AD-A210 118

REPORT DOCUMENTATION PAGE

1. OFFICIAL USE ONLY

2. REPORT DATE
November 1988

3. TYPE OF REPORT AND DATES COVERED
Final

4. TITLE AND SUBTITLE
Significant Digit Computation of the Incomplete Beta Function Ratios

5. FUNDING NUMBERS

6. AUTHOR(S)
Armido R. DiDonato
Alfred H. Morris, Jr.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Surface Warfare Center (K33)
Dahlgren, VA 22448-5000

8. PERFORMING ORGANIZATION REPORT NUMBER
NSWC TR 88-365

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (MAXIMUM 200 WORDS)

An algorithm is given for evaluating the incomplete beta function ratio $I_x(a,b)$ and its complement $1-I_x(a,b)$. Two new procedures are used with classical results. A listing of a transportable Fortran subroutine using this algorithm is given. The subroutine is accurate to 14 significant digits when the precision is not restricted by inherent error.

14. SUBJECT TERMS
incomplete beta function ratio   subroutine   Fortran software

15. NUMBER OF PAGES
82

16. PRICE CODE

17a. SECURITY CLASSIFICATION (OF REPORT)
UNCLASSIFIED

17b. SECURITY CLASSIFICATION (OF PAGE)
UNCLASSIFIED

17c. SECURITY CLASSIFICATION (OF ABSTRACT)
UNCLASSIFIED

18. MEDIA TYPE

FOREWORD

The work described in this report was done in the Space and Surface Systems Division and the Computer and Information Systems Division of the Strategic Systems Department. Its purpose was to design Fortran software, suitable for a high quality mathematics and/or statistics subroutine library, which yields the incomplete beta function ratios over an extensive range of the variables.

The authors are indebted to Dr. James Perry of NSWC for proving a result which is used in this work. His proof is given in Appendix A. The authors also wish to thank Ms. Dottie Burgess for making Figures 1 and 3.

Approved by:

R. L. Schmidt

R. L. SCHMIDT, Head
Strategic Systems Department

CONTENTS

I. INTRODUCTION

The incomplete beta function $I_x(a,b)$ is defined by

$$I_x(a,b) = G(a,b)\int_0^x t^{a-1}(1-t)^{b-1}dt, \qquad a>0,\ b>0,\ 0\le x\le 1 \tag{1}$$

$$B(a,b) = 1/G(a,b) = \int_0^1 t^{a-1}(1-t)^{b-1}dt = \Gamma(a)\Gamma(b)/\Gamma(a+b), \qquad [10,\ p.36], \tag{2}$$

where the gamma function $\Gamma(u)$ is given by

$$\Gamma(u) = \int_0^\infty e^{-t}\, t^{u-1}\, dt, \qquad u>0. \tag{3}$$

Thus $I_1(a,b)=1$. In addition, if $0<x<1$ then $I_x(0,b)=1$ and $I_x(a,0)=0$.

The quantity $1-I_x(a,b)$ is called the ***complement*** of $I_x(a,b)$. Using $u=1-t$ in (1) and (2) yields

$$1-I_x(a,b) = I_y(b,a), \qquad y=1-x. \tag{4}$$

The function I_x occurs in many branches of science, including atomic physics, fluid dynamics, transmission theory, lattice theory, and operations research. It is perhaps best known for its extensive applications in statistics. In particular, the well-known central F-distribution $P(F_0|\nu_1,\nu_2)$ can be obtained from I_x by the following substitutions :

$$a=\nu_1/2\ , \quad b=\nu_2/2\ , \quad t=\nu_1 F/[\nu_2+\nu_1 F]\ , \quad F_0=\frac{bx}{a(1-x)}, \tag{5}$$

where

$$P(F_0|\nu_1,\nu_2) = \nu_1^{\nu_1/2}\nu_2^{\nu_2/2}\, G\left(\frac{\nu_1}{2},\frac{\nu_2}{2}\right)\int_0^{F_0} F^{(\nu_1-2)/2}\,(\nu_2+\nu_1 F)^{-(\nu_1+\nu_2)/2}dF \tag{6}$$

$$Q(F_0|\nu_1,\nu_2) = 1-P(F_0|\nu_1,\nu_2) = I_{1-x}(\nu_2/2,\nu_1/2). \tag{7}$$

