AN ABSTRACT OF THE THESIS OF

FREDERICK THOMAS KROGH for the Ph.D. in MATHEMATICS
(Name) (Degree) (Major)

Date thesis is presented May 15, 1964

Title STABILITY AND ACCURACY OF PREDICT-CORRECT METHODS IN DIFFERENTIAL EQUATIONS

Abstract approved Redacted for privacy
(Major professor)

The thesis is divided into two parts. The first part is concerned with the error in formulas which may be used for numerical integration or the numerical solution of ordinary differential equations. The most general way of investigating the error is the one making use of the influence function G(s). The error may be written in the form

$$\int_{-\infty}^{\infty} f^{(n+1)}(s)\, G(s)\, ds,$$

where the function G is independent of the function f under consideration. It is proved that this approach gives the best possible bound on the error of the form $K \mid f^{(n+1)}(\xi) \mid$. When G does not change sign, the error may be expressed in the form

$$f^{(n+1)}(\xi) \int_{-\infty}^{\infty} G(s)\, ds.$$

Theorems relating to this are proved for some specific cases.

A computer program was written which makes use of $G(s)$ to determine the error in formulas of a general form. This program was used to investigate the error term of a large number of formulas, and it was found to be a practical method of finding the error in a systematic manner. Part one concludes with some interesting results obtained from the program.

The second part of the thesis is devoted to the selection of general purpose methods for the numerical solution of ordinary differential equations. A discussion of factors to be considered in the choice of method leads to predict-correct methods which use a predictor of the form

$$y_{n+1} = y_{n-1} + h \sum_{i=0}^{k} b_i \, y'_{n+i-k}$$

along with a corrector of the form

$$y_{n+1} = y_n + h \sum_{i=1}^{k+1} B_i \, y'_{n+i-k}.$$

Methods of this type are tabulated for $4 \leq k \leq 8$. Iterative starting procedures are recommended, and two examples of these are given. All of the formulas involved were derived using a program described in the thesis.

A major section is devoted to the problem of stability. The stability of a method when solving a system of real differential equations, depends on its stability for associated differential equations of the form $y' = \lambda y$, where λ may be complex. The importance of examining stability for complex values of λ is demonstrated. A procedure is developed for determining stability in actual practice, and the computer program used in this procedure is described.

Numerical results are given comparing various methods and verifying the theoretical conclusions on stability.

STABILITY AND ACCURACY OF PREDICT-CORRECT METHODS IN DIFFERENTIAL EQUATIONS

by

FREDERICK THOMAS KROGH

A THESIS

submitted to

OREGON STATE UNIVERSITY

in partial fulfillment of
the requirements for the
degree of

DOCTOR OF PHILOSOPHY

June 1964

APPROVED:

Redacted for privacy

Professor of Mathematics

In Charge of Major

Redacted for privacy

Chairman of Department of Mathematics

Redacted for privacy

Dean of Graduate School

Date thesis is presented May 15, 1964

Typed by Illa W. Atwood

ACKNOWLEDGMENT

The author wishes to express his appreciation to Professor Milne for his suggestions and criticism, and especially for being such a wonderful person to work with.

TABLE OF CONTENTS

STABILITY AND ACCURACY OF PREDICT-CORRECT METHODS IN DIFFERENTIAL EQUATIONS

PART I

THE INFLUENCE FUNCTION G(s)

A. INTRODUCTION

It has been indicated (12, p. 149-152), and with some justification, that there is no really satisfactory general way to determine the error in formulas used for numerical integration and the numerical solution of ordinary differential equations. At present the most general approach is the one making use of the influence function $G(s)$, to be described below. The primary objection to this approach is the amount of labor required to determine if the function $G(s)$ is of constant sign, and if it is not, to evaluate

$$\int |G(s)|\, ds.$$

A person who wishes to examine a large collection of formulas, and who has access to a digital computer, would naturally let the computer do this job. This the author has done, and found it to be a satisfactory approach. A description of this program will be given. First it will be made clear why we are interested in adopting the $G(s)$ approach (theorems 1 and 2), and some results of general interest will be given. This will be followed by an investigation of the behavior of G(s) for a very simple class of formulas. After a description of the program for calculating G(s), results for some selected cases are given.

We shall consider the general quadrature formula

$$(1) \qquad y(c+hp_{00}) = \sum_{\mu=0}^{m} \sum_{\nu=1}^{n_\mu} A_{\mu\nu} h^{\mu} y^{(\mu)}(c+hp_{\mu\nu}) + R.$$

This formula is sufficiently general to include any formula which is likely to be useful for numerical integration or the numerical solution of differential equations, and yet permits us to obtain most of the results we desire. In (1) c, h, the $A_{\mu\nu}$'s and the $p_{\mu\nu}$'s are constants. The constant c positions the formula on the abscissa, h is a measure of the mesh size in numerical integration or of the step length in the numerical solution of differential equations, and the $p_{\mu\nu}$'s determine the points where values of the function and its derivatives are used. As usual, $y^{(\mu)}(c + hp_{\mu\nu})$ is the value of y's μ^{th} derivative evaluated at $c + hp_{\mu\nu}$. The quantity R is called the remainder and is our principal concern. In practice the coefficients $A_{\mu\nu}$ are determined such that R in (1) is zero if y is any polynomial of degree $\leq$ n. It can be shown that n is independent of c and h, and it is therefore appropriate to define the degree of (1) to be n.

When working with a formula of degree n, it will be assumed that the class of functions under consideration is the class C^{n+1}, that is, functions with continuous $(n+1)^{st}$ derivatives. For formulas of practical interest $m < n$ (m is the order of the highest derivative used in (1)), and we shall assume that this condition is satisfied.

In formula (1) we regard all quantities as being known except $y(c + hp_{00})$ and R. If formula (1) is replaced by

$$(2) \qquad y(c + hp_{00}) = \sum_{\mu=0}^{m} \sum_{\nu=1}^{n_\mu} A_{\mu\nu} h^{\mu} y^{(\mu)}(c + hp_{\mu\nu}),$$

we may calculate $y(c + hp_{00})$ directly, and R gives a measure of the error when this is done. Formula (2) is the one actually used, and in order to have some idea of the accuracy when using this formula, we desire some estimate of R. In (1) if y equals some function $f(x)$, then R depends on the function f. We shall denote this symbolically by writing $R(f)$ for the remainder, when a given function f is under consideration. It can be shown that

$$(3) \qquad R(f) = \int_{-\infty}^{\infty} f^{(n+1)}(s) G(s)\, ds, \text{ where}$$

$$G(s) = \frac{1}{n!} R_x [\overline{(x-s)}^n], \text{ and}$$

$$\overline{(x-s)}^n = \begin{cases} (x-s)^n & \text{if } x \geq s \\ 0 & \text{if } x \leq s. \end{cases}$$

See, for example, (20, p. 501-511) and (19, p. 108-114). The function $G(s)$ is known as the generating or influence function. It is significant because it depends only on the formula (1), and not on the function f under consideration.

Let $A = c + \min_{\mu\nu} [hp_{\mu\nu}]$, and $B = c + \max_{\mu\nu} [hp_{\mu\nu}]$. From (3) it follows immediately, (19, p. 114), that

$$(4) \qquad |R(f)| \leq \max_{A \leq \xi \leq B} |f^{(n+1)}(\xi)| \int_{-\infty}^{\infty} |G(s)|\, ds.$$

B. GENERAL THEOREMS CONCERNING $G(s)$

Given a formula of degree n, suppose we seek a number K such that

$$(5) \qquad |R(f)| \leq K \max |f^{(n+1)}(\xi)| \text{ for every } f \epsilon C^{n+1}.$$

Such a K exists by (4).

Theorem 1: The smallest K satisfying (5) is

$$K = \int_{-\infty}^{\infty} |G(s)|\, ds.$$

Proof: If $G(s)$ is definite, i.e. $G(s)$ doesn't change sign, the result follows by applying the mean value theorem to (3), from which we get

$$R(f) = f^{(n+1)}(\xi) \int_{-\infty}^{\infty} G(s)\, ds. \tag{6}$$

Suppose $G(s)$ changes sign at one point s_o. Without loss of generality we may assume that

$$G(s) \begin{cases} \leq 0 & \text{if } s < s_o \\ \geq 0 & \text{if } s > s_o . \end{cases}$$

Let $M = \max|G(s)|$. Assume there exists a

$$K' < K = \int_{-\infty}^{\infty} |G(s)|\, ds,$$

such that

$$|R(f)| \leq K' \max|f^{(n+1)}(\xi)|$$

for every $f \epsilon C^{n+1}$. Let $K - K' = \epsilon$, and let f be a function such that

$$f^{(n+1)}(s) = \begin{cases} -1 & \text{if } s < s_o - (\epsilon/3M) \\ 1 & \text{if } s > s_o + (\epsilon/3M) \\ \frac{3M}{\epsilon}(s - s_o) & \text{if } s_o - (\epsilon/3M) \leq s \leq s_o + (\epsilon/3M). \end{cases} \tag{7}$$

Then f is a function in C^{n+1} and hence by (3)

$$R(f) = \int_{-\infty}^{\infty} f^{(n+1)}(s)\, G(s)\, ds$$

$$= \int_{-\infty}^{\infty} |G(s)|\,ds + \int_{s_o-(\epsilon/3M)}^{s_o+(\epsilon/3M)} f^{(n+1)}(s)G(s)ds - \int_{s_o-(\epsilon/3M)}^{s_o+(\epsilon/3M)} |G(s)|ds .$$

$$R(f) \geq K - \int_{s_o-(\epsilon/3M)}^{s_o+(\epsilon/3M)} M\, ds = K - \frac{2}{3}\epsilon = K' + \frac{\epsilon}{3} .$$

Since $\max |f^{(n+1)}(\xi)| = 1$, we have contradicted the assumption that $|R(f)| \leq K' \max |f^{(n+1)}(\xi)|$.

The extension of the proof to the case where $G(s)$ changes sign any finite number of times is obvious.

Theorem 2: The function $G(s)$ is definite if and only if the error term can be expressed in the form $R(f) = Kf^{(n+1)}(\xi)$, where K is independent of f.

Proof: If in (3) we take

$$f(x) = \frac{x^{n+1}}{(n+1)!} ,$$

we have

$$R\left(\frac{x^{n+1}}{(n+1)!}\right) = \int_{-\infty}^{\infty} G(s)\, ds. \tag{8}$$

Thus if a K exists it must satisfy

(9) $$K = R\left(\frac{x^{n+1}}{(n+1)!}\right) = \int_{-\infty}^{\infty} G(s)\, ds.$$

If G is definite (6) gives immediately

$$R(f) = K f^{(n+1)}(\xi).$$

This completes the proof of the "only if" part of the theorem.

If G is not definite, we may without loss of generality assume that

$$K = \int_{-\infty}^{\infty} G(s)\, ds \geq 0.$$

Suppose $G(s)$ changes sign at one point s_o. Again, as in theorem 1, we may assume without loss of generality that

$$G(s) \begin{cases} \leq 0 & \text{if} \quad s < s_o \\ \geq 0 & \text{if} \quad s > s_o. \end{cases}$$

Let

$$\epsilon = \int_{-\infty}^{\infty} |G(s)|\, ds - \int_{-\infty}^{\infty} G(s)\, ds.$$

It follows from the definition of G that $\epsilon > 0$. Let $M = \max|G(s)|$ as in the proof of theorem 1 and let f be a function which satisfies (7). As in the proof of theorem 1 we get

$$R(f) \geq \int_{-\infty}^{\infty} |G(s)|\, ds - \frac{2}{3}\epsilon = \int_{-\infty}^{\infty} G(s)\, ds + \epsilon/3.$$

Since $|f^{(n+1)}(\xi)| \leq 1$ there cannot exist a ξ such that

$$R(f) = f^{(n+1)}(\xi) \int_{-\infty}^{\infty} G(s)\,ds.$$

Again it is simple to extend the proof to the case where $G(s)$ changes sign any finite number of times.

The condition that $R(f)$ can be put in the form $K f^{(n+1)}(\xi)$ is equivalent to Daniell's condition (7, p. 238) that a formula be simplex. He calls a formula simplex if $R(f) = 0$ implies $f^{(n+1)}(\xi) = 0$ at some point ξ, $A < \xi < B$, where A and B are defined as in (4). Thus the condition that a formula be simplex is equivalent to the condition that a formula have a definite G function. In many cases proving a formula to be simplex is simpler than proving that $G(s)$ is definite. However, there is no direct way of determining whether some formulas are simplex. The power of the $G(s)$ approach lies in the fact that it works for all formulas of the form (1). If G is definite, R can be expressed in the form

$$R(f) = K f^{(n+1)}(\xi).$$

If G is not definite R can not be given this form, but we still are able to get an upper bound on the absolute value of the remainder.

For reasons which will be given later in this thesis, formulas of the form (1) with the $A_{\mu\nu}$'s chosen so as to maximize n (the degree), are of particular interest. Such formulas we shall call optimal formulas. The results given in the rest of this section will be for optimal formulas, with $m = 1$, and $n_0 = 1$. For formulas of this form the $A_{1\nu}$'s may be found by integrating Lagrange's interpolating polynomial.

Theorem 3: The influence function $G(s)$ is definite if neither $p_{00} < p_{1\nu} < p_{01}$ nor $p_{01} < p_{1\nu} < p_{00}$ is satisfied for any ν, $1 \leq \nu \leq n_1$.

Proof: Let values of $y'(x)$ be given for $x = c + hp_{1\nu}$, $1 \leq \nu \leq n_1$. If we use Lagrange's interpolating polynomial to find $y'(x)$ from these values, the error is given by

$$\frac{1}{(n_1)!} y^{(n_1+1)}(\xi) \prod_{\nu=1}^{n_1} (x - c - hp_{1\nu}).$$

Note that

$$\prod_{\nu=1}^{n_1} (x - c - hp_{1\nu})$$

does not change sign for x between $c + hp_{01}$ and $c + hp_{00}$. Integrating the error of the interpolating polynomial from $c + hp_{01}$ to $c+hp_{00}$, and making use of the mean value theorem, we get for the remainder in formula (1),

$$R = \frac{1}{(n_1)!} \; y^{(n_1+1)}(\xi') \int_{c+hp_{01}}^{c+hp_{00}} \prod_{\nu=1}^{n_1} (x - c - hp_{1\nu})\, dx.$$

It then follows from theorem 2 that $G(s)$ is definite.

This result is equivalent to one found in Kunz (18, p. 141), where slightly different terminology is used.

When determining $G(s)$ for a specific formula, two different methods may be used. We illustrate with an example.

Example: Let

$$y(c+hp_{00}) = y(c+hp_{01}) + h \sum_{\nu=1}^{n_1} A_{1\nu}\, y'(c+hp_{1\nu})$$

be a quadrature formula of degree n. Let $h > 0$, and let $p_{01} < p_{11} < p_{12} < \cdots < p_{1n_1} < p_{00}$. Then

$$G(s) = \frac{1}{n!} R[\overline{(x-s)}^n] = \frac{1}{n!} \{ \overline{(c+hp_{00}-s)}^n - \overline{(c+hp_{01}-s)}^n - nh \sum_{\nu=1}^{n_1} A_{1\nu} \overline{(c+hp_{1\nu}-s)}^{n-1} \}$$

If

$$c + hp_{1j} < s < c + hp_{1j+1},$$

then by the definition of $\overline{(x-s)}$,

$$(10) \quad G(s) = \frac{1}{n!} \{ (c+hp_{00}-s)^n - nh \sum_{\nu=j+1}^{n_1} A_{1\nu} (c+hp_{1\nu}-s)^{n-1} \}.$$

But since our quadrature formula is of degree n,

$$\frac{1}{n!} R[(x-s)^n] = 0.$$

Subtracting this from the right member of (10) we get

$$(11) \quad G(s) = \frac{1}{n!}\{(c+hp_{01}-s)^n + nh\sum_{\nu=1}^{j} A_{1\nu}(c+hp_{1\nu}-s)^{n-1}\}.$$

Formulas (10) and (11) illustrate the two different methods which may be used.

