal, Experience with the Integral Package FAG1, University of Liverpool, Research Report, GSS/78/8.
 136. L.M. Delves, A Fast Method for the Solution of Fredholm Integral Equations, JIMA 20 (1977), 173.
 137. L.M. Delves, L.F. Abd-Elal, J.A. Hendry, A Fast Galerkin Algorithm for Singular Integral Equations, JIMA 23 (1979), 139.
 138. A. Gerasoulis, R.P. Srivastav, On the Solvability of Singular Integral Equations via Gauss-Jacobi Quadrature, Intern. J. Computer. Math. 12 (1982), 59.
 139. F. Stenger, The approximate solution of Wiener-Hopf integral equations, J. Math. Analysis 37 (1972), 687.
 140. K.E. Atkinson, Extension of the Nyström method for the numerical solution of linear integral equations of the Second kind, Ph.D Thesis, University of Wisconsin, (Also MRC Tech. Summ. Rep. 686) (1966).
 141. K.E. Atkinson, The numerical solution of Fredholm-integral equations of the second kind, SIAM. J. numer. Anal. (Soc. inf. appl. Math.) 4 (1967), 337.
 142. K.E. Atkinson, The numerical solution of Fredholm integral of the second kind with singular kernels, Report, Computer Centre, Australian National University, Canberra (1972).
 143. J.L. Phillips, Collocation as a projection method for solving integral and other operator equations, Ph.D. thesis, Purdue University Lafayette, Indiana (1969).

144. S. Chandrasekhar, An Introduction to the Study of Stellar Structure, Chicago (1939).
145. B. Van der Pol, "On 'Relaxation Oscillations'", Phil. Mag. 2(7) (1926), 978.
146. Lord Rayleigh, "on Maintained Vibrations.", Phil. Mag. 15(1) (1883), 229.
147. H. Schlichting, Boundary Layer Theory, Mc Graw Hill (1960)
148. L.B. Benny, Mathematics for students of Engineering and Applied Science, Oxford (1966).
149. H. Weyl, Ann. Math. 43 (1942), 381.
150. L. Howarth, On the solution of the laminar boundary layer equations, Proc. Roy. Soc. London A164 (1938), 547.
151. C.W. Gear, The Numerical Integration of Ordinary Differential Equations, Mathematics of Computation 21, 146-156.