The incomplete beta function is also directly related to the Student's t-distribution $A(t_0|\nu)$ and the binomial distribution $E(n,r,x)$, where

$$P(|t|\le t_0) = A(t_0|\nu) = \frac{2}{\sqrt{\nu}}\, G\left(\frac{1}{2},\frac{\nu}{2}\right)\int_0^{t_0}(1+t^2/\nu)^{-(\nu+1)/2}dt \tag{8}$$

$$= 1-I_x(\nu/2,1/2), \qquad x=\nu/[\nu+t_0^2],$$

$$E(n,r,x) = \sum_{i=r}^{n}\binom{n}{i}x^i(1-x)^{n-i} = I_x(r,n-r+1). \tag{9}$$

Derivations of these well-known results are given in [4].

Procedures for computing I_x date back to Newton. A historical survey outlining some of the analytical and approximation methods used for evaluating I_x is given by Dutka [7]. An extensive literature search for a robust algorithm to compute I_x did not reveal a publication that would lead to a subroutine acceptable for inclusion in a high quality main frame mathematics library such as the NSWC Library of Mathematics Subroutines (NSWCLIB), [9]. The best procedure found was in a report written by Amos and Daniel [2]. For a special case however, where a and b take positive integer and half-integer values, an algorithm exists, with an associated Fortran subroutine ISUBX, which yields I_x with 9-10 decimal-digit accuracy, [4, 5]. It is contained in NSWCLIB.

In this report an algorithm is given for computing I_x and $1-I_x$. A transportable Fortran subroutine named BRATIO has been written which uses the algorithm. BRATIO is designed for use on computers having k-digit single precision floating arithmetics where $6 \leq k \leq 14$. On the CDC 6000-7000 series computers, BRATIO yields results accurate up to 14 significant digits for both I_x and $1-I_x$. BRATIO is available for general use in the NSWC mathematics subroutine library, [9].

A primary region of difficulty for computing $I_x(a,b)$ and $1-I_x(a,b)$ has been when a and/or b is large and $x \approx a/(a+b)$. In this region I_x changes rapidly from 0 to 1. A continued fraction, with new weighting factors, and a new asymptotic expansion are used to treat this region satisfactorily.

Section II contains the basic equations and algorithms used for BRATIO. Section III describes in a, b, x space the regions of use for these basic relations. An associated flowchart for BRATIO is included. Section IV describes a number of specialized algorithms required in order to use the basic relations effectively. Section V briefly summarizes the accuracy and efficiency of BRATIO, and section VI contains a few examples using BRATIO. Appendices A, B, and C contain proofs for some results which are used in BRATIO. A Fortran listing of BRATIO and its required subprograms is given in Appendix D.

II. BASIC RELATIONS

Throughout we shall use

$$p=a/(a+b) \qquad q=1-p=b/(a+b). \tag{10}$$

Also since I_x and $1-I_x$ are to be computed to the greatest possible accuracy,

$$y=1-x \tag{11}$$

is required as input in addition to a, b, and x. In BRATIO, relations (12)−(16) are used to compute either I_x or its complement. Their domains of application are given in Section III.

BPSER

$$I_x(a,b)\approx G(a,b)\frac{x^a}{a}\left(1+a\sum_{j=1}^{N}\frac{(1-b)(2-b)\cdots(j-b)}{j!\,(a+j)}x^j\right) \tag{12}$$

The series is obtained from (1) by replacing the second factor in the integrand with its binomial expansion. Relation (12) is used only when $x\le 0.7$ and $b\le 1$, or $bx\le 0.7$.

Given a tolerance $\epsilon>0$, (12) is computed by the function BPSER(a,b,x,ϵ), where

$$I_x(a,b)\approx G(a,b)\frac{x^a}{a}\left(1+a\sum_{n=1}^{N}w_n\right) \tag{12.1}$$

$$w_n=C_n/(a+n), \qquad C_n=C_{n-1}(1-b/n)x, \quad C_0=1 \tag{12.2}$$

N = the smallest integer such that $a|w_N|\le\epsilon$.