Theorem 4: If the degree of (1) is odd, and either $p_{00} < p_{1\nu} < p_{01}$ or $p_{01} < p_{1\nu} < p_{00}$ is true for all ν, $1 \leq \nu \leq n_1$, then G(s) is not definite.

Proof: Without loss of generality we may assume $p_{01} < p_{11} < p_{12} < \cdots < p_{1n_1} < p_{00}$, and that $h > 0$. Making use of the methods described above we get

for $c + hp_{01} < s < c + hp_{11}$, $G(s) = \dfrac{(c+hp_{01}-s)^n}{n!} < 0$, and

for $c + hp_{1n_1} < s < c + hp_{00}$, $G(s) = \dfrac{(c+hp_{00}-s)^n}{n!} > 0$.

We shall now state an important result given by Steffensen (26, p. 154-165).

First we define what is meant by a symmetric formula. A formula is said to be symmetric with equally spaced points if:

(a) When n_1 is odd it is possible to choose c and h such that the derivatives will be evaluated at $0, \pm 1, \pm 2, \cdots, \pm \frac{n_1 - 1}{2}$, and $p_{00} = -p_{01} = k$ (k an integer).

(b) When n_1 is even it is possible to choose c and h such that the derivatives will be evaluated at $\pm\frac{1}{2}, \pm\frac{3}{2}, \cdots, \pm \frac{n_1 - 1}{2}$, and $p_{00} = -p_{01} = k + \frac{1}{2}$.

The result given by Steffensen follows.

The error term for a symmetric formula with equally spaced points can be put in the form $R(f) = Kf^{(n+1)}(\xi)$, where $A < \xi < B$, and A and B are defined as in (4).

Thus by theorem 2 the function $G(s)$ for a symmetric formula with equally spaced points is definite. This result includes the well-known Newton-Cotes formulas.

Using arguments similar to those used by Steffensen, Bragg and Leach (2, p. 70-76) have extended this last result to some of the cases where the $p_{1\nu}$ are not equally spaced.

C. THE BEHAVIOR OF $G(s)$ FOR OPTIMAL FORMULAS WITH $m = 1$, $n_0 = 1$, AND $n_1 = 2$

In order to simplify notation we shall assume that c and h have been chosen in such a way that $c + hp_{00} = 1$ and $c + hp_{01} = 0$.

With c and h thus chosen we let $a = c + hp_{11}$ and $b = c + hp_{12}$. Integrating Lagrange's interpolation polynomial we get the formula

(12) $$y(1) = y(0) + \frac{\frac{1}{2} - b}{a - b}\, y'(a) + \frac{\frac{1}{2} - a}{b - a}\, y'(b).$$

It is easy to verify that $R(1) = R(x) = R(x^2) = 0$. It is also easy to show that $R(x^3) = 0$ if and only if $2 - 3(a + b) + 6ab = 0$. We shall consider the case when (12) is of degree 2, and hence $2 - 3(a + b) + 6ab \neq 0$. We may assume without loss of generality that $a < b$. By definition

(13) $$G(s) = \tfrac{1}{2}\left\{ \overline{(1 - s)}^2 - \overline{(-s)}^2 + \frac{1 - 2b}{b - a}\, \overline{(a - s)} - \frac{1-2a}{b-a}\, \overline{(b - s)} \right\}.$$

Making use of the theorems in section B we know that G will be definite except possibly for the cases:

(i) $a \leq 0 < b < 1$

(ii) $0 < a < b < 1$

(iii) $0 < a < 1 \leq b$.

We shall examine these separately.

(i) $a \leq 0 < b < 1$

$$\text{If } a \leq s \leq 0,\ 2G(s) = -\frac{1 - 2b}{b - a}\,(a - s) \begin{cases} \leq 0 & \text{if } \frac{1}{2} < b \leq 1 \\ = 0 & \text{if } b = \frac{1}{2} \\ \geq 0 & \text{if } 0 \leq b < \frac{1}{2}. \end{cases}$$

If $0 \leq s \leq b$, $2G(s) = s^2 + \frac{1 - 2b}{b - a}(s - a)$.

$$2G'(s) = 2s + \frac{1 - 2b}{b - a}$$

Thus G will have a minimum for

$$s = -\frac{\frac{1}{2} - b}{b - a},$$

provided

$$0 \leq -\frac{\frac{1}{2} - b}{b - a} \leq b,$$

which in turn is satisfied if and only if $b \geq \frac{1}{2}$. If $b < \frac{1}{2}$ then the minimum will occur at $s = 0$, or $s = b$. Substituting into (13) we get:

$$2G(0) = \frac{1 - 2b}{b - a}(-a) \begin{cases} \leq 0 & \text{if } \frac{1}{2} < b \leq 1 \\ = 0 & \text{if } b = \frac{1}{2} \\ \geq 0 & \text{if } 0 \leq b < \frac{1}{2}. \end{cases}$$

$$2G(b) = b^2 + 1 - 2b = (b - 1)^2 \geq 0,$$

with equality if and only if $b = 1$.

$$2G\left(-\frac{\frac{1}{2} - b}{b - a}\right) = \left(\frac{\frac{1}{2} - b}{b - a}\right)^2 + \left(\frac{1 - 2b}{b - a}\right)\left(-\frac{\frac{1}{2} - b}{b - a} - a\right)$$

$$= -\left(\frac{\frac{1}{2} - b}{b - a}\right)^2 - a\left(\frac{1 - 2b}{b - a}\right) \leq 0 \quad \text{for } b \geq \frac{1}{2},$$

which is the only case of interest.

If $b \leq s \leq 1$, $2G(s) = (1 - s)^2 \geq 0$.

We may thus conclude for case (i) that

$$G(s) \geq 0 \qquad \text{if } 0 < b \leq \tfrac{1}{2},$$

$$G(s) \leq 0 \qquad \text{if } b = 1, \text{ and}$$

$$G(s) \text{ is indefinite if } \tfrac{1}{2} < b < 1.$$

(ii) $0 < a < b < 1$

If $0 \leq s \leq a$, $2G(s) = s^2 \geq 0$.

If $a \leq s \leq b$, $2G(s) = s^2 + \dfrac{1 - 2b}{b - a}(s - a)$.

We have

$$2G(a) = a^2 \geq 0$$

and $$2G(b) = (b - 1)^2 \geq 0.$$

Again $G(s)$ will have minimum at $s = -\dfrac{\frac{1}{2} - b}{b - a}$. Thus in order that $G(s)$ be negative we must have

$$a < -\frac{\frac{1}{2} - b}{b - a} < b \quad \text{and} \quad G\left(-\frac{\frac{1}{2} - b}{b - a}\right) < 0.$$

We shall take these conditions one at a time.

$$a < -\frac{\frac{1}{2} - b}{b - a} \quad \text{implies}$$

(*) $$b > \frac{1 - 2a^2}{2(1 - a)}, \text{ and}$$

$$b > -\frac{\frac{1}{2} - b}{b - a} \quad \text{implies}$$

(**) $$a < \frac{2b^2 - 2b + 1}{2b}$$

Assume $b \leq \frac{1}{2}$. Then by (*)

$$\frac{1 - 2a^2}{2(1 - a)} \leq \frac{1}{2}.$$

And this implies $a \geq \frac{1}{2}$, which is impossible since we have assumed $a < b \leq \frac{1}{2}$. Hence $b > \frac{1}{2}$. Similarly if we assume $a \geq \frac{1}{2}$, by using (**) we will arrive at a contradiction. Thus $a < \frac{1}{2}$.

$$G\left(-\frac{\frac{1}{2} - b}{b - a}\right) = -\left(\frac{\frac{1}{2} - b}{b - a}\right)^2 - a\left(\frac{1 - 2b}{b - a}\right).$$

If

$$G\left(-\frac{\frac{1}{2} - b}{b - a}\right) < 0,$$

then

$$a\left(\frac{2b - 1}{b - a}\right) < \left(\frac{b - \frac{1}{2}}{b - a}\right)^2,$$

$$2a < \left(\frac{b - \frac{1}{2}}{b - a}\right),$$

and this implies

$$\frac{1}{2} + a < b.$$

It is easy to verify that (*) and (**) will be satisfied if $\frac{1}{2} + a < b$. This then is a necessary condition that $G(s)$ take negative values. It is also a sufficient condition.

If $b \leq s \leq 1$, $2G(s) = (1 - s)^2 \geq 0$.

We may thus conclude for case (ii) that

$$G(s) \geq 0 \quad \text{if} \quad b - a \leq \tfrac{1}{2}$$

$$G(s) \text{ is indefinite if } \quad b - a > \tfrac{1}{2}.$$

(iii) $0 < a < 1 \leq b$

This case is very similar to case (i), and only the results will be given.

$$G(s) \geq 0 \quad \text{if} \quad \tfrac{1}{2} \leq a < 1$$

$$G(s) \leq 0 \quad \text{if} \quad a = 0$$

$$G(s) \text{ is indefinite if } \quad 0 < a < \tfrac{1}{2}.$$

These results can be summarized as follows. Let $\alpha = \min\,[c + hp_{00},\ c + hp_{01}]$, and $\beta = \max\,[c + hp_{00},\ c + hp_{01}]$. Let a formula of the form (1) be given with $m = 1$, $n_0 = 1$, $n_1 = 2$, and of degree 2. Then the associated G function is definite except when the following condition is satisfied. One of the derivatives is evaluated in the interior of either $\left(\alpha, \frac{\alpha + \beta}{2}\right)$ or $\left(\frac{\alpha + \beta}{2}, \beta\right)$. The other derivative is evaluated at a point, which is a distance of at least $\left|\frac{\alpha + \beta}{2}\right|$ from the point where the first derivative is evaluated, and the two derivatives are evaluated on opposite sides of the point $\frac{\alpha + \beta}{2}$. Thus for optimal formulas of degree two it is rather simple to determine if $G(s)$ is definite. It appears unprofitable to continue this approach to the case $m = 1$, $n_0 = 1$, $n_1 = 3$ as the algebra becomes extremely messy.

D. THE PROGRAM FOR INVESTIGATING G(s)

As an aid in investigating the error in formulas for the numerical solution of differential equations, the author wrote a program which determines whether $G(s)$ is definite, and gives an estimate of

$$\int_{-\infty}^{\infty} |G(s)|\, ds.$$

The program was written for the "modified" Alwac IIIE at Oregon State University. The Alwac is a drum machine with an optimal access time of a half millisecond, a word length of 32 bits plus a sign bit, a direct access memory of 128 words, and a back up memory of 8000 words.

The program accepts as an input a formula of the form (2), with the restrictions:

(i) It is assumed that $c = 0$ and $h = 1$,

(ii) the $p_{\mu\nu}$ and $A_{\mu\nu}$ are rational numbers,

(iii) $m < 8$,

(iv) $n_0 + n_1 + \cdots + n_m < 15$, and

(v) n (the degree) < 12.

For formulas of high degree with "bad" coefficients overflow is liable to be encountered. This frequently happens when (iv) and (v) are both near their acceptable limits. Restriction (i) is not really much of a restriction since it is known that R is proportional to h^{n+1} and c affects nothing but the interval in which ξ must lie.

The program computes

$$R(x^j) = (p_{00})^j - \sum_{\mu=0}^{m} \sum_{\nu=1}^{n_\mu} A_{\mu\nu} \frac{j!}{(j-\mu)!} (p_{\mu\nu})^{j-\mu},$$

where

$$\frac{j!}{(j-\mu)!} (p_{\mu\nu})^{j-\mu}$$

is defined to be 0 if $\mu > j$, for $j = 0, 1, \cdots$ until $R(x^j) \neq 0$. An output is then made of

$$\frac{1}{(n+1)!} R(x^{n+1}),$$

which is the error when G is definite.

Values of the parameters h and c are determined so as to fit the points $c + hp_{\mu\nu}$ into the interval (0, 1). (Originally $h = 1$, and $c = 0$.) The program computes

$$G\left(\frac{s-c}{h}\right) = \frac{1}{n!} R\left[\left(\overline{p_{\mu\nu} - \left(\frac{s-c}{h}\right)}\right)^n\right] = \frac{1}{h^n n!} R\left[\left(\overline{c + hp_{\mu\nu} - s}\right)^n\right]$$

$$= \frac{1}{h^n n!}\left\{\left(\overline{c+hp_{00}-s}\right)^n - \sum_{\mu=0}^{m} \sum_{\nu=1}^{n_\mu} A_{\mu\nu} \frac{n!}{(n-\mu)!} \left(\overline{c+hp_{\mu\nu}-s}\right)^{n-\mu}\right\}$$

for $s = 0, .01, .02, \cdots, .99$. Relative maxima and minima of G are typed out. There was no special reason for evaluating G(s) at intervals of .01. For most formulas a larger interval could be used. It was found that with an interval of .01 one could be quite certain of learning whether G(s) was definite or not. This interval also allows a fairly accurate estimate of

$$\int_{-\infty}^{\infty} |G(s)|\, ds$$

to be obtained by calculating

$$(.01) \sum_{j=0}^{99} \left| G\left(\frac{.01j - c}{h}\right) \right| .$$

This estimate was adequate for the comparison of different methods, and of course if G is definite,

$$\frac{1}{(n+1)!} \quad R(x^{n+1})$$

gives the exact value of

$$\int_{-\infty}^{\infty} |G(s)|\, ds,$$

except perhaps for the sign. If a better estimate is desired in the case when G(s) is indefinite it is a simple matter to use a more refined method of numerical integration, such as Simpson's rule, to evaluate this integral.

An example of results obtained directly from the computer is given on page 69.

E. RESULTS FROM COMPUTATIONS

There are well-known formulas, such as Weddle's rule, for which the influence function is not definite. It was shown in Section C that there are optimal formulas with an indefinite G function. A closer examination reveals that in all of these cases the formulas wouldn't be useful for the numerical solution of differential equations. This leads to the following question. Are there optimal formulas with indefinite G functions which are practical for the numerical solution of differential equations? Many optimal formulas were investigated for the case $m = 1$, and to the best of the author's knowledge, the answer for this case is no.

With $m = 2$ it was not uncommon to find formulas with indefinite G functions, which were also of possible use in solving differential equations. In general, these formulas allow a bound for the error to be given, which is lower than would be expected from looking at closely related formulas with definite G functions. Since this is partially a subjective judgment, three typical examples have been selected for the interested reader. The first two formulas have indefinite G functions, and the last has a definite G function.

