5-1968

Tests of Methods that Control Round-Off Error

Dale M. Rasmuson
Utah State University

TESTS OF METHODS THAT CONTROL
ROUND-OFF ERROR

by

Dale M. Rasmuson

A thesis submitted in partial fulfillment
of the requirements for the degree

of

MASTER OF SCIENCE

in

Mathematics

Approved:

UTAH STATE UNIVERSITY
Logan, Utah

1968

ACKNOWLEDGMENTS

I wish to express my appreciation to Dr. Richard A. Hansen for the many hours of help and guidance he has given me in the preparation of this paper.

I would like to thank Dr. James D. Watson for his help and suggestions.

Special thanks are extended to my wife, Linda, for her help, patience, and support.

Dale M. Rasmuson

TABLE OF CONTENTS

Page

ACKNOWLEDGMENTS ii

LIST OF FIGURES iv

ABSTRACT v

INTRODUCTION 1

THEORETICAL CUMULATIVE ROUNDING 8

ROUND-OFF ERROR LIMITS 15

THEORETICAL PARTIAL CUMULATIVE ROUNDING 18

IMPLEMENTATION AND EXAMPLES 20

REFERENCES 23

APPENDIX 24

VITA 29

LIST OF FIGURES

Figure Page

1. Error at $x = 0.25$ in solution of $y' = y^2$, $y(0) = 1.0$ . . 22

ABSTRACT

Tests of Methods that Control Round-Off Error

by

Dale M. Rasmuson, Master of Science

Utah State University, 1968

Thesis Director: Dr. Richard A. Hansen
Major Professor: Dr. James D. Watson
Department: Mathematics

Methods of controlling round-off error in one-step methods in the numerical solution of ordinary differential equations are compared. A new Algorithm called theoretical cumulative rounding is formulated. Round-off error bounds are obtained for single precision, and theoretical cumulative rounding. Limits of these bounds are obtained as the step length approaches zero. It is shown that the limit of the bound on the round-off error is unbounded for single precision and double precision, is constant for theoretical partial double precision, and is zero for theoretical cumulative rounding.

The limits of round-off bounds are not obtainable in actual practice. The round-off error increases for single precision, remains about constant for partial double precision and decreases for cumulative rounding as the step length decreases. Several examples are included.

(34 pages)

INTRODUCTION

Consider the initial value problem

$$y' = f\ (x,y) \tag{0.1}$$

$$y(a) = c$$

where f (x,y) satisfies conditions guaranteeing a unique solution on the interval $[a, b]$, [3, pp. 15-25]. In this paper we will be concerned with methods for approximating solutions of (0.1) that are based on the principle of discretization. These methods make no attempt to approximate the <u>exact solution</u> y (x) of (0.1) over the continuous range $[a, b]$ of the independent variable x, but approximate values are sought only on a discrete set of points $\{x_0, x_1, x_2, \ldots\}$ that are contained in the interval $[a, b]$. We shall be concerned only with the set of equidistant points $x_n = a + nh$ $(n = 0, 1, 2, \ldots)$ where h is some predetermined constant referred to as the <u>step length</u> and the points x_n are called <u>lattice points</u>.

In general, a discrete variable method for solving (0.1) consists of an algorithm which, corresponding to each point x_n, gives a number y_n which is to be regarded as an approximation to the value y (x_n), the exact solution at x_n.

The theoretical algorithms for one-step methods can be represented in general by the difference equation

$$y_0 = c \tag{0.2}$$

$$y_{n+1} = y_n + h\Phi\ (x_n, y_n; h).$$

The function Φ $(x_n, y_n; h)$ is called the increment function, and the product $h\Phi$ $(x_n, y_n; h)$ is called the increment. The sequence $\{y_n\}$ which is the solution of (0.2) is called the theoretical approximate solution of (0.1). We will assume that Φ $(x_n, y_n; h)$ is Lipschitz in the variable y with Lipschitz constant L.

There are two sources of error in solving a differential equation by a numerical method. First, the number y_n calculated from the algorithm (0.2) will rarely agree with the corresponding value of the true solution y (x_n). The difference, e_n, where

$$e_n = y\ (x_n) - y_n,$$

is called the theoretical error.

Algorithm (0.2) is said to be convergent if for any arbitrary initial value c and an arbitrary x in $[a, b]$, we have

$$\lim_{\substack{h \to 0 \\ x_n = x}} y_n = y(x),$$

i.e., the theoretical error e_n at x vanishes as the steplength h approaches zero. In this paper we will assume that algorithm (0.2) is convergent.

In most applications y_n cannot be calculated with unlimited precision because of the limited capacity of the computing machinery. Therefore, numerical algorithms which contain a sequence of arithmetic operations prescribed by (0.2) will fall within the limits of the computing machinery. We shall denote by $\tilde{y}_n$ the value that is actually computed in place of y_n. The difference, r_n, where

$$r_n = y_n - \tilde{y}_n,$$

will be called the round-off error, and the sequence $\{\tilde{y}_n\}$ is called the numerical approximate solution of (0.2).

The numerical approximate solution $\tilde{y}_n$ satisfies the difference equation

$$\tilde{y}_{n+1} = \tilde{y}_n + \left[h\Phi\ (x_n, \tilde{y}_n; h)\right]_S \tag{0.3}$$

where $\left[h\Phi\ (x_n, \tilde{y}_n; h)\right]_S$ is the evaluation of the n-th increment $h\Phi\ (x_n, y_n; h)$ using a sequence S of arithmetic and round-off operations. A final addition of the increment to $\tilde{y}_n$ is required to obtain $\tilde{y}_{n+1}$.