BUP

$$I_x(a,b)-I_x(a+N,b)=x^a y^b\sum_{j=1}^{N}\frac{\Gamma(a+b+j-1)}{\Gamma(b)\Gamma(a+j)}x^{j-1}, \qquad (N\ge 1) \tag{13}$$

Given a tolerance ϵ, (13) is computed by the function BUP(a,b,x,y,N,ϵ). BUP is used only with BPSER or the next relation to be described, BGRAT. Relation (13) follows from $I_x(a+1,b)=I_x(a,b)-x^a y^b G(a,b)/a$, which can be obtained by substituting

$$(a+b)t^a(1-t)^{b-1} = at^{a-1}(1-t)^{b-1}-\frac{d}{dt}\left[t^a(1-t)^b\right] \tag{13.1}$$

in

$$I_x(a+1,b)=\frac{a+b}{aB(a,b)}\int_0^x t^a(1-t)^{b-1}dt.$$

Equation (13) can be rewritten in the form

$$I_x(a,b)-I_x(a+N,b)=\frac{x^a y^b}{a\,B(a,b)}\sum_{i=0}^{N-1}d_i x^i, \qquad d_{i+1}=\frac{a+b+i}{a+1+i}d_i, \qquad d_0=1. \tag{13.2}$$

If $b \le 1$ then for $i \ge 0$, $d_{i+1} \le d_i$ and the sequence $h_i \equiv d_i x^i$ is monotonically decreasing. Thus, the computation of the sum $\sum h_i$ can be terminated when a term $h_m = d_m x^m$ is reached that satisfies $h_m \le \epsilon \sum_{i=0}^{m} h_i$, or $m = N-1$.

When $b>1$ then $h_i \ge h_{i+1}$ if and only if $x \le (a+1+i)/(a+b+i) \equiv r_i$. Also we note that $r_i < r_j$ when $i<j$. Thus if k is the largest integer such that $x \ge r_{k-1}$, then

$$h_0 \le h_1 \le \dots \le h_k \ge h_{k+1} \ge \dots \ge h_{n-1}.$$

Therefore, if k is the largest integer for which $k \le (b-1)x/y - a$, then the computation of the sum $\sum h_i$ can be terminated when a term h_m $(m>k)$ is met that satisfies $h_m \le \epsilon \sum_{i=0}^{m} h_i$, or $m=N-1$.

BGRAT

$$I_x(a,b) \approx M \sum_{n=0}^{L} p_n J_n(b,u), \quad M = \frac{\Gamma(a+b)}{\Gamma(a) T^b} r, \quad a>b \tag{14}$$

$$T = a + \frac{b-1}{2}, \quad u = -T \ln x, \quad r = e^{-u} u^b / \Gamma(b), \quad J_0(b,u) = Q(b,u)/r \tag{14.1}$$

$$p_n = (b-1)c_n + \frac{1}{n} \sum_{m=1}^{n-1} (mb-n)\, c_m\, p_{n-m}\,, \quad c_m = \frac{1}{(2m+1)!}\,, \quad p_0 = 1 \tag{14.2}$$

$$Q(b,u) = \int_u^{\infty} \frac{e^{-t}\, t^{b-1}}{\Gamma(b)}\, dt \quad \text{(incomplete gamma function)} \quad [1;\ \text{p.260}] \tag{14.3}$$

$$J_{n+1}(b,u) = \frac{(b+2n)(b+2n+1)}{4T^2} J_n(b,u) + \frac{u+b+2n+1}{4T^2} \left(\frac{\ln x}{2}\right)^{2n} \tag{14.4}$$

$$L = \text{smallest integer such that } |p_L J_L| \le \epsilon \left(\sum_{n=0}^{L} p_n J_n + w_0/M \right). \tag{14.5}$$

Relation (14) is used only when $a \ge 15$, $b \le 1$, and $x>0.7$. Given ϵ, (14) is computed by the subroutine BGRAT(a,b,x,y,w,ϵ,IERR) where w and IERR are variables. Given an initial value w_0 for w, then BGRAT assigns w the value $w_0 + I_x(a,b)$. IERR is an indicator which is set when underflow forces the computation of (14) to end prematurely.