(i) $$y_{n+1} = y_{n-1} + 2hy'_{n-1} + h^2\left[\frac{1}{18}y''_{n+1} + \frac{52}{45}y''_n + \frac{13}{15}y''_{n-1} - \frac{4}{45}y''_{n-2} + \frac{1}{90}y''_{n-3}\right]$$

$$\int_{-\infty}^{\infty} |G_1(s)|\,ds \approx (.0042)h^7 \quad \text{and} \quad \int_{-\infty}^{\infty} G_1(s)\,ds \approx (.0032)\,h^7$$

(ii) $$y_{n+1} = y_n + hy'_{n-1} + h^2\left[\frac{97}{1440}y''_{n+1} + \frac{361}{360}y''_n + \frac{37}{80}y''_{n-1} - \frac{13}{360}y''_{n-2} + \frac{1}{288}y''_{n-3}\right]$$

$$\int_{-\infty}^{\infty} |G_2(s)|\,ds \approx (.002)\,h^7 \quad \text{and} \quad \int_{-\infty}^{\infty} G_2(s)\,ds \approx (-.0005)\,h^7$$

(iii) $$y_n = y_{n-1} + hy'_{n-1} + h^2\left[\frac{-17}{1440}y''_{n+1} + \frac{11}{72}y''_n + \frac{97}{240}y''_{n-1} - \frac{19}{360}y''_{n-2} + \frac{11}{1440}y''_{n-3}\right]$$

$$\int_{-\infty}^{\infty} G_3(s)\,ds = -\int_{-\infty}^{\infty} |G_3(s)|\,ds \approx (-.0037)\,h^7$$

In these formulas y_j is an abbreviation for $y(x_j)$ where $x_j = c + jh$. Let A and B be defined as in (4), and let ξ_i satisfy $A \leq \xi_i \leq B$. The error for (iii) is given by

$$-\,y^{(7)}(\xi_1)\int_{-\infty}^{\infty} |G_3(s)|\,ds \approx (-.0037)\,h^7 y^{(7)}(\xi_1).$$

In formula (i) the absolute value of the error is bounded above by

$$|y^{(7)}(\xi_2)|\int_{-\infty}^{\infty} |G_1(s)|\,ds \approx (.0042)\,h^7|y^{(7)}(\xi_2)|.$$

Thus even though the bound on the error for (i) is higher, it is possible that for a given y the absolute value of the error in (i) will be less than the absolute value of the error in (iii). This appears especially likely if $y^{(7)}$ does not change sign in the interval (A, B).

PART II

FORMULAS AND METHODS FOR SOLVING DIFFERENTIAL EQUATIONS

A. INTRODUCTION

This part of the thesis is devoted to numerical methods for solving the initial-value problem

(14) $$y' = f(x, y), \; y(x_0) = y_0,$$

where f, y, and y_0 are N dimensional vectors, and $y' = \frac{dy}{dx}$. To simplify the language we shall state things as if N = 1, being careful to indicate the vector character of y when this is of importance.

All commonly used methods for the numerical solution of (14) are step-by-step methods. A step-by-step method forms the solution a step at a time, by approximating the function y over the step with a suitably chosen approximating function. We divide the computations into two parts. One part is the evaluation of f(x, y) and any other computations which are peculiar to a given f. This part of the computation must be reprogrammed for each new problem. The rest of the computations are concerned with the integration of (14), and this part we call a method for the numerical solution of the initial-value problem (14). The method is programmed once, and then may be used for many different problems.

We now begin a discussion which will lead us to the selection of some methods, which are suitable for use as standard library routines in a computing center.

B. BASIC ASSUMPTIONS CONCERNING THE PROBLEMS TO BE SOLVED

It is clear that no method can be best for all possible problems. We shall be interested in methods effective for solving systems of differential equations likely to arise in applications. Before discussing possible methods it behooves us to look for common characteristics in these problems which will affect our choice of method.

It is becoming increasingly common for the system of equations (14) to be so complicated, that the evaluation of the derivatives takes a major amount of the total computation time. Furthermore it is common to carry the solution a large number of steps. These two characteristics are the result of problems being solved today which were not even attempted before digital computers became available.

Before selecting a method it is necessary to choose the functions which will be used in approximating the solution over each step. Polynomials have proven in practice to be more flexible and easier to work with than any other choice. We shall assume that over any reasonably small interval the solution can be approximated well by a polynomial, and that in most cases the approximation will be improved by increasing the degree of the polynomial. For some special problems other approximating functions will serve better. The following references give examples of non-polynomial approximating functions: (12, p. 216-219), (3, p. 63-78), (24, p. 491-493), (8, p. 375-379), and (9, p. 22-32).

C. PRELIMINARY DISCUSSION ON THE CHOICE OF METHOD

A typical step for a step-by-step method uses computed information about the solution $y(x)$ up to and including $x = x^*$, to calculate the value of $y(x^*+h)$. The quantity h is called the step length.

In computing $y(x^*+h)$ some methods use higher derivatives of y than appear in the differential equation. Methods of this type are practical for some problems. However, for problems as discussed in Section B the evaluation of higher derivatives is liable to be extremely time consuming. Thus methods of this type are not suitable for use as general purpose methods, and will not be considered further.

The remaining methods can be put into two classes. There are those of the Runge-Kutta type, for example see (21, p. 72-75), which make use of only $y(x^*)$ and $y'(x^*)$ to compute $y(x^*+h)$. Then there are those based on quadrature formulas, for example see (21, p. 53-71), which in addition use values of y and y' from previous steps.

The following factors should be considered in the choice of a method.

(i) Errors

There are three types of errors to be considered. These are truncation error, round-off error and instability. Instability will be discussed in detail later. Round-off error is a result of carrying only a finite number of decimal places, or in a computer, binary places. Round-off error will be discussed briefly in Section D. For a detailed discussion see (13, p. 37-47). Truncation error is the result of approximating the solution to (14) by a polynomial. From Section B, it follows that truncation error will generally be improved by increasing the degree of the approximating polynomial.

We shall call a method m^{th} order if the solution to (14) is approximated locally by an $(m-1)^{st}$ degree polynomial. For an m^{th} order method the truncation error is of order h^m, where h is the step length. (Some authors call this an $(m-1)^{st}$ order method.)

(ii) Check on Errors

In a general purpose method it is important to have a check on two types of error. These are machine error and truncation error. Even the most reliable computer will make occasional errors, and with no check on this many hours of computation are liable to be wasted. A check on truncation error is useful for adjusting h so as to get the required accuracy, without taking such a small h as to be inefficient.

A serious objection to Runge-Kutta type methods is that they give no check on errors without considerable extra computation. For this reason and (v) below they are not suitable for our purposes.

(iii) Starting the Solution

The remaining methods use information from previous steps to compute $y(x^*+h)$. Thus some special procedure is needed to get the solution started. Since we assume the solution is to be carried a large number of steps, the accuracy of the starting procedure is of greater importance than its speed. It is for this reason that the iterative starting procedures described in Section F are to be preferred. Two other frequently suggested methods for starting the solution are Taylor's series and the Runge-Kutta method. However Taylor's series are not suitable for machine computation. The Runge-Kutta method is not practical because of the difficulty in estimating its accuracy, and because we need starting procedures with higher order accuracy than the Runge-Kutta method gives.

(iv) Changing the Step Length

It is sometimes desirable in the course of a computation to change the step length. Nordsieck (23, p. 22-49) has devised methods which are well-suited for solving problems where the step length will be altered frequently. For most problems, however, it will be desirable to change the step length only occasionally. There are two commonly suggested methods for doing this. One way is using the starting procedure to start with $y(x^*)$ and $y'(x^*)$. The second way is to have programs for halving and doubling the step length, for example see (12, p. 208). The first way has the advantage of requiring little additional programming, and of allowing h to be changed by any factor. The second way has the advantage of being faster, and is probably preferable unless changes in h are rare.

(v) Evaluation of the Derivatives

From what has been said in Section B, it is clear that we desire a method which requires as few evaluations of the derivatives as is consistant with accuracy. Predict-correct methods, described in the next section, require two of these evaluations per step. The 5^{th} order Runge-Kutta method requires four and higher order Runge-Kutta methods require even more.

At present it appears that two evaluations of the derivatives per step are necessary to insure stability. In the section on stability it is shown that the 8^{th} order Adams' method, which requires but one evaluation per step is quite unstable. Hull and Creemer (15, p. 301) give computational evidence that predict-correct methods of the Adams' type which do not evaluate the derivatives after y has been corrected, are unstable. The question of whether stable methods can be developed which require but one evaluation of the derivatives per step is one worthy of further study.

D. PREDICT-CORRECT METHODS

A predict-correct method is defined by the difference equations

(15) $$p_{n+1} = \sum_{i=0}^{k} a_i y_{n+i-k} + h \sum_{i=0}^{k} b_i y'_{n+i-k}$$

(16) $$y_{n+1} = \sum_{i=0}^{k} A_i y_{n+i-k} + h \sum_{i=0}^{k} B_i y'_{n+i-k} + h B_{k+1} p'_{n+1} .$$

Associated with (15) and (16) respectively, are the quadrature formulas

(17) $$y_{n+1} = \sum_{i=0}^{k} a_i y_{n+i-k} + h \sum_{i=0}^{k} b_i y'_{n+i-k} + R_p$$

(18) $$y_{n+1} = \sum_{i=0}^{k} A_i y_{n+i-k} + h \sum_{i=0}^{k+1} B_i y'_{n+i-k} + R_c .$$

The a_i, b_i, A_i, and B_i are real numbers some of which may be zero. The y_j are an abbreviation for $y(x_j)$, where $x_j = x_0 + jh$. The symbols y'_j, p_{n+1} and p'_{n+1} are similarly defined. In (17) and (18) R_p and R_c denote the remainders as defined in Section A of Part I.

When used as a basis for a predict-correct method, (17) and (18) will be of the same degree. If R_p and R_c are both of order h^m the method is of m^{th} order. Since R_p and R_c are zero when $y = 1$, it follows from (17) and (18) that

$$\sum_{i=0}^{k} a_i = \sum_{i=0}^{k} A_i = 1. \tag{19}$$

A step in a predict-correct method is made as follows:

(i) Formula (15) is used to compute p_{n+1}. This is the predicted value of y_{n+1}.

(ii) The routine for evaluating the derivatives is then used to compute $p'_{n+1} = f(x_{n+1}, p_{n+1})$.

(iii) Formula (16) is used to compute y_{n+1}. At this stage an estimate of the truncation error is given by

$$\frac{R_c}{R_p - R_c} (p_{n+1} - y_{n+1}).$$

A sudden jump in this quantity is an indication of machine error.

(iv) Finally $y'_{n+1} = f(x_{n+1}, y_{n+1})$ is computed using the routine for evaluating the derivatives.

The choice of the quadrature formula (18) which will be used as a corrector is the most important, since both the truncation error and the stability of the method are largely determined by this choice.

The basic structure of the corrector is obtained by choosing k, and setting some of the A_i and B_i equal to zero. Some of the coefficients are then left free as parameters, and the remaining coefficients are found as functions of these parameters, in such a way as to maximize the degree of the formula.

To minimize round-off error

$$\sum_{i=0}^{k} |A_i|$$

should be kept small. By (19) the smallest value is 1. It is also a good idea to keep

$$\sum_{i=0}^{k+1} |B_i|$$

small, but this isn't as important since these coefficients are multiplied by h. By theorem 6 in the section on stability, in order that the formula be stable for all systems (14) when

$$\left| h \frac{\partial f}{\partial y} \right|$$

is arbitrarly small, it is necessary that the polynomial

$$x^{k+1} - \sum_{i=0}^{k} A_i x^i$$

have all its roots inside the unit circle except for the root at 1. That there is a root at 1 follows from (19).

With these things to take into consideration, many papers, for example (11, p. 42-47), (17, p. 355-357), (16, p. 31-47), and

(5, p. 104-117), have been devoted to the choice of coefficients in order to balance the conflicting interests of stability, round-off error, and truncation error. In all of these papers some of the coefficients have been left free for use as parameters. If optimal formulas (see Part I, Section B) were used instead, there would result formulas of higher degree which require the same amount of computation. The choice of optimal formulas requires a little care. Dahlquist (6, p. 51) has shown that many optimal formulas of the form (18) are strongly unstable.

Optimal formulas with more than one nonzero A_i have poor round-off characteristics. The only stable formula with one non-zero A_i is

(20) $$y_{n+1} = y_n + h \sum_{i=0}^{k+1} B_i y'_{n+i-k} + R_c .$$

If $B_i = 0$ for $i < j$ then this formula requires storage for y_{n+1}, y_n, y'_{n+1}, y'_n, . . . , y'_{n+j-k}.

When the storage requirements of (20) are taken into consideration, similar reasoning leads to the following two formulas for use as predictors.

(21) $$y_{n+1} = y_n + \sum_{i=0}^{k} b_i y'_{n+i-k} + R_p, \text{ and}$$

(22) $$y_{n+1} = y_{n-1} + \sum_{i=0}^{k} b_i y'_{n+i-k} + R_p .$$

These two possibilities have been examined. It was found that (22) has better coefficients, a better truncation error, and results in a more stable method than (21). These considerations have led to the methods

(23)
$$p_{n+1} = y_{n-1} + h \sum_{i=0}^{k} b_i y'_{n+i-k}$$

$$y_{n+1} = y_n + h \sum_{i=1}^{k} B_i y'_{n+i-k} + h B_{k+1} p'_{n+1} ,$$

where the b_i and B_i have been chosen to maximize the degree of the associated quadrature formulas. The method (23) is a $(k+1)^{st}$ order method.

A modification of the method composed of (15) and (16) which raises the order of the truncation error by one has been suggested by Hamming (11, p. 37-38). The modified method uses the formulas

(24)
$$p_{n+1} = \sum_{i=0}^{k} a_i y_{n+i-k} + h \sum_{i=0}^{k} b_i y'_{n+i-k}$$

$$m_{n+1} = p_{n+1} + \frac{R_p}{R_c - R_p} (p_n - c_n)$$

$$c_{n+1} = \sum_{i=0}^{k} A_i y_{n+i-k} + h \sum_{i=0}^{k} B_i y'_{n+i-k} + h B_{k+1} p'_{n+1}$$

$$y_{n+1} = c_{n+1} + \frac{R_c}{R_c - R_p} (p_{n+1} - c_{n+1}).$$

For methods of the type (23) it was found that a modified m^{th} order method gave results comparable to those for an $(m+1)^{st}$ order method. The computation time for each is about the same, and they require the same amount of storage. The $(m+1)^{st}$ order method is a little easier to program, and has advantages when starting and when changing the step-length.

E. A PROGRAM FOR OBTAINING OPTIMAL FORMULAS

In the course of this research it was desired to investigate optimal formulas of a general form. The author wrote a program for the Alwac IIIE (described in Part I, Section D) to derive such formulas.

The program finds the coefficients $A_{\mu\nu}$ with the restrictions:

(i) The $p_{\mu\nu}$ are rational numbers,

(ii) $m < 8$, and

(iii) $n_0 + n_1 + \cdot \cdot \cdot + n_m < 15$.

As in Section D of Part I overflow becomes a problem when

$n_0 + n_1 + \cdots + n_m$, and hence n (the degree), is large.

The $A_{\mu\nu}$ are found using the method of undetermined coefficients. This involves setting $c = 0$ and $h = 1$ in (2), and solving the system of equations which result from letting $y = 1, x, x^2, \cdots, x^{n_0+n_1+\cdots+n_m-1}$. This system is solved using Gauss-Jordan reduction as described in (25, p. 43-55). Rational arithmetic subroutines are used in the computations.

After the output of the $A_{\mu\nu}$, this program enters the one described in Section D of Part I to investigate the error. All of the formulas given in this thesis were derived using this program, although many of the formulas have already appeared in the literature. A sample of the format used in this program is given on page 68.