In the evaluation of the increment on the n-th iteration there are a certain number of intermediate values generated that require rounding. We shall refer to these values as round-off variables and denote them by $P_{j,n}$ ($j = M, M-1, \ldots, 0$). After a round-off operation has occurred, these variables are denoted by $P^*_{j,n}$.

In order to clarify the notation needed for the above mentioned round-off variables, we will consider three particular examples. First, we consider the increment function $\Phi\ (x, y; h) = y^2$ and the sequence S:

i) squaring $\tilde{y}_n$,

ii) truncating the square to single precision,

iii) multiplying the truncated value by h, and

iv) truncating this product to single precision.

The round-off variables are $P_{1,n} = (\tilde{y}_n)^2$ $P_{0,n} = h\ P^*_{1,n}$. This completes the evaluation of the increment. The additional operation of adding the increment to $\tilde{y}_n$ may result in a rounding error due to a shift operation. This error can be included in the rounding of $P_{0,n}$.

Secondly, we again consider the above increment function. We require $\tilde{y}_n$ to be a double precision variable and define S by:

i) truncating $\tilde{y}_n$ to single precision,

ii) squaring the truncated value $\tilde{y}^*_n$,

iii) truncating the squared value to single precision, and

iv) multiplying this truncated product by h.

The round-off variables are $P_{2,n} = \tilde{y}_n$, $P_{1,n} = (\tilde{y}^*_n)^2$, and $P_{0,n} = h\, P^*_{1,n}$. We do not truncate $P_{0,n}$ since $\tilde{y}_n$ is double precision, and we want the addition $\tilde{y}_n + P_{0,n}$ to be a double precision variable. This technique is called partial double precision. Henrici [3, p. 94] defines

> . . . the algorithm of partial double precision . . . by
>
> $$\tilde{y}_0 = y_0$$
>
> $$\tilde{y}_{n+1} = \tilde{y}_n + h\tilde{\Phi}\ (x_n,\ \tilde{y}^*_n;\ h),\ n = 0,\ 1,\ 2,\ .\ .\ .$$
>
> Here the $\tilde{y}_n$ are double precision numbers. The two important features of the algorithm are:
>
> i) the product $h\tilde{\Phi}$ is left unrounded, and is in its entirety added to $\tilde{y}_n$;
>
> ii) the function $\tilde{\Phi}$ is evaluated with the more significant portion of $\tilde{y}_n$ only. Thus the time required for computing $\tilde{\Phi}$ is not increased in comparison with ordinary single precision operation.

For our third example consider the increment function $\Phi\ (x,\ y;\ h) = xy$ and define the sequence S as:

i) multiplying x_n by $\tilde{y}_n$

ii) truncating the product to single precision,

iii) multiplying the truncated product by h, and

iv) truncating the final product to single precision.

The round-off variables are $P_{1,n} = x_n\, \tilde{y}_n$ and $P_{0,n} = h\, P^*_{1,n}$.

The successive approximations of the increment function $h\Phi\ (x_n,\ \tilde{y}_n;\ h)$ given by the sequence S will be denoted by

$$h\Phi\ (x_n,\ \tilde{y}_n;\ h) = \theta_M\ (x_n,\ \tilde{y}_n,\ P_{M,n};\ h),$$

$$\theta_M\ (x_n,\ \tilde{y}_n,\ P^*_{M,n};\ h) = \theta_{M-1}\ (x_n,\ \tilde{y}_n,\ P_{M-1,n};\ h),$$

$$\circ\ \circ\ \circ$$

$$\theta_1\ (x_n,\ \tilde{y}_n,\ P^*_{1,n};\ h) = \theta_0\ (x_n,\ \tilde{y}_n,\ P_{0,n};\ h),$$

$$\theta_0\ (x_n,\ \tilde{y}_n,\ P^*_{0,n};\ h) = h\phi\left[(x_n,\ y_n;\ h)\right]_S.$$

For the first example above the approximations are

$$h\Phi\ (x_n,\ \tilde{y}_n;\ h) = h\ (\tilde{y}_n)^2 = h\ P_{1,n} = \theta_1\ (x_n,\ \tilde{y}_n,\ P_{1,n};\ h),$$

$$\theta_1\ (x_n,\ \tilde{y}_n,\ P^*_{1,n};\ h) = h\ P^*_{1,n} = P_{0,n} = \theta_0\ (x_n,\ \tilde{y}_n,\ P_{0,n};\ h),$$

$$\theta_0\ (x_n,\ \tilde{y}_n,\ P^*_{0,n};\ h) = P^*_{0,n} = \left[h\ (\tilde{y}_n)^2\right]_S = \left[h\Phi\ (x_n,\ \tilde{y}_n;\ h\right]_S.$$

Thus, we can see that

$$\left[h\Phi\ (x_n,\ \tilde{y}_n;\ h)\right]_S = h\Phi\ (x_n,\ y_n;\ h) - \sum_{m=0}^{M} \delta_{m,n}$$

where

(0.4) $$\delta_{m,n} = \theta_m\ (x_n,\ y_n,\ P^*_{M,n};\ h) - \theta_m\ (x_n,\ y_n,\ P_{M,n};h).$$

The variable $\delta_{m,n}$ is the round-off error made in the n-th evaluation of the increment function due to the rounding off of the round-off variable $P_{M,n}$. The rounding error $\delta_{o,n}$ will also include an error resulting from a shift operation.