Equation (14) was derived by Wise [15]. Our derivation follows:

If $t = e^{-y/T}$, then

$$\int_0^x t^{a-1}(1-t)^{b-1} dt = \frac{2^{b-1}}{T} \int_u^{\infty} e^{-y} \sinh^{b-1}\left(\frac{y}{2T}\right) dy \tag{14.6}$$

where T and u are defined above. Substituting

$$\sinh^{b-1}z = z^{b-1}\left(\sum_{n=0}^{\infty} c_n z^{2n}\right)^{b-1} = \sum_{j=0}^{\infty} p_j\, z^{2j+b-1} \tag{14.7}$$

into (14.6), where the c_n and p_j satisfy (14.2), yields (14) where

$$J_n(b,u) = \frac{1}{(2T)^{2n}}\,\frac{\Gamma(b+2n)}{r\Gamma(b)}\,Q(b+2n,u). \tag{14.8}$$

Then (14.4) follows from (14.8) and

$$Q(b+2n+2,u) = Q(b+2n,u) + \frac{e^{-u}\,u^{b+2n}}{\Gamma(b+2n+2)}\Big(u+b+2n+1\Big), \qquad [1;\ 6.5.21] \tag{14.9}$$

where (14.9) is the result of two integrations by parts of its left hand side.

BFRAC

$$I_x(a,b) \approx \frac{x^a\,y^b}{B(a,b)}\left(\frac{\alpha_1}{\beta_1+}\;\frac{\alpha_2}{\beta_2+}\;\cdots\;\frac{\alpha_m}{\beta_m}\right), \qquad x \le p \equiv \frac{a}{a+b} \tag{15}$$

$$\alpha_1 = 1, \qquad \beta_1 = \frac{a}{a+1}(\lambda+1), \qquad \lambda = a-(a+b)x = (a+b)(p-x) \tag{15.1}$$

$$\alpha_{n+1} = \frac{(a+n-1)(a+b+n-1)}{(a+2n-1)^2}\,n(b-n)x^2, \qquad n \ge 1 \tag{15.2}$$

$$\beta_{n+1} = n + \frac{n(b-n)x}{a+2n-1} + \frac{a+n}{a+2n+1}\Big[\lambda+1+n(1+y)\Big], \qquad n \ge 0 \tag{15.3}$$

$$\left\{\begin{array}{lll} A_{n+1} = \beta_{n+1}A_n + \alpha_{n+1}A_{n-1} & A_0 = 0, & A_1 = \alpha_1 \\ B_{n+1} = \beta_{n+1}B_n + \alpha_{n+1}B_{n-1} & B_0 = 1, & B_1 = \beta_1 \end{array}\right. \tag{15.4}$$

$$\frac{A_n}{B_n} = \frac{\alpha_1}{\beta_1+}\;\frac{\alpha_2}{\beta_2+}\;\cdots\;\frac{\alpha_n}{\beta_n}. \tag{15.5}$$

$m =$ the smallest integer such that

$$|A_m/B_m - A_{m-1}/B_{m-1}| \le \epsilon\,|A_m/B_m|.$$

Equation (15) is new and is used over a very large part of the domain: $a>1$, $b\ge 40$, with $x \le p$. Given a tolerance ϵ, (15) is computed by the function BFRAC(a,b,x,y,λ,ϵ). The weighting factors c_n, given below, which control overflow and underflow, appear to be new.

Relation (15) is obtained by considering the classical expansion

$$I_x(a,b) = \frac{x^a y^b}{aB(a,b)} \left(\frac{1}{1+} \frac{d_1}{1+} \frac{d_2}{1+} \cdots \right) \tag{15.6}$$

$$d_{2n} = \frac{n(b-n)}{(a+2n-1)(a+2n)} x, \quad n>0 \tag{15.7}$$

$$d_{2n+1} = -\frac{(a+n)(a+b+n)}{(a+2n)(a+2n+1)} x, \quad n \geq 0. \qquad [1;\ 26.5.8],\ [3],\ [13] \tag{15.8}$$

For large a where $a \gg b$, we note that $d_{2n} \approx 0$ and $d_{2n+1} \approx -(1+b/a)x$. Thus if R_n is the iterate

$$R_n = \frac{1}{1+} \frac{d_1}{1+} \frac{d_2}{1+} \cdots \frac{d_{n-2}}{1+d_{n-1}},$$

then most of the change of values of these iterates occurs in every other iterate. Consequently, the "associated" continued fraction $\frac{a_1}{b_1+} \frac{a_2}{b_2+} \cdots$ is considered where

$$a_1 = 1, \qquad a_{n+1} = -d_{2n-1} d_{2n}$$

$$b_1 = 1 + d_1, \qquad b_{n+1} = 1 + d_{2n} + d_{2n+1}, \qquad n \geq 1. \qquad [14;\ p.20]$$