F. RECOMMENDED METHODS

In this section quadrature formulas for methods of the type (23) are given. The stability range (see Section G) for each method is indicated. Two examples of starting procedures are given. These starting procedures require that $f(x, y)$ be defined for some values of $x < x_0$, and that $y(x)$ have no singularity for these values. Starting procedures of this type were selected because they have better accuracy than those which restrict $x \geq x_0$. Similar starting procedures could be developed for the case in which the restriction $x \geq x_0$ must be satisfied.

In the formulas below, $y^{(n)}$ is an abbreviation for the n^{th} derivative of y evaluated at a point ξ which satisfies $A \leq \xi \leq B$, where A and B are defined as in (4).

5th Order Method

$$\text{Predictor:}\quad y_{n+1} = y_{n-1} + h\left[\frac{8}{3}y'_n - \frac{5}{3}y'_{n-1} + \frac{4}{3}y'_{n-2} - \frac{1}{3}y'_{n-3}\right] + \frac{29}{90}h^5 y^{(5)}$$

$$\text{Corrector:}\quad y_{n+1} = y_n + h\left[\frac{3}{8}y'_{n+1} + \frac{19}{24}y'_n - \frac{5}{24}y'_{n-1} + \frac{1}{24}y'_{n-2}\right] - \frac{19}{720}h^5 y^{(5)}$$

Stable for $|s| \leq .58$

6th Order Method

$$\text{Predictor:}\quad y_{n+1} = y_{n-1} + h\left[\frac{269}{90}y'_n - \frac{133}{45}y'_{n-1} + \frac{49}{15}y'_{n-2} - \frac{73}{45}y'_{n-3} + \frac{29}{90}y'_{n-4}\right] + \frac{14}{45}h^6 y^{(6)}$$

$$\text{Corrector:}\quad y_{n+1} = y_n + h\left[\frac{251}{720}y'_{n+1} + \frac{323}{360}y'_n - \frac{11}{30}y'_{n-1} + \frac{53}{360}y'_{n-2} - \frac{19}{720}y'_{n-3}\right] - \frac{3}{160}h^6 y^{(6)}$$

Stable for $|s| \leq .55$

7th Order Method

$$\text{Predictor:}\quad y_{n+1} = y_{n-1} + h\left[\frac{33}{10}y'_n - \frac{203}{45}y'_{n-1} + \frac{287}{45}y'_{n-2} - \frac{71}{15}y'_{n-3} + \frac{169}{90}y'_{n-4} - \frac{14}{45}y'_{n-5}\right] + \frac{1139}{3780}h^7 y^{(7)}$$

Corrector: $$y_{n+1} = y_n + h\left[\frac{95}{288}y'_{n+1} + \frac{1427}{1440}y'_n - \frac{133}{240}y'_{n-1} + \frac{241}{720}y'_{n-2} - \frac{173}{1440}y'_{n-3} + \frac{3}{160}y'_{n-4}\right] - \frac{863}{60480}h^7y^{(7)}$$

Stable for $|s| \leq .53$

8th Order Method

Predictor: $$y_{n+1} = y_{n-1} + h\left[\frac{13613}{3780}y'_n - \frac{1327}{210}y'_{n-1} + \frac{4577}{420}y'_{n-2} - \frac{10168}{945}y'_{n-3} + \frac{2687}{420}y'_{n-4} - \frac{89}{42}y'_{n-5} + \frac{1139}{3780}y'_{n-6}\right] + \frac{41}{140}h^8y^{(8)}$$

Corrector: $$y_{n+1} = y_n + h\left[\frac{19087}{60480}y'_{n+1} + \frac{2713}{2520}y'_n - \frac{15487}{20160}y'_{n-1} + \frac{586}{945}y'_{n-2} - \frac{6737}{20160}y'_{n-3} + \frac{263}{2520}y'_{n-4} - \frac{863}{60480}y'_{n-5}\right] - \frac{275}{24192}h^8y^{(8)}$$

Stable for $|s| \leq .39$

9th Order Method

Predictor: $$y_{n+1} = y_{n-1} + h\left[\frac{736}{189}y'_n - \frac{703}{84}y'_{n-1} + \frac{358}{21}y'_{n-2} - \frac{79417}{3780}y'_{n-3} + \frac{1748}{105}y'_{n-4} - \frac{3473}{420}y'_{n-5} + \frac{2222}{945}y'_{n-6} - \frac{41}{140}y'_{n-7}\right] + \frac{32377}{113400}h^9y^{(9)}$$

Corrector: $$y_{n+1} = y_n + h\left[\frac{5257}{17280} y'_{n+1} + \frac{139849}{120960} y'_n - \frac{4511}{4480} y'_{n-1} + \frac{123133}{120960} y'_{n-2} - \frac{88547}{120960} y'_{n-3} + \frac{1537}{4480} y'_{n-4} - \frac{11351}{120960} y'_{n-5} + \frac{275}{24192} y'_{n-6}\right] - \frac{33953}{362880} h^9 y^{(9)}$$

Stable for $|s| \leq .28$

The starting procedures below are used as follows:

(i) Set $y_j = y_0$ and $y'_j = y'_0$.

(ii) Compute $y_1, y'_1, y_{-1}, y'_{-1}, y_2, \ldots$.

(iii) Compare the last computed value in (ii) with the one obtained on the previous iteration. If these differ by more than one in the last binary place, repeat steps (ii) and (iii).

(iv) Enter the main integration routine.

7th Order Starting Procedure

$$y_1 = y_0 + h\left[\frac{11}{1440} y'_3 - \frac{31}{480} y'_2 + \frac{401}{720} y'_1 + \frac{401}{720} y'_0 - \frac{31}{480} y'_{-1} + \frac{11}{1440} y'_{-2}\right] - \frac{191}{60480} h^7 y^{(7)}$$

$$y_{-1} = y_0 + h\left[\frac{11}{1440} y'_3 - \frac{77}{1440} y'_2 + \frac{43}{240} y'_1 - \frac{511}{720} y'_0 - \frac{637}{1440} y'_{-1} + \frac{3}{160} y'_{-2}\right] - \frac{271}{60480} h^7 y^{(7)}$$

$$y_2 = y_0 + h\left[\frac{-1}{90} y'_3 + \frac{17}{45} y'_2 + \frac{19}{15} y'_1 + \frac{17}{45} y'_0 - \frac{1}{90} y'_{-1}\right] + \frac{1}{756} h^7 y^{(7)}$$

$$y_{-2} = y_0 + h\left[\frac{-1}{90} y'_3 + \frac{1}{15} y'_2 - \frac{7}{45} y'_1 - \frac{7}{45} y'_0 - \frac{43}{30} y'_{-1} - \frac{14}{45} y'_{-2}\right] + \frac{37}{3780} h^7 y^{(7)}$$

$$y_3 = y_0 + h\left[\frac{51}{160} y'_3 + \frac{219}{160} y'_2 + \frac{57}{80} y'_1 + \frac{57}{80} y'_0 - \frac{21}{160} y'_{-1} + \frac{3}{160} y'_{-2}\right] - \frac{29}{2240} h^7 y^{(7)}$$

10th Order Starting Procedure

$$y_1 = y_0 + h\left[\frac{-3233}{3628800} y'_4 + \frac{18197}{1814400} y'_3 - \frac{108007}{1814400} y'_2 + \frac{954929}{1814400} y'_1 + \frac{13903}{22680} y'_0 - \frac{212881}{1814400} y'_{-1} + \frac{63143}{1814400} y'_{-2} - \frac{12853}{1814400} y'_{-3} + \frac{2497}{3628800} y'_{-4}\right]$$

$$y_{-1} = y_0 + h\left[\frac{-2497}{362880} y'_4 + \frac{12853}{1814400} y'_3 - \frac{63143}{1814400} y'_2 + \frac{212881}{1814400} y'_1 - \frac{13903}{22680} y'_0 - \frac{954929}{1814400} y'_{-1} + \frac{108007}{1814400} y'_{-2} - \frac{18197}{1814400} y'_{-3} + \frac{3233}{3628800} y'_{-4}\right]$$

$$y_2 = y_0 + h\left[\frac{127}{113400} y'_4 - \frac{247}{14175} y'_3 + \frac{22223}{56700} y'_2 + \frac{17741}{14175} y'_1 + \frac{1087}{2835} y'_0 - \frac{109}{14175} y'_{-1} - \frac{247}{56700} y'_{-2} + \frac{23}{14175} y'_{-3} - \frac{23}{113400} y'_{-4}\right] - \frac{23}{113400} h^{10} y^{(10)}$$

$$y_{-2} = y_0 + h\left[\frac{23}{113400} y'_4 - \frac{23}{14175} y'_3 + \frac{247}{56700} y'_2 + \frac{109}{14175} y'_1 - \frac{1087}{2835} y'_0 - \frac{17741}{14175} y'_{-1} - \frac{2223}{56700} y'_{-2} + \frac{247}{14175} y'_{-3} - \frac{127}{113400} y'_{-4}\right] - \frac{23}{113400} h^{10} y^{(10)}$$

$$y_3 = y_0 + h\left[\frac{-369}{44800} y'_4 + \frac{8101}{22400} y'_3 + \frac{28809}{22400} y'_2 + \frac{17217}{22400} y'_1 + \frac{209}{280} y'_0 - \frac{4833}{22400} y'_{-1} + \frac{1719}{22400} y'_{-2} - \frac{389}{22400} y'_{-3} + \frac{81}{44800} y'_{-4}\right] + \frac{113}{89600} h^{10} y^{(10)}$$

$$y_{-3} = y_0 + h\left[\frac{-81}{44800} y'_4 + \frac{389}{22400} y'_3 - \frac{1719}{22400} y'_2 + \frac{4833}{22400} y'_1 - \frac{209}{280} y'_0 - \frac{17217}{22400} y'_{-1} - \frac{28809}{22400} y'_{-2} - \frac{8101}{22400} y'_{-3} + \frac{369}{44800} y'_{-4}\right] + \frac{113}{89600} h^{10} y^{(10)}$$

$$y_4 = y_0 + h\left[\frac{4063}{14175} y'_4 + \frac{22576}{14175} y'_3 + \frac{244}{14175} y'_2 + \frac{32752}{14175} y'_1 - \frac{1816}{2835} y'_0 + \frac{9232}{14175} y'_{-1} - \frac{3956}{14175} y'_{-2} + \frac{976}{14175} y'_{-3} - \frac{107}{14175} y'_{-4}\right] - \frac{94}{14175} h^{10} y^{(10)}$$

$$y_{-4} = y_0 + h\left[\frac{107}{14175} y'_4 - \frac{976}{14175} y'_3 + \frac{3956}{14175} y'_2 - \frac{9232}{14175} y'_1 + \frac{1816}{2835} y'_0 - \frac{32752}{14175} y'_{-1} - \frac{244}{14175} y'_{-2} - \frac{22576}{14175} y'_{-3} - \frac{4063}{14175} y'_{-4}\right] - \frac{94}{14175} h^{10} y^{(10)}$$

G. STABILITY

As an aid to understanding the problem of stability when solving (14), we consider the single differential equation

(25) $$y' = \lambda y,$$

where λ is a complex number, and $y' = \frac{dy}{dx}$.

When the predict-correct method composed of (15) and (16) is used to solve (25), the numerical solution is given by the solution y of the difference equations

(26)
$$p_{n+1} = \sum_{i=0}^{k} (a_i + sb_i)\, y_{n+i-k}$$
$$y_{n+1} = \sum_{i=0}^{k} (A_i + sB_i)\, y_{n+i-k} + s B_{k+1}\, p_{n+1}.$$

These equations result from (15) and (16) by substituting λy for y', λp for p', and then letting $s = h\lambda$. Frequently to simplify the analysis, the effect of the predictor is ignored. Chase (4, p. 457-468) has demonstrated that the predictor has a significant effect on the stability of a method, and we shall give further evidence of this. Substituting for p_{n+1} in the second equation of (26) yields the linear, homogeneous difference equation with constant coefficients

(27) $$y_{n+1} = \sum_{i=0}^{k} [A_i + s(B_i + a_i B_{k+1}) + s^2(b_i B_{k+1})]\, y_{n+i-k}.$$

Associated with (27) is the indicial equation

$$X^{k+1} - \sum_{i=0}^{k} [A_i + s(B_i + a_i B_{k+1}) + s^2(b_i B_{k+1})] X^i = 0 . \tag{28}$$

If (28) has distinct roots $r_0, r_1, \cdots, r_k$ the general solution to (27) is given by

$$y_n = c_0 r_0^n + c_1 r_1^n + \cdots + c_k r_k^n , \tag{29}$$

where the c_i are constants.

The solution to (25) is given by $y = ce^{\lambda x}$, where c is a constant. Since $y_n = y(x_0 + nh)$ and $\lambda h = s$ we have for the exact solution of (25)

$$y_n = c e^{\lambda x_0} (e^s)^n . \tag{30}$$

One of the terms in the right hand member of (29), say $c_0 r_0^n$, approximates this solution, and hence $r_0 \approx e^s$. The other roots $r_1, r_2, \cdots, r_k$ arise in the process of solving the $(k+1)^{st}$ order difference equation (27), and are extraneous to the true solution of (25). The inevitable effect of round-off error is to introduce non-zero c_i $1 \le i \le k$ into (29). These introduced constants are small, and if $|r_i| < 1$ do not cause any difficulty. But if $|r_i| > 1$ for some i, there results a spurious solution which grows exponentially with n. When this occurs the method is said to be unstable. If $|r_i| > 1$ but $|r_0| >> |r_i|$, instability is not a problem, and the method is called relatively stable. The case where the roots of (28) are not distinct (This can occur for only finitely many values of s.) leads to essentially the same results, and will not be considered further.

In order to study the more general problem (14), we linearize f(x, y), and assume that the propagation of the error in

$$y' = f(x^*, y^*) + (x-x^*)\, f_x(x^*, y^*) + \sum_{i=1}^{N} (y_i - y_i^*)\, f_{y_i}(x^*, y^*), \tag{31}$$

is similar to the propagation of the error in (14). For example, see (14, p. 202-214). In (31) x^* and y^* are constant, y_i and y_i^* are the i^{th} components of y and y^* respectively, and $f_{y_i} = \frac{\partial f}{\partial y_i}$. Since stability depends on the behavior of the extraneous roots and these are determined by the homogeneous equation, we may neglect the constant terms and those containing x. Thus the problem is reduced to considering the stability of

$$y' = G\, y, \tag{32}$$

where y is an N-dimensional vector and G is a constant NxN matrix. Milne and Reynolds (22, p. 46-50) have shown that the stability of (32) depends on the stability of each of the equations

$$z_i' = \lambda_i z_i \qquad i = 1, 2, \cdots, N, \tag{33}$$

where the λ_i are the latent roots of G. Each equation in (33) is of the same form as (25), which we have already investigated.

Since in practice the λ_i are difficult to estimate, it is useful to know that a method is stable for all s which satisfy $|s| \leq \sigma$. The constant σ gives a means of comparing the stability of different methods, and gives an indication of how small h should be chosen when stability is liable to be a problem. The determination of σ is our next concern.

The roots r_i of (28) are branches of a single analytic function $X(s)$, see for example (1, p. 223-229). The only singularities of X are poles and branch points. A pole at a branch point can occur only if the two highest powers of X have zero coefficients. Since this does not occur in practice, X is continuous at the branch points.

Theorem 5: Let X have a branch point of order k at s_0 and denote the branches connected at this point by X_j, $j = 0, 1, \cdots, k$. Let γ be the circle $|s - s_0| = \rho$, where ρ is sufficiently small to exclude other singularities of X from being inside or on γ. Then for one of the branches $X_{j*}(s)$ there is a point $s*$ on γ such that $|X_{j*}(s*)| > |X_j(s_0)|$.