If we subtract equation (0.3) from equation (0.2), we can see that the cumulative round-off error r_n satisfies the difference equation

$$(0.5) \qquad r_0 = 0$$

$$r_{n+1} = (1 + h\,K_n)\,r_n + \sum_{m=0}^{M} \delta_{m,n}$$

where

$$K_n = \left[\Phi\,(x_n, \tilde{y}_n; h) - \Phi\,(x_n, y_n; h)\right]/r_n \text{ if } r_n \neq 0$$

and

$$K_n = 0 \text{ if } r_n = 0.$$

Theorem A of the Appendix shows that the solution of the difference equation (0.5) can be written as

$$(0.6) \qquad r_n = \sum_{i=0}^{n-1} \sum_{m=0}^{M} \delta_{m,i} + \sum_{j=1}^{n-1} h\,K_j \prod_{q=j+1}^{n-1} (1 + h\,K_q) \sum_{i=0}^{j-1} \sum_{m=0}^{M} \delta_{m,i}$$

and that the inequality

$$(0.7) \qquad 1 + \sum_{j=1}^{n-1} \left| h\,K_j \prod_{q=j+1}^{n-1} (1 + h\,K_q) \right| \leq \prod_{j=1}^{n-1} (1 + h\,|K_j|)$$

is true for all $n \geq 1$.

If we overestimate the absolute value of the sums

$$\sum_{i=0}^{j-1} \sum_{m=0}^{M} \delta_{m,i} \quad (j = 1, \ldots, n)$$

with

(0.8) $$B_n = \max_j \sum_{i=0}^{j-1} \sum_{m=o}^{M} \delta_{m,i} \quad (j = 1, 2, . . . n)$$

and use (0.7), we see that $|r_n|$ is bounded, i.e.,

(0.9) $$|r_n| \leq B_n \prod_{j=1}^{n-1} (1 + h\, |K_j|).$$

Since the Lipschitz constant L of Φ (x,y;h) bounds $|K_j|$ [2, p. 71], i.e.,

$$|K_j| \leq L \; (j = 1, 2, . . . ,n),$$

we have

(0.10) $$|r_n| \leq B_n \,(1 + h\, L)^{n-1}.$$

THEORETICAL CUMULATIVE ROUNDING

In this section we will assume that the sequence S is defined so that an approximation of the increment function is obtained first and then multiplied by h to obtain the increment. We will denote the approximations of the n-th evaluation of the increment function by

$$\Phi\ (x_n, \tilde{y}_n; h) = \phi_M\ (x_n, \tilde{y}_n, P_{M,n}; h)$$

$$\phi\ (x_n, \tilde{y}_n, P^*_{M,n}; H) = \phi_{M-1}\ (x_n, \tilde{y}_n, P_{M-1,n}; h)$$

$$\circ \quad \bullet \quad \circ$$

$$\phi_2\ (x_n, \tilde{y}_n, P^*_{2,n}; h) = P_{1,n}, \text{ and}$$

$$P^*_{1,n} = \left[\Phi\ (x_n, \tilde{y}_n; h)\right]_S .$$

Thus,

$$h\ P^*_{1,n} = h\ \left[\Phi\ (x_n, \tilde{y}_n; h)\right]_S = P_{0,n} .$$

By defining $\alpha_{m,n}$ as

$$\alpha_{m,n} = \phi_m\ (x_n, \tilde{y}_n, P^*_{m,n}; h) - \phi_m\ (x_n, \tilde{y}_n, P^*_{m,n}; h)$$

$$(m = 1, \ldots, M),$$

we see that $\delta_{m,n}$ defined by (0.4) can be expressed

$$\delta_{m,n} = h\, \alpha_{m,n} \quad (m = 1, \ldots, M).$$

Hence, B_n of (0.8) can be expressed as

$$(1.1) \qquad B_n = \max_j \left| \sum_{i=1}^{j-1} \delta_{0,i} + h \sum_{i=1}^{j-1} \sum_{m=1}^{M} \alpha_{m,i} \right| \quad (j = 1, \ldots, n).$$

With B_n expressed in this way we can see that for sufficiently small h that the greatest contribution to the bound on r_n given in (0.10) will normally be a result of the rounding of the products h $P^*_{1,i}$ (i = 0, 1, . . . ,n-1). Thus, we see that if no round-off error occurs in the multiplication by h and the addition of this product to y_n, i.e., $\delta_{0,i}$ is zero (i = 0, 1, ... , n-1), then the major portion of the round-off error bound can be eliminated. The above results of $\delta_{0,i} = 0$ (i = 0, 1, ... , n-1) are normally not obtainable in a practical application; however, if the method of partial double precision is utilized (see example 2 of the Introduction), then the $\delta_{0,i}$ are approximately zero. The above technique of assuming $\delta_{0,i} = 0$ will be called <u>theoretical partial double precision</u>. This technique is performed by requiring y_n to be the (M + 1)st round-off variable and then continuing the sequence S as above. Thus B_n of (1.1) will be given by

$$(1.2) \qquad B_n = h \max_j \left| \sum_{i=0}^{j-1} \sum_{m=1}^{M+1} \alpha_{m,i} \right| \quad (j = 1, \ldots, n).$$