The iterates R_{2n} are the iterates for this expansion as already pointed out by Aroian in [3] (with some corrections given in [13]). Hence

$$I_x(a,b) = \frac{x^a y^b}{aB(a,b)} \left(\frac{a_1}{b_1+} \frac{a_2}{b_2+} \cdots \right).$$

Now for large a where $a \gg b$, it is clear that $d_{2n} \approx 0$ forces $a_n \approx 0$. However, for $x \approx p$ we also find that $d_{2n} \approx 0$ forces $b_n \approx 0$. This can cause division of 0 by 0 when the iterates of this continued fraction are computed, or it can cause division to overflow. Thus, this continued fraction is also not considered appropriate for computational purposes.

In order to eliminate the problems arising from $d_{2n} \approx 0$, we rescale the coefficients a_n and b_n with weighting factors c_n such that

$$\bar{\alpha}_1 = c_1 a_1, \quad \alpha_n = c_{n-1} c_n a_n \ (n \geq 2), \quad \beta_n = c_n b_n \quad (n \geq 1),$$

where

$$c_n = a + 2(n-1), \qquad n \geq 1. \tag{15.9}$$

Then the iterates of $\frac{\bar{\alpha}_1}{\beta_1+} \frac{\alpha_2}{\beta_2+} \cdots$ are the scaled iterates for $\frac{a_1}{b_1+} \frac{a_2}{b_2+} \cdots$ and we obtain

$$I_x(a,b) = \frac{x^a y^b}{aB(a,b)} \left(\frac{\bar{\alpha}_1}{\beta_1+} \frac{\alpha_2}{\beta_2+} \cdots \right) = \frac{x^a y^b}{B(a,b)} \left(\frac{1}{\beta_1+} \frac{\alpha_2}{\beta_2+} \cdots \right),$$

which is relation (15). The expressions (15.4) and (15.5) for computing (15) follow from Theorem 2 [1; p.19], where $\lim_{n\to\infty} \frac{A_n}{B_n} = \frac{\alpha_1}{\beta_1+} \frac{\alpha_2}{\beta_2+} \cdots$.

If $n \le b$, then $\alpha_n \ge 0$ and β_n is a positive value not near 0. To insure that the maximum number of iterations satisfies $n \le b$, (15) is applied only when $b \ge 40$. In this case, x must also be a sufficient distance from p when $a > 100$.

BASYM

Let γ be an arbitrary positive scaling factor that is assigned below. Then

$$I_x(a,b) \approx U \frac{2}{\sqrt{\pi}} e^{-z^2} \sum_{n=0}^{N} d_n \; J_n(z) \, (\beta\gamma)^n, \qquad x \le p \tag{16}$$

$$p = \frac{a}{a+b}, \quad q = \frac{b}{a+b}, \quad \beta = \sqrt{\frac{ab}{(a+b)^3}}, \quad U = \frac{p^a q^b}{B(a,b)} \sqrt{\frac{2\pi(a+b)}{ab}}$$

$$\phi(t) = t - 1 - \ln t, \qquad t > 0 \tag{16.1}$$

$$\varphi(t) = a\phi\left(\frac{t}{p}\right) + b\phi\left(\frac{1-t}{q}\right), \qquad 0 < t < 1$$

$$z = \sqrt{\varphi(x)}$$

$$a_n = \gamma^{-n} \lambda_n, \qquad \gamma > 0 \tag{16.2}$$

$$\lambda_n = \frac{2}{n+2} \left[q \, p^{-n} + (-1)^n \, p \, q^{-n} \right], \qquad (\lambda_0 = 1)$$

$$b_0^{(r)} = 1 \tag{16.3}$$

$$b_n^{(r)} = r a_n + \frac{1}{n} \sum_{i=1}^{n-1} \left[r(n-i) - i \right] b_i^{(r)} a_{n-i}, \qquad n = 1,2,\ldots \qquad [11]$$