Proof: The branches of X may be written in the form, (1, p. 222-227),

$$X_j(s) = \sum_{n=0}^{\infty} A_n e^{n\left(\frac{2j\pi + \theta}{k}\right) i} r^{n/k}, j=0, 1, \cdots, k, \text{where } s-s_0=re^{i\theta}.$$

The function $f(z) = \sum_{n=0}^{\infty} A_n z^n$ will be regular for $|z| \leq \rho^{1/k}$. Thus by the maximum principle there exists a point z^* on the circle $|z| = \rho^{1/k}$ such that $|f(z)| \leq |f(z^*)|$ for all z inside this circle. In particular $|f(0)| < |f(z^*)|$. Define $\arg z$ so that $0 \leq \arg z < 2\pi$. Let $r = |z|^k$, and $\theta = k \arg z - 2j\pi$, where j is chosen such that $0 \leq \theta < 2\pi$. Then $f(z) = X_j(s)$, for $s = s_0 + r e^{i\theta}$. If r^*, θ^*, and j^* are defined in a similar manner using z^*, then for all s satisfying $|s - s_0| < \rho$, $|X_{j^*}(s^*)| > |X_j(s)|$, $j = 0, 1, \cdots, k$. In particular $|X_{j^*}(s^*)| > |X_j(s_0)|$.

Theorem 6: Let R be a closed region in the s-plane in which k bounded branches $X_1, X_2, \cdots, X_k$ of X are connected and let ∂R denote the boundary of R. Then

$$\max_{\substack{1 \leq i \leq k \\ s_b \in \partial R}} |X_i(s_b)| > |X_j(s)|, \; j = 1, 2, \cdots, k \text{ and all } s \text{ in the}$$

interior of R.

Proof: Let s be an arbitrary point in the interior of R. If X_j is analytic at s then it doesn't have a maximum at s by the maximum principle. If s is a branch point of X then by theorem 5 it follows that X_j can't have a maximum at s for $j = 1, 2, \cdots, k$. Since R is closed and X is continuous there is a maximum someplace in R. Hence the maximum occurs on the boundary.

The only poles of X in the finite s-plane occur where the coefficient of the highest power of X vanishes. By (28) this never happens with a predict-correct method. For methods in which stability is determined by one formula, such as a method composed of a predictor only or a method which iterates the corrector, the poles can be found by inspection of the indicial equation.

Thus in order to determine if the extraneous roots are inside the unit circle for $|s| \leq \sigma$ we must:

(i) Inspect the indicial equation to verify that there are no poles for $|s| \leq \sigma$. (For "reasonable" methods with "reasonable" σ's this condition is satisfied.)

(ii) Establish that the extraneous roots are inside the unit circle for $|s| = \sigma$.

(iii) Demonstrate that r_0 (the root in the term approximating the true solution) has no branch point in common with the extraneous roots for $|s| \leq \sigma$.

In order to accomplish (ii), the author wrote a program for the Alwac III E which finds the indicial equation and its roots for various values of s. The program finds the indicial equation for either predict-correct methods, or for methods in which stability is determined by one formula. It is possible to increment arg s while holding $|s|$ constant or to increment $|s|$ while holding arg s constant. Newton's method is used to find the roots, with the last computed values of the roots serving as initial guesses to compute the next roots. An increment of 15 degrees on arg s was found sufficiently small to establish (ii) with reasonable certainty. A sample of the output from this program is on page 70.

For small values of $|s|$, as s moves in a circle about the origin in the s-plane, r_0 traces out a closed curve which is roughly a circle in the X-plane. If $|s|$ is increased there comes a point where r_0 connects with the extraneous roots. Figures 1 and 2 show the behavior of the roots just before and just after this connection is made. Figure 3 shows typical behavior of the roots for large values of $|s|$. When r_0 behaves as in figure 1 for $|s| = \sigma$, there is strong computational evidence that condition (iii) above is satisfied. The author hopes to give a rigorous proof of this in the future.

In figure 2 the part of the curve outside the unit circle approximates the curve which would result if $r_0 = e^s$. For these values of s, (29) gives an approximation to (30). Even though the rest of the curve is inside the unit circle, it can not be concluded that this method is stable for $|s| = .59$. At $s = -.59$ there is no root of the difference equation which approximates $e^{-.59}$, and hence (29) does not approximate (30). This is also the case for $|s| > .59$. Thus condition (iii) is a necessary condition that a method be stable for $|s| < \sigma$.

The 9^{th} order method has been selected to illustrate the effect of instability. Since it has the least truncation error and the least stability of the methods (23), the effect of instability is most apparent for this method. Instability was examined directly for the 9^{th} order method applied to the two systems of differential equations

$$(34) \quad \begin{aligned} y_1' &= y_2 \qquad y_1(0) = 0 \\ y_2' &= -y_1 \qquad y_2(0) = 1 \text{ and} \end{aligned}$$

$$(35) \quad y_3' = -y_3 \qquad y_3(0) = 1 .$$

The solutions to these problems are $y_1 = \sin x$, $y_2 = \cos x$, and $y_3 = e^{-x}$. Results from the computations are in table 1, while figures 4 and 5 indicate the stability to be expected from the theory. Unstable behavior of a solution is characterized by the error alternating in sign in successive steps and gradually increasing in absolute value. Since there are several extraneous roots interacting, unstable behavior won't always be this simple, but it will still be apparent.

The results in table 1 agree exactly with what the theory predicts in figures 4 and 5. The curves in figures 4 and 5 indicate that (35) should be more stable than (34) which has latent roots $\pm i$. This is seen to be the result in table 1. The solution to (35) with $h = .4$ corresponds to $\arg s = 180°$ in figure 4. In this case there are two extraneous roots which are nearly one in absolute value, and these roots are greater in absolute value than the root which approximates the true solution. Thus, this is a case which borders on instability. The results in table 1 for this case are difficult to classify as either stable or unstable.

The importance of considering complex values of s is clear. If only real values of s were considered the 9^{th} order method would be considered stable for $|s| \leq .4$. The solution to (34), of course, is unstable for $|s| = .35$. For all the predict-correct methods examined, the maximum absolute value of the extraneous roots for a given value of $|s|$ occurred in the neighborhood of $\arg s = 100°$. However, for methods where the stability is determined by one formula, it was found that $\arg s = 180°$ gave the maximum.

Figure 6 is included to show the importance of considering the effect of the predictor. If it were assumed that the stability of a method was the same as the stability of the corrector, we would conclude from figure 6 that the 9^{th} order method is stable for $|s| \leq .49$. This value is much greater than the correct value which is .28.

Figure 7 allows a comparison to be made of the stability of the 9^{th} order method which uses a predictor of the form (21), and the one which uses a predictor of the form (22), (see figure 5). The behavior of the roots is very similar for these two methods. The one in figure 5 has a small advantage. For example at $|s| = .25$, $\arg s = 105°$, the absolute value of the largest extraneous root in figure 5 is .921 while in figure 7 this value is .934.

The method consisting of only the predictor in figure 8 is considered strongly stable since when $s = 0$, $r_0 = 1$ and the extraneous roots are all equal to 0. In reality this method is not very stable, since there are values of s which satisfy $|s| < .05$ for which the method is unstable.

H. RESULTS AND CONCLUSIONS

Tables 2 and 3 give results for solving the system (34) and the system

$$
\begin{aligned}
y_1' &= a y_2 y_3 & y_1(0) &= 0 \\
y_2' &= -a y_1 y_3 & y_2(0) &= 1 \\
y_3' &= -\tfrac{1}{2} a y_1 y_2 & y_3(0) &= 1.
\end{aligned}
\tag{36}
$$

In the calculations $a = .7416298708$. The solution of (36) gives the Jacobi elliptic functions, $y_1 = \text{sn } ax$, $y_2 = \text{cn } ax$, and $y_3 = \text{dn } ax$. This problem was selected in order to test the methods on a nonlinear system of differential equations.

As expected, accuracy tends to improve as the order of the methods increases. Hamming's method is that described in (12, p. 206). It was selected as an example of a method which uses non-optimal quadrature formulas. Of the suggested methods which make use of an arbitrary parameter to determine the coefficients, this one seemed to the author to be as good as any. It is of 6^{th} order, but requires as much computation as the 7^{th} order method of the type (23). For the problems tried the 7^{th} order was definitely the better. The Runge-Kutta-Gill method (10, p. 96-108) is included in the comparisons since it is so frequently used.

The use of the modification (24) improved accuracy, but adding another derivative generally made more of an improvement. Since it is also simpler to make use of another derivative, present evidence favors the straight predict-correct methods over the modified ones.

In table 2 are given some results where the y values were carried double precision. From these results, and some other results not included in the thesis, this procedure doesn't seem to offer any advantage. This is especially true when it is taken into account that for an equivalent amount of computation a higher order single precision method could be used.

Of the methods tried either the 8^{th} or the 9^{th} order of the type (23), is best suited for use as a general purpose routine. Although the 9^{th} order method is the least stable one of this type, it is still sufficiently stable for problems likely to be encountered. Stability seems to decrease as order increases and from the present progression it appears that the 10^{th} order method of this type is too unstable to serve as a good general purpose method. It is frequently suggested that after the second evaluation of the derivatives, the value of y be corrected once again before predicting the next value of y. This appears to be a promising means of making the higher order methods more stable. Unless this second iteration is needed for purposes of stability, the extra computation time doesn't justify the gain in accuracy. A better way to get more accuracy is to use a higher order method.

I. FIGURES, TABLES, AND EXAMPLES OF MACHINE COMPUTATIONS

In the figures are plotted the paths that the roots follow as s moves in a circle about the origin in the s-plane. The small circles denote the position of the roots for arg s = 0. The curves are solid for $0° < \arg s < 180°$ and dashed for $180° < \arg s < 360°$. A short line is drawn through the curves where arg s = 180°. On segments where it can be done without crowding, this is also done at multiples of 30°.

In the tables we denote the error by

$$E[f(x)] = \text{correct value of } f(x) - \text{computed value of } f(x).$$

In table 3 E[sn] (for example) stands for E[sn 20a]. The number of steps taken in a computation is equal to $\frac{x}{h}$, where $x = 20$ in tables 2 and 3. We use the following abbreviations.

7^{th}	(for example) is the 7^{th} order method of the type (23).
7^{th} M	is the 7^{th} order method with the modification (24) added.
R-K-G	is the Runge-Kutta-Gill method.
H	is Hamming's method, see (12, p. 206).

H_1 is Hamming's method without the modification (24).

H_2 is the same as H_1 except after the second evaluation of the derivatives y is corrected again.

H_3 is Hamming's method without the modification being used on the predictor.

H_4 is Hamming's method without the modification being used on the corrector

d.p. signifies that the y values were carried double precision

In each of the examples of machine computation the number in the top line is the calling sequence to start the program. Refer to section D of Part I for the meaning of the notation used below.

The program for finding formulas has the general format

$$p_{00}p_{01}\cdots p_{0n_0} - p_{11}p_{12}\cdots p_{1n_1} - \cdots\cdots - p_{m1}\cdots p_{mn_m}$$

$$\begin{matrix} A_{01} & A_{02} & \cdots & A_{0n_0} \\ A_{11} & A_{12} & & A_{1n_1} \\ \vdots & & & \\ A_{m1} & A_{m2} & & A_{mn_m} \end{matrix}$$

$R\left[\frac{x^{n+1}}{(n+1)!}\right]$ h * (n + 1) decimal approximation of $R\left[\frac{x^{n+1}}{(n+1)!}\right]$

$$s_1 \quad \frac{1}{h^n} G\left(\frac{s_1 - c}{h}\right)$$

$$s_2 \quad \frac{1}{h^n} G\left(\frac{s_2 - c}{h}\right)$$

$$\vdots \qquad \vdots$$

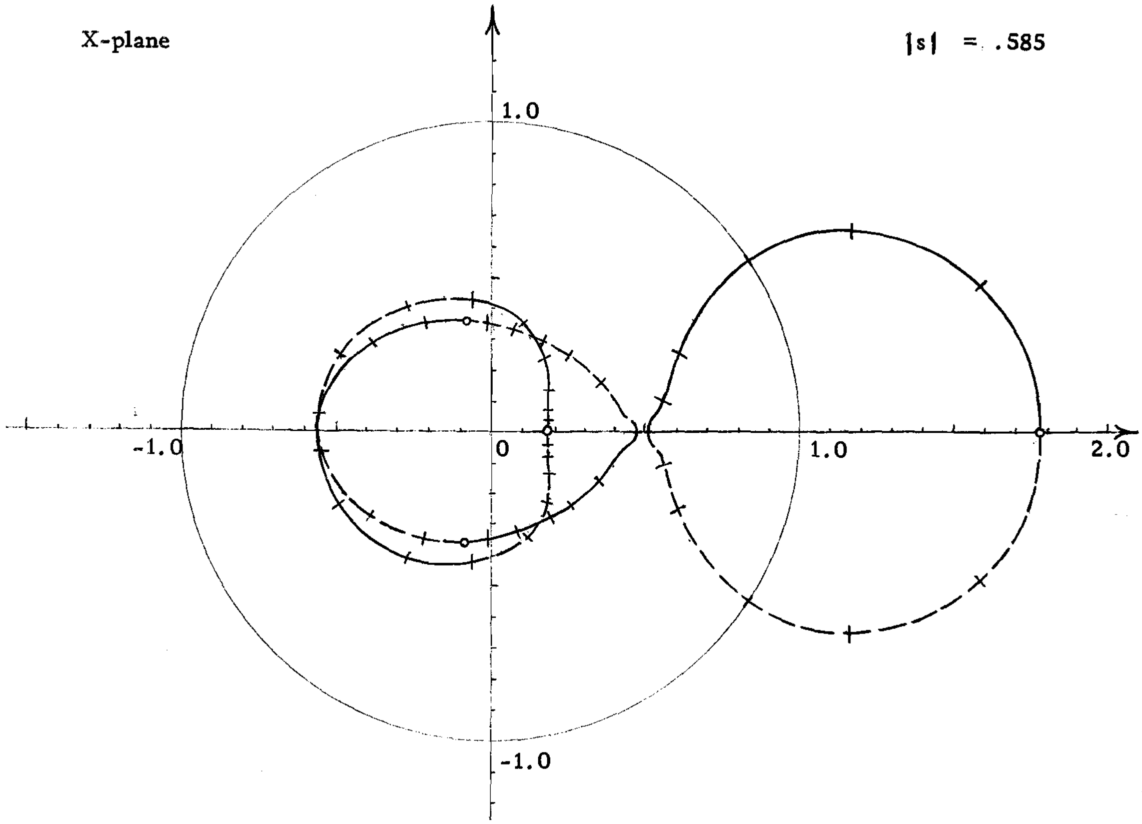
$$s_k \quad \frac{1}{h^n} G\left(\frac{s_k - c}{h}\right)$$

K = the approximate value of $\int_{-\infty}^{\infty} |G(s)| \, ds$.

The $p_{\mu\nu}$ must be scaled so that $\min(p_{\mu\nu}) = 0$. The "-" is used to indicate that μ should be increased by one, and a carriage return is used to terminate the inputs. The s_i denote values of s for which G has a relative maximum or minimum. A minus sign after the value of G indicates that G has a relative maximum at this point.

The routine for computing $G(s)$ uses the same general format. In this routine the $A_{\mu\nu}$ must be typed in as well as the $p_{\mu\nu}$. The values of $G(s)$ have been typed out at intervals of .01. The operator has a choice of this output or the one exhibited in the routine for finding the formulas by means of a switch on the console. The example given is Weddle's rule.