If we interchange the finite sums in (1.2) and let

$$A_{m,n} = \max_j \left| \sum_{i=0}^{j-1} \alpha_{m,i} \right| \quad (j = 1, \ldots, n),$$

we see from (1.2) that B_n is bounded, i.e.,

(1.3) $$B_n \leq h \sum_{m=1}^{M+1} A_{m,n}.$$

Now let us define some new terms. Let

$$R_{m,n} = P^*_{m,n} - P_{m,n},$$

$$V_{m,n} = \alpha_{m,n}/R_{m,n} \text{ if } R_{m,n} \neq 0,$$

and

$$V_{m,n} = 0 \text{ if } R_{m,n} \neq 0.$$

In the second example of the Introduction, we would have $V_{1,n} = \tilde{y}^*_n + \tilde{y}_n$ since $P_{1,n} = \tilde{y}_n$ and $V_{0,n} = 1$ because $P_{0,n} = (y^*_n)^2$.

In our application we are concerned with a fixed word length computer. Let ℓ denote the number of bits in the single precision word. We define $\mu_{m,n} = 1 \times b^{-\ell+e_{m,n}}$ where b is the base of the computer, and $e_{m,n}$ is the exponent of the variable $P_{m,n}$. For theoretical purposes we will assume that the computer is capable of handling an exponent as large or as small as we please.

We now formulate a round-off procedure called theoretical cumulative rounding.

Algorithm 1. Add the double word length variable $P_{m,n}$ to the double word length variable $\sum_{i=0}^{n-1} R_{m,i}$. Then truncate the sum to obtain the single word length variable $P^*_{m,n}$. Calculate the accumulated round-off error $\sum_{i=0}^{n} R_{m,i}$ by adding $P_{m,n} = P^*_{m,n}$ to $\sum_{i=0}^{n-1} R_{m,i}$.

In order to indicate the numerical results of the above algorithm, it will be necessary to specify the single precision word length and base of the machine. For example, let four decimal digits represent the single precision word length; the base is 10. Further, suppose

$$P_{m,n} = 2.315\ 4553$$

and

$$\sum_{i=0}^{n-1} R_{m,i} = 0.000\ 4892.$$

Then cumulative rounding can be performed by adding the double precision $P_{m,n}$ to the double precision $\sum_{i=0}^{n-1} R_{m,i}$ yielding 2.315 9445. The rounded value of $P_{m,n}$ is $P^*_{m,n} = 2.315$ if the machine truncates and $P^*_{m,n} = 2.316$ if the machine symmetric rounds in converting from double precision to single precision variable. Continuing, we compute

$$\sum_{i=0}^{n} R_{m,i} = 0.000\ 9445$$

if truncation has occurred and

$$\sum_{i=0}^{n} R_{m,i} = -0.000\ 0555$$

if symmetric rounding has occurred. For this case we have

$$\mu_{m,n} = 0.001\ 0000.$$

The value of $\mu_{m,n}$ may change as the calculation continues. Suppose

$$P_{m,n} = .9862\ 3241$$

and

$$\sum_{i=0}^{n-1} R_{m,i} = 0.000\ 3244$$

then

$$\mu_{m,n} = .0001\ 0000,$$

$$P^*_{m,n} = .9865,$$

and

$$\sum_{i=0}^{n} R_{m,i} = .0000\ 5681.$$

It is easy to see that

$$\left|\sum_{i=0}^{n} R_{m,i}\right| \leq \mu_{m,n} \quad (i = 1, 2, \ldots, n).$$

The use of theoretical cumulative rounding, as the following theorem indicates, will control the growth of each $A_{m,n}$ $(m = \Phi, \ldots, M+1)$; thus the growth of B_n.

<u>Theorem 1.</u> If there exists $\tau_{m,j}$ such that $V_{m,j} = V_{m,j+1} + h\ \tau_{m,j}$ $(j = 0, \ldots, n-1)$ and if the variables $P_{m,j}$ $(j = 0, 1, \ldots, n-1)$ are rounded using Algorithm 1, then

$$A_{m,n} \leq (N_{m,n} + (n - 1)\ h\ T_{m,n})\ \mu_m$$

where

$$N_{m,n} = \max_j |V_{m,n}|, \; T_{m,n} = \max_j |\tau_{m,j}|$$

and

$$\mu_m = \max_j \mu_{m,j} \; (j = 0, 1, \ldots, n).$$

Proof: It is obvious that

$$(1.4) \qquad \left|\sum_{i=0}^{j-1} R_{m,i}\right| \leq \mu_m \quad (j = 1, \ldots, n).$$

Since

$$(1.5) \qquad \sum_{i=0}^{j-1} V_{m,i} R_{m,i} = V_{m,j} \left(\sum_{i=0}^{j-1} R_{m,i}\right) + h \sum_{q=0}^{j-1} \tau_{m,q} \sum_{i=0}^{q} R_{m,i}$$

(see Lemma A of the Appendix), we see that

$$\left|\sum_{i=0}^{j-1} \alpha_{m,i}\right| \leq |V_{m,j}| \left|\sum_{i=0}^{j-1} R_{m,i}\right| + h \sum_{q=0}^{j-1} |\tau_{m,q}| \left|\sum_{i=0}^{q} R_{m,i}\right|.$$

The utilization of the inequality (1.4) yields

$$\left|\sum_{i=0}^{j-1} \alpha_{m,i}\right| \leq \left(|V_{m,j}| + h \sum_{q=0}^{j-1} |\tau_{m,q}|\right) \mu_m.$$

An overestimation of $V_{m,j}$ and $\tau_{m,j}$ $(j = 0, 1, \ldots, n)$ yields

$$(1.6) \qquad A_{m,n} \leq (N_{m,n} + (n - 1) h T_{m,n}) \mu_m.$$

Thus the theorem is proved.