$$c_n = \frac{1}{n} b_{n-1}^{(-n/2)}, \qquad (n \ge 1)$$

$$d_0 = 1, \quad d_n = -\sum_{i=0}^{n-1} d_i \, c_{n-i+1} \tag{16.4}$$

$$J_n(z) = 2^{\frac{n}{2}-1} e^{z^2} \int_z^{\infty} e^{-v^2} v^n \, dv. \tag{16.5}$$

$$\beta\gamma = \sqrt{q/a}, \qquad a \le b \tag{16.6}$$

$$\beta\gamma = \sqrt{p/b}, \qquad a \ge b.$$

N is defined as the smallest integer such that

$$\left| d_N \; J_N(z) (\beta\gamma)^N \right| + \left| d_{N-1} \; J_{N-1}(z) (\beta\gamma)^{N-1} \right| \le \epsilon \sum_{n=0}^{N} d_n \; J_n(z) (\beta\gamma)^n.$$

Relation (16) is used only when a and b are large and $x \approx p$. It appears to be new. Given $\lambda = (a+b)(p-x)$ and a tolerance ϵ, (16) is computed by the function BASYM(a,b,x,y,λ,ϵ). The expansion is obtained by writing (1) in the form

$$I_x(a,b) = \frac{p^a q^b}{B(a,b)} \int_0^x \left(\frac{t}{p}\right)^a \left(\frac{1-t}{q}\right)^b \frac{dt}{t(1-t)}, \qquad 0<x<1. \tag{16.7}$$

From (16.1), $\varphi(t) > 0$ for $t \neq p$ and

$$\varphi(x) = -\left(a \ln\frac{x}{p} + b \ln\frac{y}{q}\right) \tag{16.8}$$

Thus if $u = \sqrt{\varphi(t)}$ for $t \leq p$ and $u = -\sqrt{\varphi(t)}$ for $t > p$, then

$$I_x(a,b) = U\, \beta \sqrt{\frac{2}{\pi}} \int_z^\infty e^{-u^2} \frac{u}{p-t}\, du, \tag{16.9}$$

where $z = \sqrt{\varphi(x)}$ if $x \leq p$ and $z = -\sqrt{\varphi(x)}$ if $x > p$. Hereafter, we shall assume that $x \leq p$. In order to use (16.9), $u/(p-t)$ will be replaced by its Maclaurin series in u.

From (16.8) and

$$a \ln\frac{t}{p} = a \ln\left(1 - \frac{p-t}{p}\right) = -a \sum_{n=1}^\infty \frac{1}{n} \left(\frac{p-t}{p}\right)^n$$

$$b \ln\frac{1-t}{q} = b \ln\left(1 + \frac{p-t}{q}\right) = -b \sum_{n=1}^\infty \frac{1}{n} \left(\frac{t-p}{q}\right)^n,$$

one obtains

$$u^2 = \varphi(t) = \frac{1}{2\beta^2}(p-t)^2 \sum_{n \geq 0} \lambda_n (p-t)^n, \qquad |p-t| < \min\{p,q\},$$

where λ_n is given in (16.2). Thus if $a_n = \gamma^{-n} \lambda_n$ for $\gamma > 0$, then $2(\beta\gamma u)^2 = s^2 A$, where $A = \sum_{n \geq 0} a_n s^n$ and $s = \gamma(p-t)$. Hence, if $P(s) = s\sqrt{A}$ then $\sqrt{2}\, \beta\gamma u = P(s)$. If $v = P(s)$, since P is analytic at 0 with $P(0) = 0$, by the Lagrange-Bürmann expansion [8; p.58] the inverse $s = P^{-1}(v)$ of P is given by

$$s = \sum_{n \geq 1} c_n v^n,$$

$$c_n = \frac{1}{n} \mathbf{res}(P^{-n}) = \frac{1}{n} \mathbf{res}(s^{-n} A^{-n/2}),$$

where $\mathbf{res}(P^{-n})$ is the residue of the series P^{-n}. Also for any $r \neq 0$

$$A^r = \sum_{k \geq 0} b_k^{(r)} s^k,$$

where $b_k^{(r)}$ is given in (16.3), [11]. Consequently,

$$c_n = \frac{1}{n} b_{n-1}^{(-n/2)}$$

$$\sqrt{2}\,\beta \frac{u}{p-t} = \frac{v}{s} = 1/\sum_{n\geq 1} c_n v^{n-1} = \sum_{n\geq 0} d_n v^n,$$

where d_n is given in (16.4), [11].