The program for investigating stability takes the same sort of input as the routine for computing $G(s)$, except twice, once for the corrector and once for the predictor. The indicial equation (28) is the first output. The computer then accepts inputs as shown. If $|\Delta| < 1$, $|s|$ is incremented instead of arg s. The columns of output from left to right give; the real part of the roots, the imaginary part, the absolute value of the roots, and the absolute value of the polynomial evaluated for the particular root. If desired just one root can be traced by setting a switch on the console.



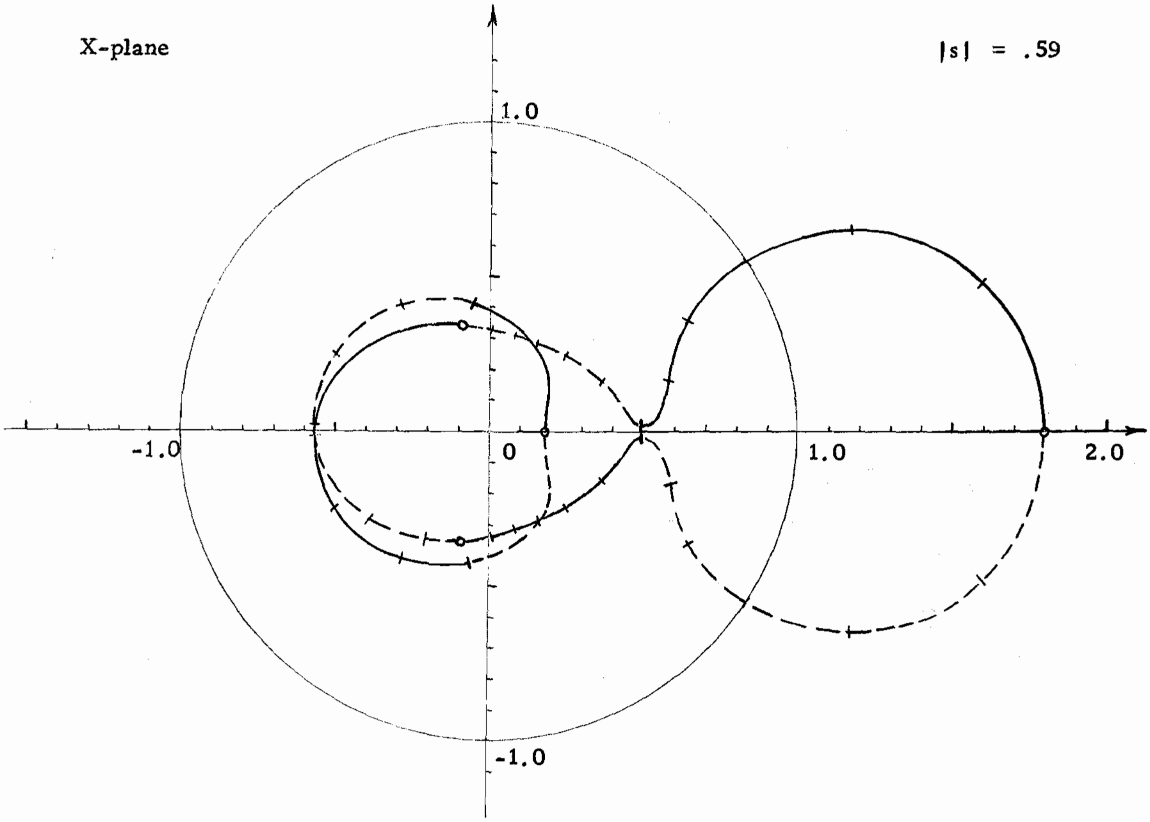
predictor: $y_{n+1} = y_{n-1} + h\left[\frac{8}{3}y'_n - \frac{5}{3}y'_{n-1} + \frac{4}{3}y'_{n-2} - \frac{1}{3}y'_{n-3}\right]$

corrector: $y_{n+1} = y_n + h\left[\frac{3}{8}y'_{n+1} + \frac{19}{24}y'_n - \frac{5}{24}y'_{n-1} + \frac{1}{24}y'_{n-2}\right]$

indicial equation: $X^4 - (1 + .791667s + s^2)X^3 + (-.166667s + .625s^2)X^2$

$-(.041667s + .5s^2)X + .125s^2 = 0$

Figure 1



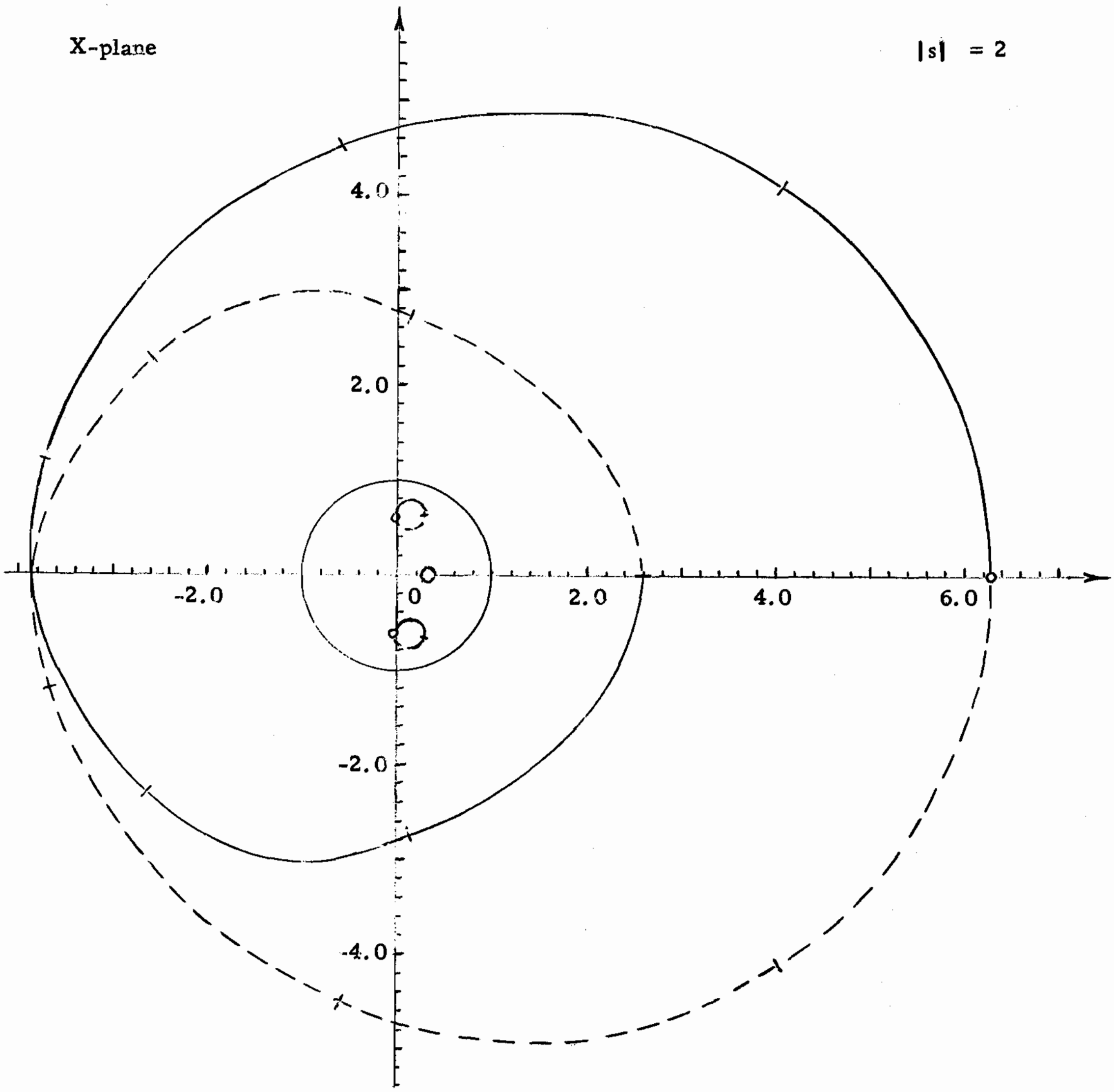
predictor: $y_{n+1} = y_{n-1} + h\left[\frac{8}{3}y'_n - \frac{5}{3}y'_{n-1} + \frac{4}{3}y'_{n-2} - \frac{1}{3}y'_{n-3}\right]$

corrector: $y_{n+1} = y_n + h\left[\frac{3}{8}y'_{n+1} + \frac{19}{24}y'_n - \frac{5}{24}y'_{n-1} + \frac{1}{24}y'_{n-2}\right]$

indicial equation: $X^4 - (1 + .791667\,s + s^2)\,X^3 + (-.166667\,s + .625\,s^2)\,X^2$

$-(.041667\,s + .5\,s^2)\,X + .125\,s^2 = 0$

Figure 2



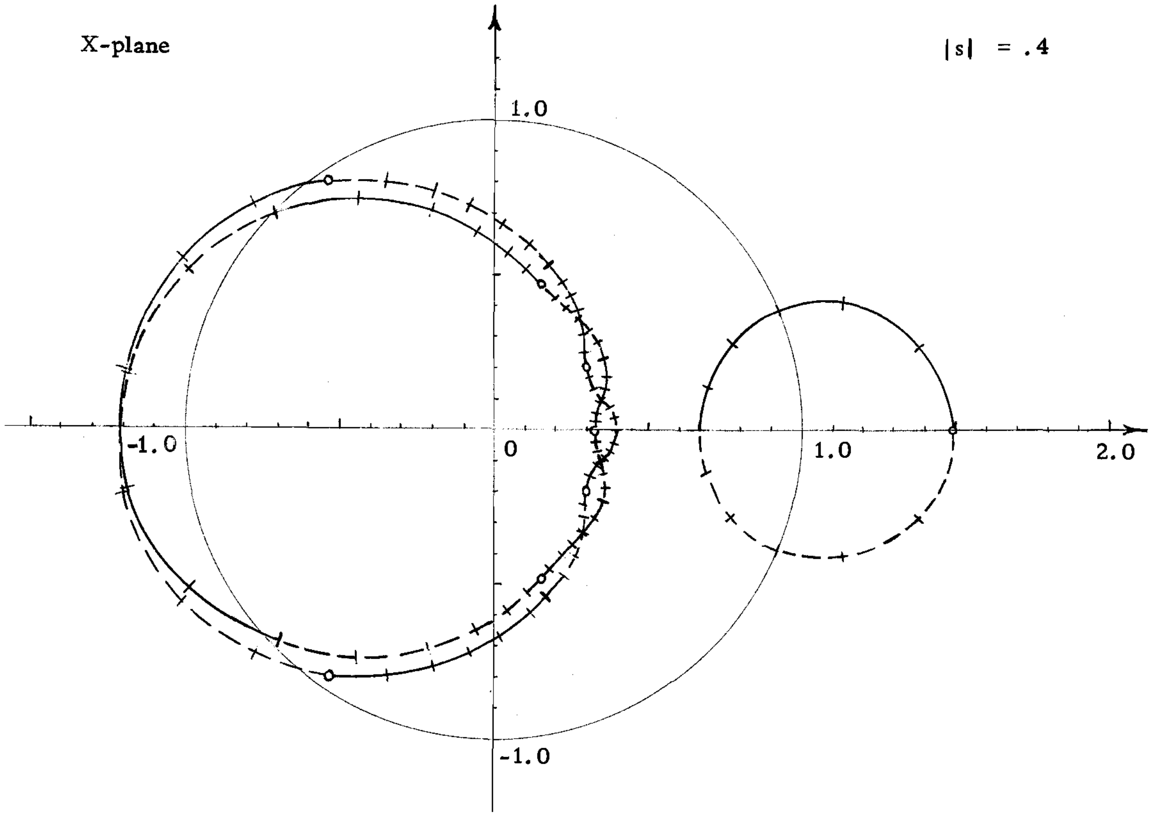
predictor: $y_{n+1} = y_{n-1} + h\left[\frac{8}{3}y'_n - \frac{5}{3}y'_{n-1} + \frac{4}{3}y'_{n-2} - \frac{1}{3}y'_{n-3}\right]$

corrector: $y_{n+1} = y_n + h\left[\frac{3}{8}y'_{n+1} + \frac{19}{24}y'_n - \frac{5}{24}y'_{n-1} + \frac{1}{24}y'_{n-2}\right]$

indicial equation: $X^4 - (1 + .791667\,s + s^2)\,X^3 + (-.166667\,s + .625\,s^2)\,X^2$

$- (.041667\,s + .5\,s^2)\,X + .125\,s^2 = 0$

Figure 3

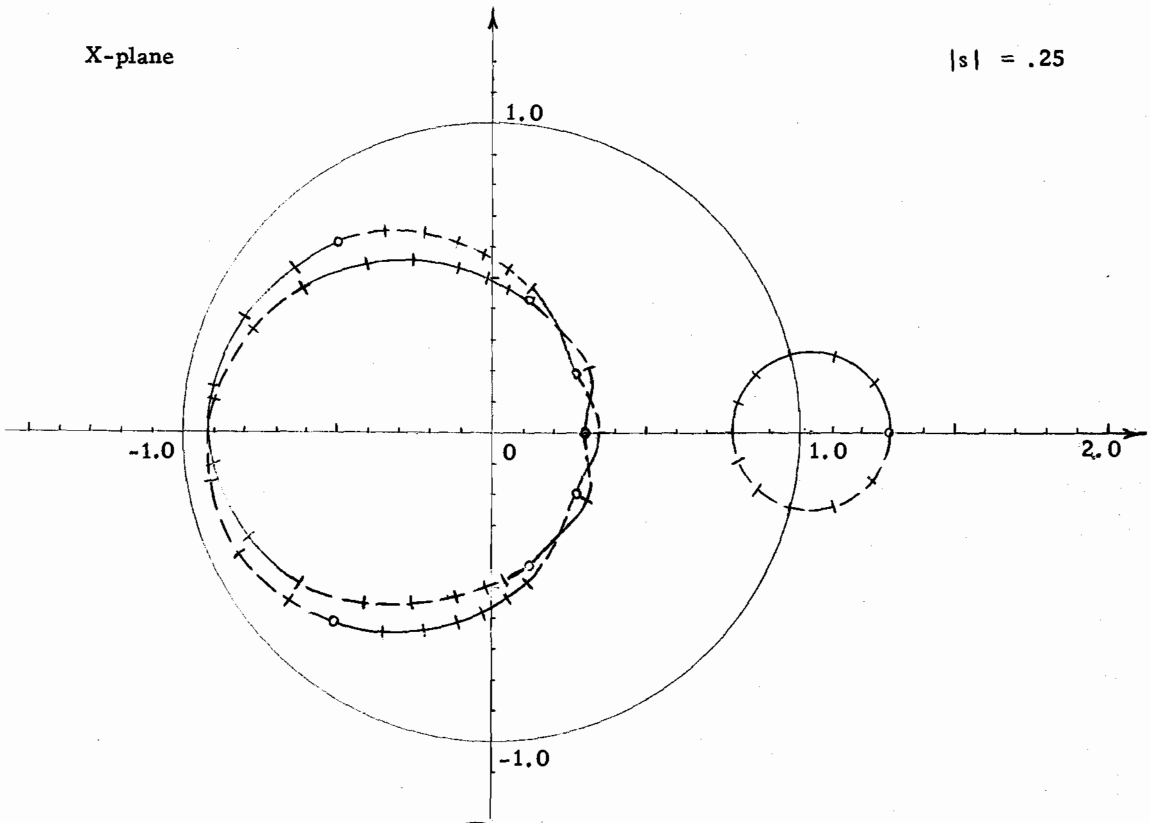


predictor: $$y_{n+1} = y_{n-1} + h\left[\frac{736}{189} y'_n - \frac{703}{84} y'_{n-1} + \frac{358}{21} y'_{n-2} - \frac{79417}{3780} y'_{n-3} + \frac{1748}{105} y'_{n-4} - \frac{3473}{420} y'_{n-5} + \frac{2222}{945} y'_{n-6} - \frac{41}{140} y'_{n-7}\right]$$

corrector: $$y_{n+1} = y_n + h\left[\frac{5257}{17280} y'_{n+1} + \frac{139849}{120960} y'_n - \frac{4511}{4480} y'_{n-1} + \frac{123133}{120960} y'_{n-2} - \frac{88547}{120960} y'_{n-3} + \frac{1537}{4480} y'_{n-4} - \frac{11351}{120960} y'_{n-5} + \frac{275}{24192} y'_{n-6}\right]$$

indicial equation: $$X^8 - (1 + 1.15616\,s + 1.18470\,s^2)X^7 + (.702695\,s + 2.54607\,s^2)X^6 - (1.01796\,s + 5.18630\,s^2)X^5 + (.732035\,s + 6.39169\,s^2)X^4 - (.343080\,s + 5.06461\,s^2)X^3 + (.093841\,s + 2.51565\,s^2)X^2 - (.011367\,s + .715330\,s^2)X + .089094\,s^2 = 0$$