From (1.3) and (1.6) we can see that

$$(1.7) \qquad B_n \leq h U_n$$

where

$$U_n = \sum_{m=1}^{M+1} (N_{m,n} + (n - 1) h T_{m,n}) \mu_m. \tag{1.8}$$

Now U_n is a finite number since $N_{m,n}$, $T_{m,n}$, μ_m, are all finite (m = 1, . . . , M+1). Thus, from (0.7) and (1.7) we have

$$|r_n| \leq h U_n (1 + h L)^{n-1}, \tag{1.9}$$

i.e., r_n is bounded by a term of order 0 (h).

ROUND-OFF ERROR LIMITS

In this section we want to investigate the limit of round-off error bounds as h approaches zero for the cases single precision, double precision, theoretical partial double precision, and theoretical cumulative rounding.

It is difficult, in general, to get an exact form for the error r_n; therefore, we will investigate the limit of the error bounds (0.10) and (1.9) as h approaches zero. Since $x = a + nh$, where n is a positive integer, requires h divide $x - a$, we will restrict ourselves to the set $H = \{h \mid h \text{ divides } x - a\}$.

Let us define

$$r\,(x,h) = r_n,$$

$$B\,(x,h) = B_n,$$

and

$$U\,(x,h) = U_n \text{ where } n = (x - a)/h.$$

Let us first consider the case of single precision. The round-off error bound for this case is given by equation (0.10), or in the notation of this section,

$$|r\,(x,h)| \leq B\,(x,h)\,(1 + h\,L)^{n-1} \tag{2.1}$$

and

$$B\,(x,h) = \max_j \left| \sum_{i=0}^{j-1} \sum_{m=0}^{M} \delta_{m,i} \right| \quad (j = 1, \ldots, n)$$

where n = (x - a)/h.

If the sum $\sum_{m=0}^{M} \delta_{m,i}$ is bounded by some constant k, we have

$$(2.3) \qquad B\ (x,h) \leq \max_{j} \left| \sum_{i=0}^{j-1} k \right| = n\ k.$$

Thus we see that

$$(2.4) \qquad |r\ (x,h)| \leq n\ k \cdot (1 + h\ L)^{n-1}.$$

Replacing n by (x - a)/h in (2.4) and taking the limit as h approaches zero, we have

$$\lim_{h \to 0} |r\ (x,h)| \leq \lim_{h \to 0} \frac{(x - a)}{h} k\ (1 + h\ L)^{((x - a)/h)-1}$$

$$\leq k\ e^{L(x - a)} \lim_{h \to 0} (x - a)/h$$

Thus, the bound on the round-off error for single precision becomes unbounded as h approaches zero. We should note that the bound on the round-off error for double precision or higher order precisions also becomes unbounded as h approaches zero.

The bound on the round-off error r (x,h) for theoretical partial double precision is given in (2.1) where

$$B\ (x,h) = h \max_{j} \left| \sum_{i=0}^{j-1} \sum_{m=1}^{M+1} \alpha_{m,i} \right| \quad (j = 1, 2, \ldots, n).$$

If the sum

$$\sum_{m=1}^{M+1} \alpha_{m,i}$$

is bounded by a constant C, it follows that

$$(2.5) \qquad |r\ (x,h)| \leq hn\ C\ (1 + h\ L)^{n-1}.$$

Replacing n with (x - a)/h in (2.5), we have

$$\lim_{h\to 0} |r\ (x,h)| \leq \lim_{h\to 0} \frac{h\ (x - a)}{h}\ C\ (1 + h\ L)^{((x - a)/h)-1}$$

$$\leq C\ (x - a) \lim_{h\to 0} (1 + h\ L)^{((x - a)/h)-1}$$

$$\leq (x - a)\ C\ e^{(x - a)L}.$$

Hence, the round-off error is bounded as h approaches zero for theoretical partial double precision.

The bound for the round-off error in theoretical cumulative rounding is given by (1.9), i.e., $|r\ (x,h)| \leq h\ U\ (x,h)(1 + h\ L)^{(n-1)}$ where n = (x - a)/h. If the hypotheses of Theorem 1 are satisfied, then the variable U (x,h) is bounded, i.e., U (x,h) $\leq$ D. We then have

$$(2.6) \qquad |r\ (x,h)| \leq h\ D\ (1 + h\ L)^{n-1}.$$

Taking the limit as h approaches zero on both sides of (2.6), we have

$$\lim_{h\to 0} |r\ (x,h)| = 0.$$

Therefore, in theoretical cumulative rounding, we see that the round-off error vanishes as h approaches zero. It should be mentioned that the above results are not obtainable in actual practice.

THEORETICAL PARTIAL CUMULATIVE ROUNDING

Let Φ (x, y; h) be an increment function which can be written in the form

$$(3.1) \qquad \Phi(x, y; h) = \Phi_1(x, y; h) + h\Phi_2(x, y; h);$$

and let the sequence S be redefined in such a way that arithmetic operations in the evaluation of $\Phi_2(x_n, \tilde{y}_n; h)$ uses single precision, and the evaluation of $\Phi_1(x_n, \tilde{y}_n; h)$ uses theoretical cumulative rounding. Let β_n denote the accumulated round-off error encountered in the n-th evaluation of $\Phi_2(x_n, \tilde{y}_n; h)$ and γ_n the accumulated round-off error in the n-th evaluation of $\Phi_1(x_n, \tilde{y}_n; h)$.

From the equation (0.10) we see that

$$|\beta_n| \leq B_n (1 + h L)^{n-1},$$

and from the equation (1.9) we have

$$|\gamma_n| \leq h U_n (1 + h L)^{n-1}$$

where U_n is expressed by (1.7).

We can now see that

$$|r_n| \leq h U_n (1 + h L)^{n-1} + h^2 B_n (1 + h L)^{n-1}$$

or

$$|r_n| \leq h (U_n + h B_n) (1 + h L)^{n-1}.$$

If B_n is bounded by G, we have

$$|r_n| \leq h\ (U_n + h\ G)\ (1 + h\ L)^{n-1}.$$

Using the fact that $n = (x - a)/h$ and expressing r_n as $r\ (x,h)$ and U_n as $U\ (x,h)$, we have

$$\lim_{h\to 0} |r\ (x,h)| \leq \lim_{h\to 0} h\ (U\ (x,h) + h\ G)\ (1 + h\ L)^{((x-a)/h)-1}$$

is equal to zero.

Let us summarize the above as follows:

Algorithm 2. (Theoretical Partial Cumulative Rounding). If the increment function $\Phi\ (x,\ y;\ h)$ can be written in the form (3.1), we can perform all arithmetic operations in the evaluation of $\Phi_2\ (x,\ y;\ h)$ in single precision and then use cumulative rounding (Algorithm 1) in the remainder of the evaluations.

IMPLEMENTATION AND EXAMPLES

Partial double precision is performed by evaluating the increment function in single precision, i.e., $\tilde{y}_n$ is the only double precision variable. There is no way to guarantee that the round-off error associated with the above addition of the increment to $\tilde{y}_n$ is zero; only that it is nearly zero. Therefore, the remarks about round-off error limits should be modified by saying that the limit of the round-off error is nearly constant as the step length approaches some $H > 0$ where h depends upon (0.1), (0.2), and the individual computer [1, p. 249].

Implementation of the cumulative rounding is easy. We want to perform all additions and subtractions in double precision and all multiplications and divisions in single precision. Moreover, we want to round off using Algorithm 1. This is best illustrated by an example. Consider the initial value problem $y' = y^2$, $y\ (0) = 1$ using Euler's method. A typical FORTRAN program follows:

```
    DOUBLE PRECISION YN,YS,YTT,YSTT,HH,R1,R2
  1 READ 20,H,K
  2 HH=H
    X=0.0
    YN=1.DO
    R1=0.DO
    R2=0.DO
    DO 10 I=1,K
    X=X+H
    YT=YN+R2
    R2=(YN+R2)-YT
  3 YTT=YT
  4 YS=YTT*YTT
    YST=YS+R1
    R1=(YS+R1)-YST
  5 YN=YN+HH*YST
    PRINT 25,I,X,YN
 10 CONTINUE
    GO TO 1
 20 FORMAT (F15.8,I10)
 25 FORMAT (5X,I5,5X,F15.8,5X,D25.16)
    END
```

The above program should work on most computers that have the FORTRAN IV option. If a computer handles a statement similar to Y=YN+X*Z, where Y and YN are double precision variables, and X and Z are single precision, by converting X and Z to double precision <u>before</u> performing the multiplication as a standard machine operation; then in the above program statements 2 and 3 may be omitted, and statements 4 and 5 changed to

```
4 YS=YT*YT
5 YN=YN+H*YST.
```

The remarks about round-off error limits should also be modified for cumulative rounding. Due to the limited capacity of the computer, the step length h will have a lower bound. Therefore, we cannot consider the limit of the round-off error as h approaches zero, but the limit as h approaches some constant $H > 0$ where H depends upon (0.1), (0.2), and the individual computer. Hence, the limit of the round-off error will be approximately zero as the step length approaches H.

Several problems have been run on the IBM 7040 computer using step sizes $h = 2^{-5}, 2^{-6}, \ldots, 2^{-16}$ using double precision, partial double precision, and partial cumulative rounding. In each case the error was calculated by the relationship $(y(x_n) - y_n)/y(x_n)$. The same problems were also run on an IBM 360/44 and an UNIVAC 1108 computer with comparable results. The results in Figure 1 are typical.

In conclusion, we may say that Figure 1 represents the general curve of the errors for double precision, partial double precision, and partial cumulative rounding.