Figure 4

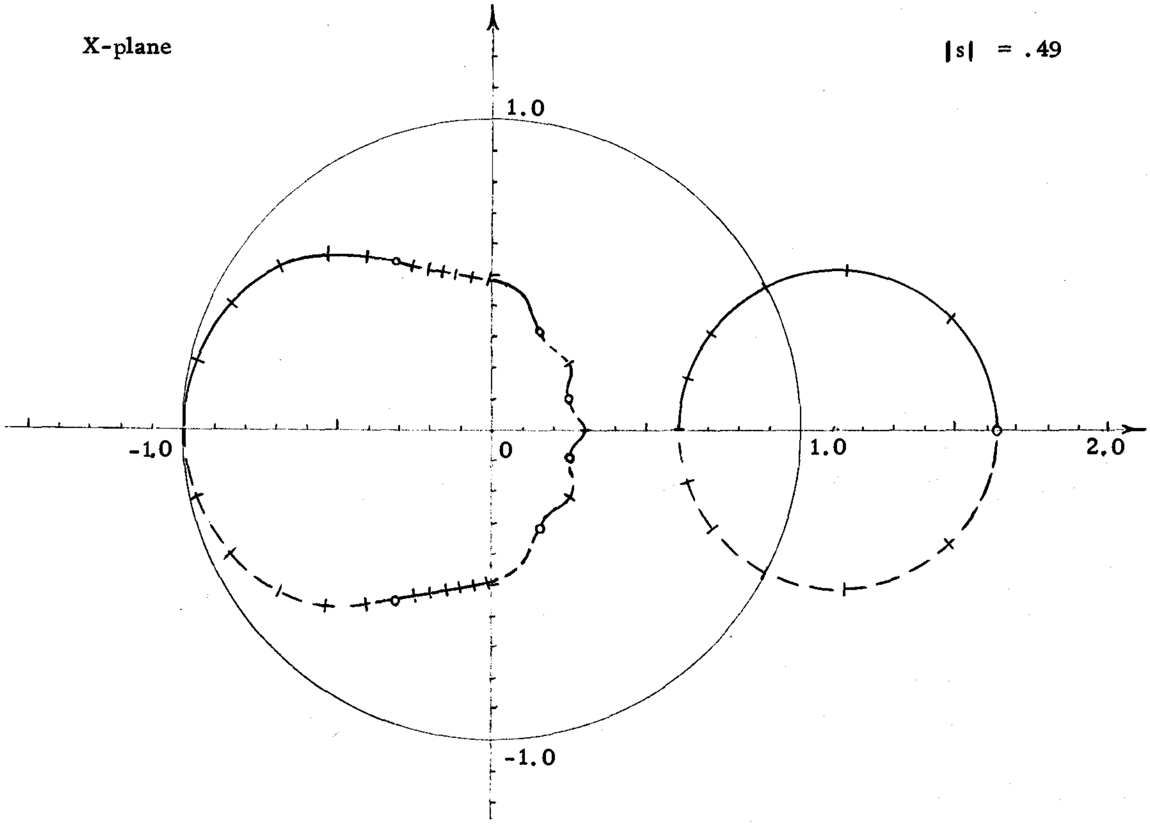


predictor: $$y_{n+1} = y_{n-1} + h\left[\frac{736}{189}y'_n - \frac{703}{84}y'_{n-1} + \frac{358}{21}y'_{n-2} - \frac{79417}{3780}y'_{n-3} + \frac{1748}{105}y'_{n-4} - \frac{3473}{420}y'_{n-5} + \frac{2222}{945}y'_{n-6} - \frac{41}{140}y'_{n-7}\right]$$

corrector: $$y_{n+1} = y_n + h\left[\frac{5257}{17280}y'_{n+1} + \frac{139849}{120960}y'_n - \frac{4511}{4480}y'_{n-1} + \frac{123133}{120960}y'_{n-2} - \frac{88547}{120960}y'_{n-3} + \frac{1537}{4480}y'_{n-4} - \frac{11351}{120960}y'_{n-5} + \frac{275}{24192}y'_{n-6}\right]$$

indicial equation: $$X^8 - (1 + 1.15616\,s + 1.18470\,s^2)X^7 + (.702695\,s + 2.54607\,s^2)X^6 - (1.01796\,s + 5.18630\,s^2)X^5 + (.732035\,s + 6.39169\,s^2)X^4 - (.343080\,s + 5.06461\,s^2)X^3 + (.093841\,s + 2.51565\,s^2)X^2 - (.011367\,s + .715330\,s^2)X + .089094\,s^2 = 0$$

Figure 5



$$y_{n+1} = y_n + h\left[\frac{5257}{17280}y'_{n+1} + \frac{139849}{120960}y'_n - \frac{4511}{4480}y'_{n-1} + \frac{123133}{120960}y'_{n-2}\right.$$

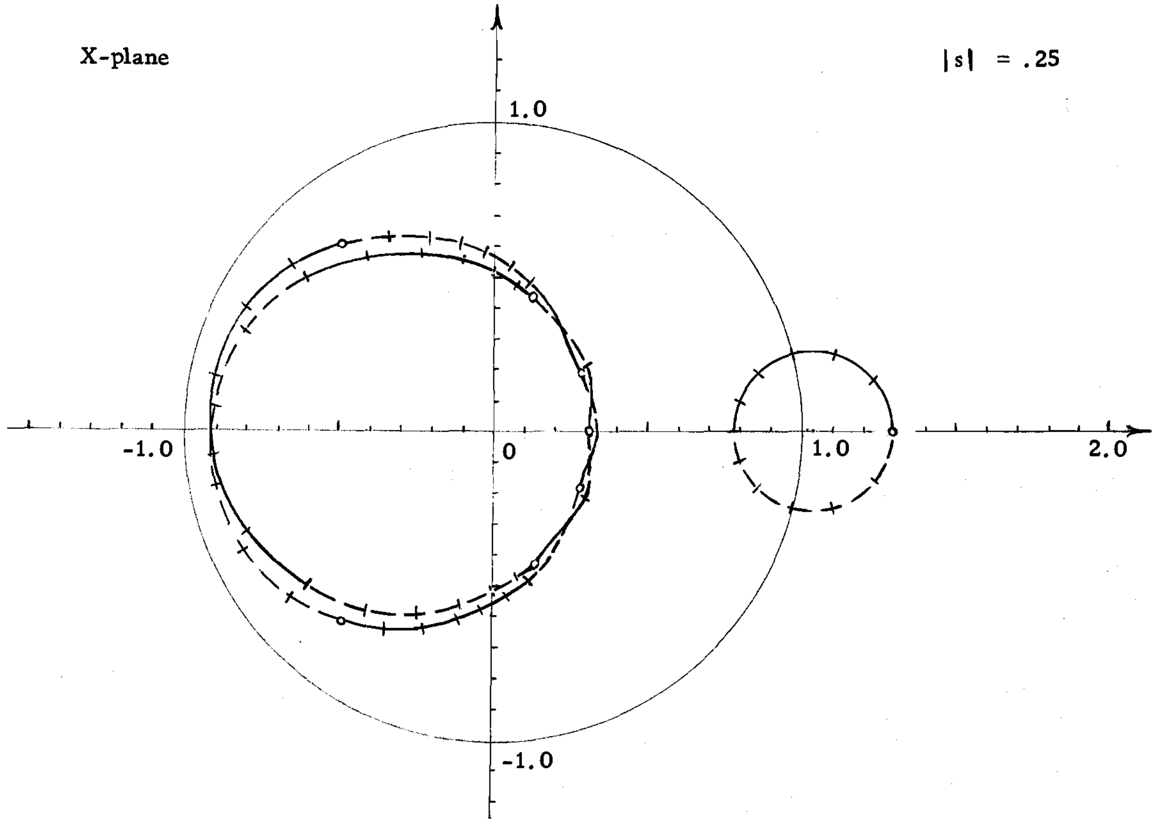
$$\left. - \frac{88547}{120960}y'_{n-3} + \frac{1537}{4480}y'_{n-4} - \frac{11351}{120960}y'_{n-5} + \frac{275}{24192}y'_{n-6}\right]$$

indicial equation: $(1 - .304225\,s)X^7 - (1 + 1.15616\,s)X^6 + (1.00692\,s)X^5$

$- (1.01796\,s)X^4 + (.732035\,s)X^3 - (.343080\,s)X^2$

$+ (.093841\,s)X - .011367\,s = 0$

Figure 6

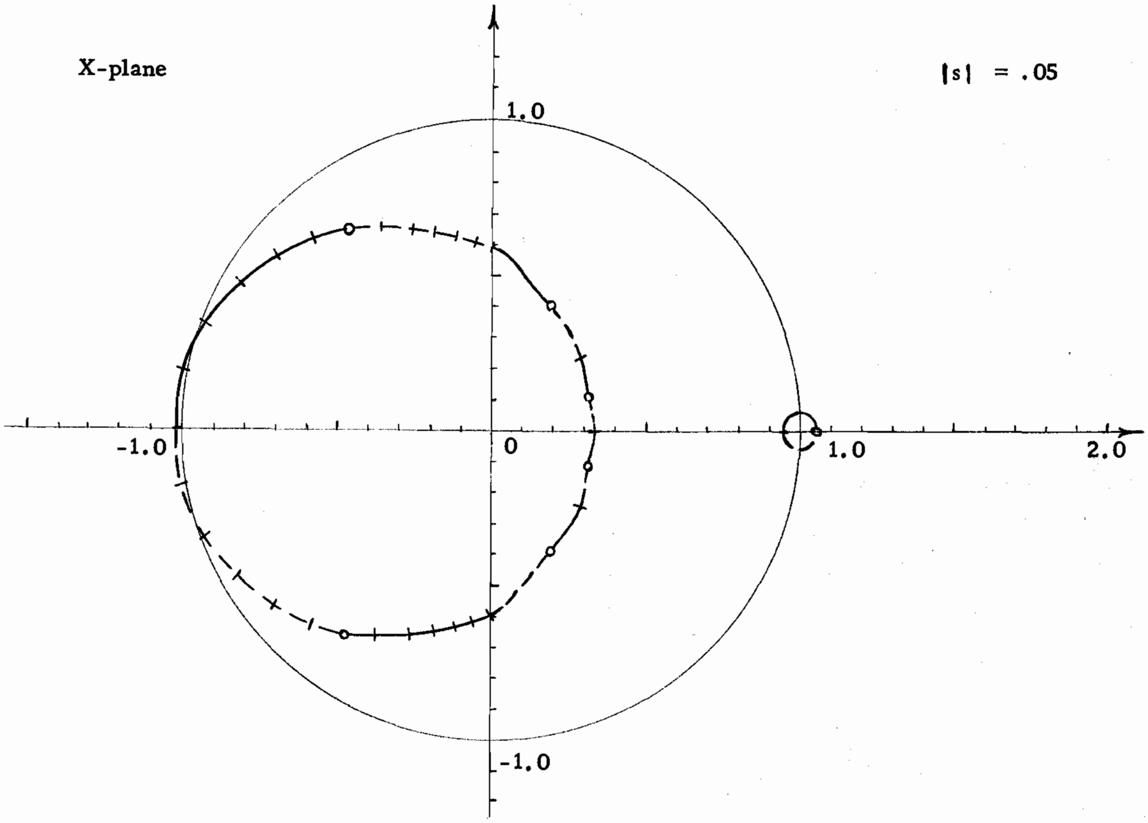


predictor: $$y_{n+1} = y_n + h\left[\frac{16083}{4480}y'_n - \frac{1152169}{120960}y'_{n-1} + \frac{242653}{13440}y'_{n-2} - \frac{296053}{13440}y'_{n-3} + \frac{2102243}{120960}y'_{n-4} - \frac{115747}{13440}y'_{n-5} + \frac{32863}{13440}y'_{n-6} - \frac{5257}{17280}y'_{n-7}\right]$$

corrector: $$y_{n+1} = y_n + h\left[\frac{5257}{17280}y'_{n+1} + \frac{139849}{120960}y'_n - \frac{4511}{4480}y'_{n-1} + \frac{123133}{120960}y'_{n-2} - \frac{88547}{120960}y'_{n-3} + \frac{1537}{4480}y'_{n-4} - \frac{11351}{120960}y'_{n-5} + \frac{275}{24192}y'_{n-6}\right]$$

indicial equation: $$X^8 - (1 + 1.46038\,s + 1.09215\,s^2)X^7 + (1.00692\,s + 2.89780\,s^2)X^6 - (1.01796\,s + 5.49263\,s^2)X^5 + (.732035\,s + 6.70138\,s^2)X^4 - (.343080\,s + 5.28732\,s^2)X^3 + (.093841\,s + 2.62002\,s^2)X^2 - (.011367\,s + .743879\,s^2)X + .092553\,s^2 = 0$$

Figure 7



$$y_{n+1} = y_n + h\left[\frac{198721}{60480}y'_n - \frac{18637}{2520}y'_{n-1} + \frac{235183}{20160}y'_{n-2} - \frac{10754}{945}y'_{n-3}\right.$$

$$\left. + \frac{135713}{20160}y'_{n-4} - \frac{5603}{2520}y'_{n-5} + \frac{19087}{60480}y'_{n-6}\right]$$

indicial equation: $X^7 - (1 + 3.28573\,s)\,X^6 + (7.39563\,s)\,X^5 - (11.6658\,s)\,X^4$

$+ (11.3799\,s)\,X^3 - (6.73180\,s)\,X^2 + (2.223413)\,X - .315592\,s = 0$

Figure 8

TABLE 1

Errors Exhibiting Instability in the 9^{th} Order Method[1]

$|s| = .5$

x	15	15.5	16	16.5	17	17.5	18	18.5	19
$E[\sin x] \times 10^5$	6642	-9490	12264	-16117	19367	-22224	22978	-19033	7966
$E[\cos x] \times 10^5$	-54	-1950	5039	-10113	18499	-30269	48235	-72468	-16004
$E[e^{-x}] \times 10^7$	2415	-1851	-221	2610	-3612	2160	1236	- 4502	5176

$|s| = .4$

x	20	20.4	20.8	21.2	21.6	22	22.4	22.8	23.2	23.6	24.0
$E[\sin x] \times 10^5$	-722	836	-941	973	-1051	902	-784	388	128	-917	2051
$E[\cos x] \times 10^5$	-179	292	-603	867	-1255	1758	-2274	2949	-3539	4301	-4873
$E[e^{-x}] \times 10^9$	60	551	-833	631	- 78	-503	782	-605	93	458	-734

$|s| = .35$

x	35	35.35	35.7	36.05	36.4	36.75	37.10	37.45	37.8	38.15	38.5
$E[\sin x] \times 10^5$	964	-1251	1523	-1789	2068	-2262	2474	-2531	2573	-2419	2168
$E[\cos x] \times 10^5$	-845	780	-579	387	-42	- 324	813	-1374	1995	-2727	3442
$E[e^{-x}] \times 10^9$	- 2	0	0	0	0	- 2	0	0	0	0	-2

$|s| = .25$

x	75	75.25	75.5	75.75	76	76.25	76.5	76.75	77	77.25	77.5
$E[\sin x] \times 10^8$	1944	1570	1095	548	- 35	- 621	-1171	-1653	-2035	-2291	-2408
$E[\cos x] \times 10^8$	-1290	-1736	-2078	-2291	-2364	-2289	-2071	-1723	-1264	- 723	- 134
$E[e^{-x}] \times 10^9$	0	0	0	0	0	0	0	0	0	0	0

In all cases $h = |s|$.