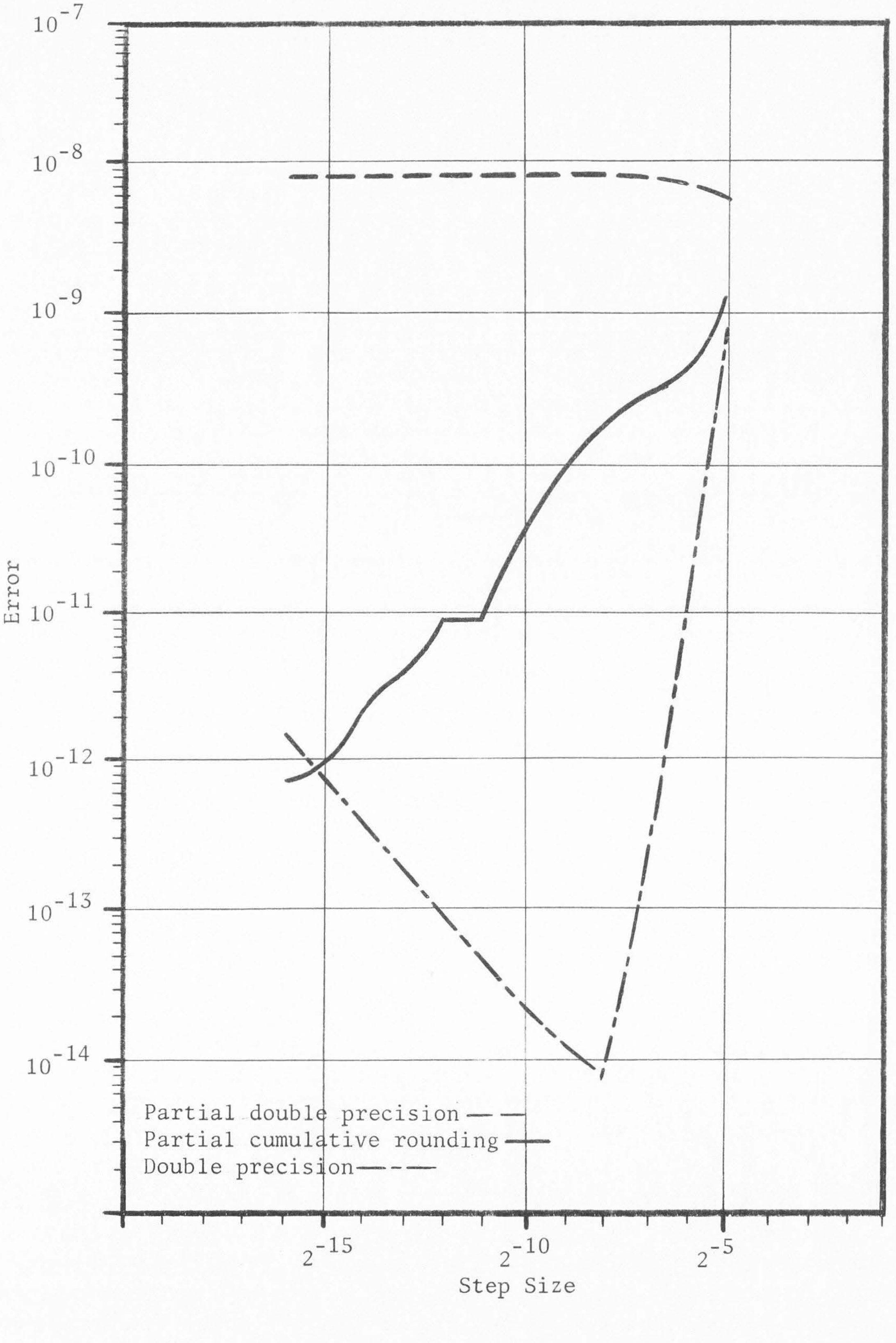

Figure 1. Error at x = 0.25 in solution of $y' = y^2$, $y(0) = 1.0$

REFERENCES

1. Conte, S. D. Elementary Numerical Analysis. McGraw-Hill, New York, New York. 1965.

2. Hansen, R. A. Round-off Procedures in the Numerical Treatment of Differential Equations. Ph.D. dissertation, University of Utah, Salt Lake City, Utah. 1964.

3. Henrici, P. Discrete Variable Methods in Ordinary Differential Equations. John Wiley and Sons. New York, New York. 1962.

APPENDIX

APPENDIX

The following theorem is a proof of (0.6) and (0.7). It is taken from Hansen [2, pp. 7-8] and is included for completeness.

Theorem A. If h, a_n, b_n, (n = 0, 1, 2, . . .), are real numbers, then the solution of the difference equation

$$r_0 = 0 \tag{A.1}$$

$$r_{n+1} = r_n (1 + h\, a_n) + b_n$$

is given by

$$r_n = \sum_{i=0}^{n-1} b_i + \prod_{j=1}^{n-1} \left(h\, a_j \prod_{k=j+1}^{n-1} (1 + h\, a_k) \right) \sum_{i=0}^{j-1} b_i, \tag{A.2}$$

and the inequality

$$1 + \sum_{j=1}^{n-1} \left| h\, a_j \prod_{k=j+1}^{n-1} (1 + h\, a_k) \right| \leq \prod_{j=1}^{n-1} (1 + h\, |a_j|) \tag{A.3}$$

is true for all $n \geq 1$.

Proof: First, (A.2) is true for n = 0. Therefore, the boundary condition is satisfied. Secondly, (A.2) and (A.3) are true for n = 1 since $b_0 = b_0$ and 1 = 1. Now let ℓ be any integer such that (A.2) and (A.3) are true, that is,

$$r_\ell = \sum_{i=0}^{\ell-1} b_i + \sum_{j=1}^{\ell-1} \left(h\, a_j \prod_{k=j+1}^{\ell-1} (1 + h\, a_k) \right) \sum_{i=0}^{j-1} b_i \tag{A.4}$$

and

$$\text{(A.5)} \qquad 1 + \sum_{j=1}^{\ell-1} \left| h\, a_j \prod_{k=j+1}^{\ell-1} (1 + h\, a_k) \right| \leq \prod_{j=1}^{\ell-1} (1 + h\, |a_j|).$$

For $\ell + 1$ we have from the difference equation that

$$r_{\ell+1} = b_\ell + (1 + h\, a_\ell)\, r_\ell$$

and

$$r_{\ell+1} = b_\ell + (1 + h\, a_\ell) \sum_{i=0}^{\ell-1} b_i + (1 + h\, a_\ell) \sum_{j=1}^{\ell-1} \left(h\, a_j \prod_{k=j+1}^{\ell-1} (1 + h\, a_k) \right) \sum_{j-1}^{u-1} b_i$$

$$= \sum_{i=0}^{\ell} b_i + h\, a_\ell \sum_{i=0}^{\ell-1} b_i + \sum_{j=1}^{\ell-1} \left(h\, a_j \prod_{k=j+1}^{\ell} (1 + h\, a_k) \right) \sum_{i=o}^{j-1} b_i$$

$$= \sum_{i=0}^{\ell} b_i + \sum_{j=1}^{\ell} \left(h\, a_j \prod_{k=j+1}^{\ell} (1 + h\, a_k) \right) \sum_{j-1}^{u-1} b_i .$$

Hence, (A.2) is true for all $n \geq 0$.

Now

$$1 + \sum_{j=1}^{\ell} \left| h\, a_j \prod_{k=j+1}^{\ell} (1 + h\, a_k) \right| = 1 + |h\, a_\ell| + \sum_{j=1}^{\ell-1} \left| h\, a_j \prod_{k=j+1}^{\ell} (1 + h\, a_k) \right|$$

$$\leq 1 + h|a_\ell| + \sum_{j=1}^{\ell-1} |h\, a_j \prod_{k=j+1}^{\ell-1} (1 + h\, a_k)|)(1 + h\, |a_\ell|)$$

$$\leq (1 + h\, |a_\ell|) \left[1 + \sum_{j=1}^{\ell-1} \left| h\, a_j \prod_{k=j+1}^{\ell-1} (1 + h\, a_k) \right| \right]$$

$$\leq \prod_{j=1}^{\ell} (1 + h\, |a_j|).$$

Thus (A.3) is true for all $n \geq 1$.

The following Lemma is a proof of equation (1.5)

Lemma A. If there exists $\tau_{m,j}$ such the $V_{m,j} = v_{m,j} = v_{m,j+1}\ h\ \tau_{m,j}$ $(j = 0, 1, 2, \ldots, n-1)$, then

$$\text{(A.6)} \qquad \sum_{i=0}^{j-1} V_{m,i} R_{m,i} = V_{m,j} \sum_{i=0}^{j-1} R_{m,i} + h \sum_{q=0}^{j-1} \tau_{m,q} \sum_{i=0}^{q} R_{m,i}$$

$$(j = 1, \ldots, n).$$

Proof: Let $n \geq 1$ be any integer. For $j = 1$ we have $V_{m,0} = V_{m,1} + h\ \tau_{m,0}$. Thus

$$V_{m,0} R_{m,0} = V_{m,1} R_{m,0} + h\ \tau_{m,0} R_{m,0}.$$

Assume (A.6) is true for $j = k$, i.e.,

$$\text{(A.7)} \qquad \sum_{i=0}^{k-1} V_{m,i} R_{m,i} = V_{m,k} \sum_{i=0}^{k-1} R_{m,i} + h \sum_{q=0}^{k-1} \tau_{m,q} \sum_{i=0}^{q} R_{m,i}.$$

Now

(A.8) $$V_{m,k} = V_{m,k+1} + h\ \tau_{m,k}$$ implies

(A.9) $$V_{m,k}\ R_{m,k} = V_{m,k+1}\ R_{m,k} + h\ \tau_{m,k}\ R_{m,k}.$$

Adding (A.7) and (A.9), we have

$$\sum_{i=0}^{k-1} V_{m,i}\ R_{m,i} + V_{m,k}\ R_{m,k} = V_{m,k} \sum_{i=0}^{k-1} R_{m,i}$$

$$+ h \sum_{q=0}^{k-1} \tau_{m,q} \sum_{i=0}^{q} R_{m,i} + V_{m,k+1}\ R_{m,k} + h\ \tau_{m,k}\ R_{m,k}.$$

Using (A.8) and collecting terms, we have

(A.10) $$\sum_{i=0}^{k} V_{m,i}\ R_{m,i} = V_{m,k+1} \sum_{i=0}^{k} R_{m,i} + h \sum_{q=0}^{k} \tau_{m,q} \sum_{i=0}^{q} R_{m,i}.$$

Thus, (A.6) is true for $j \leq n$.

VITA

Dale M. Rasmuson

Candidate for the Degree of

Master of Science

Thesis: Tests of Methods that Control Round-off Error

Major Field: Mathematics

Biographical Information:

Personal Data: Born at Logan, Utah, June 25, 1942, son of Ellwood W. and Fawn McFarland Rasmuson; married Linda White September 2, 1964; one child--Eric.

Education: Attended elementary school in Logan, Utah; graduated from Logan High School in 1960; received the Honors Degree of Bachelor of Arts from the University of Utah, with a major of mathematics, in 1966; completed requirements for the Master of Science degree, specializing in mathematics, at Utah State University in 1968.