[1] See the beginning of Section I for an explanation of notation used.

TABLE 2

Errors in Solving the System (34)[1]

Table entries should be multiplied by 10^{-9}.

h	.2		.16		.1		.08		.0625		.05		.04	
Method	E[sin 2θ]	E[cos 2θ]	E[sin 2θ]	E[cos 2θ]	E[sin 2θ]	E[cos 2θ]	E[sin 2θ]	E[cos 2θ]	E[sin 2θ]	E[cos 2θ]	E[sin 2θ]	E[cos 2θ]	E[sin 2θ]	E[cos 2θ]
9th M	-115	113	-14	13	-7	14	-7	20	-2	4	-8	8	-8	16
9th	-658	-569	-112	-64	-10	11	-6	17	-5	3	-6	8		
8th M	-687	-533	-107	-63	-21	-5	-12	14	1	1	-4	11		
8th	2506	-4069	265	-864	-93	-63	-6	5	2	2	-5	11		
7th M	2202	-4170	217	-841	-11	-12	-16	21	6	-9	-9	10	-5	17
7th	24978	9869	6297	865	305	-76	64	-17	17	-4	-4	13	-5	17
6th M	24952	7867	6038	500	285	-76	53	-19	30	7	-33	-7	280	-57
6th			1862	47570	1627	3739	669	1120	229	292	76	99	14	37
5th M					1581	3330	624	991	206	264	76	91	16	31
5th					-46612	28972	-17344	13752	-5836	5695	-2206	2503	-844	1088
R-K-G					8039	-14587	3191	-6024	1161	-2274	460	-924	183	-375
H					512	1292	196	394	60	112	9	43	-18	36
H d.p.					509	1284	192	384	49	104	3	26	-23	16
H_1					-15940	10212	-5848	4702	-1953	1921	-741	828	-291	368
H_1 d.p.					-15944	10203	-5852	4695	-1962	1908	-746	819	-306	358
H_2					-8548	17321	-3266	6864	-1171	2496	-467	1006	-201	426
H_2 d.p.					-8548	17316	-3269	6856	-1173	2491	-480	999	-217	414
H_3					-6842	-4102	-2292	-1279	-692	-353	-243	-118		
H_4					-8169	15947	-3222	6467	-1177	2395	-485	971	-219	406

[1] See the beginning of Section I for an explanation of notation used.

TABLE 3

Errors in Solving the System (36) [1]

Table entries should be multiplied by 10^{-9}.

h	.4			.25			.2			.16			.125			.1		
Method	E[sn]	E[cn]	E[dn]	E[sn]	E[cn]	E[dn]	E[sn]	E[cn]	E[dn]	E[sn]	E[cn]	E[dn]	E[sn]	E[cn]	E[dn]	E[sn]	E[cn]	E[dn]
9th M	1366	522	173	13	0	- 3	9	0	1	18	2	0	3	3	2	- 3	0	- 1
9th	2003	1652	1107	89	29	17	35	5	6	19	0	0	5	1	1	0	0	0
8th M	3259	2041	1161	115	33	18	50	0	7	7	2	- 2	6	1	2	5	1	1
8th	- 623	1129	411	- 33	65	51	- 8	14	7	13	5	3	4	2	1	6	2	0
7th M	1650	3031	2168	44	109	70	19	17	14	5	0	0	- 9	2	- 1	1	- 2	2
7th	-10969	- 3513	- 1151	- 996	- 39	- 4	-273	0	3	- 85	- 1	- 2	-36	3	0	-10	1	1
6th M	-12878	- 2542	- 427	-1065	- 8	19	-291	5	7	- 93	1	- 3	-41	- 6	- 6	30	- 1	5
6th	-16496	-14110	-10972	-1264	- 672	- 549	-299	-123	- 110	- 50	- 12	- 16	-26	1	- 2	2	0	0
5th M	-27787	-20785	-12850	-2210	-1078	- 700	-590	-235	- 159	-147	- 41	- 31	1947	- 6	392	5	4	1
5th	-78835	-34576	-29620	16182	-4910	-3449	7171	-1777	-1190	3109	-626	-405	1208	-1911	-121	508	-67	-40
R-K-G	2775	2345	967	4194	224	92	1713	74	31	705	24	10	262	7	3	101	2	1
H.	- 4809	- 5953	- 3424	- 258	- 218	- 124	- 54	- 40	- 23	- 28	- 5	- 6	- 30	- 2	- 5	-31	- 3	- 7
H d. p.	- 4812	- 5952	- 3425	- 265	- 221	- 126	- 61	- 40	- 24	- 22	- 8	- 7	- 27	- 6	- 7	-36	- 9	- 9

[1] See the beginning of Section I for an explanation of notation used.

Example of Computations from Routine for Finding Formulas [1]

```
5200

5 4 - 5 4 3 2 1 0

 1/1
 95/288  1427/1440 -133/240  241/720 -173/1440  3/160
-863/60480    h*7    -0.014269180

 .57      -0.046472556
K=       0.014269134

6 4 - 5 4 3 2 1 0

 1/1
 33/10 -203/45  287/45 -71/15  169/90 -14/45
 1139/3780    h*7     0.301322751

 .50       1.054418326-
K=       0.301327088

2 1 - 3/2 1/2 - 1 0

 1/1
 2/1  -1/1
-23/24 -1/24
-7/5760    h*5     -0.001215278

 .38      -0.018988715
 .60       0.016866560-
K=       0.007459253

2 1 0 - - 2 1 0

 2/1 -1/1
*
 1/12  5/6  1/12
-1/240    h*6     -0.004166667

 .50      -0.011574074
K=       0.004166665
```

[1] See the beginning of Section I for an explanation.

Example of Computations from Routine for Investigating G(s)[1]

5300

6 0 - 6 5 4 3 2 1 0

1
3/10 3/2 3/10 9/5 3/10 3/2 3/10
-1/140 h*7 -0.007142857

.00	0.000000000	.34	-0.022569655	.68	-0.024984098-
.01	0.000000964	.35	-0.020743343	.69	-0.025541629-
.02	0.000000482	.36	-0.018541167	.70	-0.025663785-
.03	0.000000482	.37	-0.016003123	.71	-0.025363094
.04	-0.000000241	.38	-0.013180048	.72	-0.024663891
.05	-0.000002650	.39	-0.010130974	.73	-0.023600389
.06	-0.000012770	.40	-0.006924808	.74	-0.022217162
.07	-0.000037827-	.41	-0.003637687	.75	-0.020567458
.08	-0.000091074-	.42	-0.000352493	.76	-0.018711992
.09	-0.000188654-	.43	0.002843072	.77	-0.016715577
.10	-0.000352252-	.44	0.005860342	.78	-0.014646640
.11	-0.000604755-	.45	0.008608967	.79	-0.012573607
.12	-0.000972186-	.46	0.011003893	.80	-0.010561290
.13	-0.001482734-	.47	0.012964649	.81	-0.008668960
.14	-0.002163385-	.48	0.014422085	.82	-0.006944565
.15	-0.003035100-	.49	0.015321267	.83	-0.005422075
.16	-0.004118117-	.50	0.015625331	.84	-0.004118358
.17	-0.005422075-	.51	0.015321508-	.85	-0.003035341
.18	-0.006944324-	.52	0.014422085-	.86	-0.002163144
.19	-0.008668478-	.53	0.012964890-	.87	-0.001483457
.20	-0.010561049-	.54	0.011003653-	.88	-0.000972909
.21	-0.012573125-	.55	0.008609208-	.89	-0.000604996
.22	-0.014646640-	.56	0.005860101-	.90	-0.000352734
.23	-0.016715095-	.57	0.002843072-	.91	-0.000189377
.24	-0.018711992-	.58	-0.000352734-	.92	-0.000091797
.25	-0.020567217-	.59	-0.003637687-	.93	-0.000038309
.26	-0.022217162-	.60	-0.006924808-	.94	-0.000013011
.27	-0.023600389-	.61	-0.010131215-	.95	-0.000003132
.28	-0.024663650-	.62	-0.013180048-	.96	-0.000000482
.29	-0.025363094-	.63	-0.016003363-	.97	0.000000000
.30	-0.025663544-	.64	-0.018541408-	.98	0.000000000
.31	-0.025541629	.65	-0.020743343-	.99	0.000000000
.32	-0.024984339	.66	-0.022569896-	K=	0.010298426
.33	-0.023989986	.67	-0.023989986-		

The last three digits are significantly affected by round-off error, and should be ignored.

[1] See the beginning of Section I for an explanation.

Example of Computations from Routine for Investigating Stability [1]

```
5400

corrector    5 4 - 5 4 3 2 1 0

1
95/288  1427/1440  -133/240  241/720  -173/1440  3/160

predictor    6 4 - 5 4 3 2 1 0

1
33/10  -203/45  287/45  -71/15  169/90  -14/45

( 1.0000000   +s ( .00000000 ) +s*2 ( .00000000 ) ) X*6
(-1.0000000   +s (-.99097222 ) +s*2 (-1.0885416 ) ) X*5
(-.00000000   +s ( .22430551 ) +s*2 ( 1.4880400 ) ) X*4
(-.00000000   +s (-.33472222 ) +s*2 (-2.1037807 ) ) X*3
(-.00000000   +s ( .12013888 ) +s*2 ( 1.5613425 ) ) X*2
(-.00000000   +s (-.01875000 ) +s*2 (-.61940581 ) ) X*1
(-.00000000   +s ( .00000000 ) +s*2 ( .10262345 ) )

|s|= .5     arg s = 0     Δ = 15
guesses
1.65 0    .2 .2   .2 -.2   .3 0
-.4 .6    -.4 -.6

|s|=   .50000000     arg s = .00000000
    1.6486354   .00000000     1.6486354   .00000000
    .21061887   .27433518     .34586138   .00000000
    .21061888  -.27433521     .34586141   .00000000
    .28235471   .00000000     .28235471   .00000000
   -.29230315   .61263597     .67879593   .00000000
   -.29230315  - 61263597     .67879593   .00000000

|s|=   .50000000     arg s =15.000000
    1.6073531   .20908877     1.6208954   .00000000
    .19820721   .29385316     .35445136   .00000000
    .22293998  -.25671402     .34000635   .00000000
    .28286108   .01482852     .28324950   .00000000
   -.38253966   .61025691     .72024304   .00000000
   -.21454290  -.60700446     .64380360   .00000000
```

[1] See the beginning of Section I for an explanation.

BIBLIOGRAPHY

1. Ahlfors, Lars V. Complex analysis. New York, McGraw-Hill, 1953. 247 p.

2. Bragg, L. R. and E. B. Leach. The remainder terms in numerical integration formulas. The American Mathematical Monthly 70:70-76. 1963.

3. Brock, P. and F. J. Murray. The use of exponential sums in step-by-step integration. Mathematical Tables and Other Aids to Computation 6:63-78. 1952.

4. Chase, P. E. Stability properties of predictor-corrector methods for ordinary differential equations. Journal of the Association for Computing Machinery 7:46-56. 1960.

5. Crane, Roger L. and Robert J. Lambert. Stability of a generalized corrector formula. Journal of the Association for Computing Machinery 9:104-117. 1962.

6. Dahlquist, Germund. Convergence and stability in the numerical integration of ordinary differential equations. Mathematica Scandinavica 4:33-53. 1956.

7. Daniell, P. J. Remainders in interpolation and quadrature formulae. The Mathematical Gazette 24:238-244. 1940.

8. Dennis, S. C. R. Step-by-step integration of ordinary differential equations. Quarterly of Applied Mathematics 20:359-372. 1962-63.

9. Gautschi, Walter. Numerical integration of ordinary differential equations based on trigonometric polynomials. Numerische Mathematik 3:381-397. 1961.

10. Gill, S. A process for the step-by-step integration of differential equations in an automatic digital computing machine. Proceedings of the Cambridge Philosophical Society 47:96-108. 1951.

11. Hamming, R. W. Stable predictor-corrector methods for ordinary differential equations. Journal of the Association for Computing Machinery 6:37-47. 1959.

12. Hamming, R. W. Numerical methods for scientists and engineers. New York, McGraw-Hill, 1962. 411 p.

13. Henrici, Peter. Error propagation for difference methods. New York, Wiley, 1963. 73 p.

14. Hildebrand, F. B. Introduction to numerical analysis. New York, McGraw-Hill, 1956. 511 p.

15. Hull, T. E. and A L. Creemer. Efficiency of predictor-corrector procedures. Journal of the Association for Computing Machinery 10:291-301. 1963.

16. Hull, T. E. and A. C. R. Newbery. Integration procedures which minimize propagated errors. Journal of the Society for Industrial and Applied Mathematics 9:31-47. 1961.

17. Hull, T. E. and A. C. R. Newbery. Corrector formulas for multi-step integration methods. Journal of the Society for Industrial and Applied Mathematics 10:351-369. 1962.

18. Kunz, Kaiser S. Numerical analysis. New York, McGraw-Hill, 1957, 381 p.

19. Milne, W. E Numerical calculus. Princeton, Princeton University Press, 1949. 393 p.

20. Milne, W. E. The remainder in linear methods of approximation. Journal of Research of the National Bureau of Standards 43:501-511. 1949.

21 Milne, W E. Numerical solution of differential equations. New York, Wiley, 1953. 275 p.

22. Milne, W. E., and R. R. Reynolds. Stability of a numerical solution of differential equations--part II. Journal of the Association for Computing Machinery 7:46-56. 1960.

23. Nordsieck, Arnold. On numerical integration of ordinary differential equations. Mathematics of Computation 16:22-49. 1962.

24. Pope, David A. An exponential method of numerical integration of ordinary differential equations. Communications of the ACM 6:491-493. 1963.

25. Ralston, Anthony and Herbert S. Wilf eds. Mathematical methods for digital computers. New York, Wiley, 1960. 293 p.

26. Steffensen, J. F. Interpolation. 2d ed. New York, Chelsea, 1950. 248 p.