Now if $h=a/b$ for $a\leq b$ and $h=b/a$ for $a>b$, let $\gamma=(1+h)/h$, (see (16.6)). Then $A=\sum_{n\geq 0} a_n s^n$ for $|s|<1$ and

$$a_n = \frac{2}{n+2} q\left[1+(-1)^n h^{n+1}\right], \quad a\leq b \tag{16.10}$$

$$a_n = \frac{2}{n+2} p\left[(-1)^n + h^{n+1}\right], \quad a>b.$$

It can be shown for $a=b$ that the series $v/s=\sum_{n\geq 0} d_n v^n$ has a radius of convergence no larger than $\sqrt{2\pi}$. Indeed, we have

$$u^2 = av^2 = -a\ln(1-s^2),$$

and, with v complex, $s=0$ when $v=\sqrt{2\pi}\exp(i\pi/4)$.

If $a\leq b$ then $A=\sum_{n\geq 0} a_n s^n \geq 0$ for $0\leq s<1$ since each $a_n\geq 0$. Hence $v=s\sqrt{A}$ is defined and real for $0\leq s<1$. Also, from (16.8) we *note that* $v\to\infty$ *when* $s\to 1$, and $dv/ds=(s/v)[(1-s)(1+hs)]^{-1}\neq 0$ for $0<s<1$. Consequently, $f(v)=v/s=\sqrt{A}$ can be regarded as a function of v for all $v\geq 0$. Since it is also true that the derivatives $f^{(n)}(v)$ are bounded for $v\geq 0$ (see Appendix C), let $M_n=\sup\{|f^{(n)}(v)|:v\geq 0\}$ for $n\geq 1$. Then $M_1\leq 1$ (see Appendix C), and for

$$\psi_n(v)=f(v)-\sum_{i=0}^{n-1} d_i v^i,$$

$|\psi_n(v)|\leq M_n v^n/n!$ $(v>0)$ by the Taylor formula with remainder. Therefore, from $\sqrt{2}\beta u/(p-t)=f(v)$ and (16.9) we obtain

$$I_x(a,b) = U\frac{2}{\sqrt{\pi}} e^{-z^2}\left[\sum_{i=0}^{n-1} d_i\,(\beta\gamma)^i J_i(z) + E_n\right], \qquad z\geq 0$$

$$E_n = \frac{1}{2} e^{z^2}\int_z^\infty e^{-u^2}\psi_n(\sqrt{2}\beta\gamma u)du,$$

where $J_n(z)$ is given by (16.5). Also

$$|E_n|\leq M_n/n!(\beta\gamma)^n J_n(z) = M_n/n!J_n(z)[a(1+h)]^{-n/2} \tag{16.11}$$

so that $|E_n| \to 0$ when $a \to \infty$ for fixed h. Consequently, (16) is asymptotic for $a \le b$. Also we note that

$$J_n(z) = 2^{(n/2)-2} e^{z^2} \Gamma[(n+1)/2]\ Q[(n+1)/2, z^2)],$$

where Q is defined by (14.3), and that $J_n(z)$ can be computed recursively by

$$J_0(z) = (\sqrt{\pi}/4) e^{z^2} \operatorname{erfc}(z), \qquad J_1 = \alpha^{-3/2}$$

$$J_n(z) = 2^{-3/2} (\sqrt{2}\, z)^{n-1} + (n-1) J_{n-2}(z), \qquad n \ge 2.$$

When $a > b$ the situation is less satisfactory, since it has not yet been shown that (16) is asymptotic. Nevertheless, the utility of (16) has been established for $y < 1.05q$ when $b \ge 100$ by extensive computer testing.

Finally, we observe that the defintion of U given in (16) is not suitable for computational purposes. U can be accurately evaluated using

$$\Delta(a) = \ln\ \Gamma(a) - (a - \tfrac{1}{2}) \ln a + a - \tfrac{1}{2} \ln (2\pi) \tag{16.12}$$

$$\ln U = \Delta(a+b) - \Delta(a) - \Delta(b).$$

III. DOMAINS FOR SUBPROGRAMS

In this section we specify the regions of application for the five subprograms discussed in the previous section. A flowchart is given in Fig. 1.

In order to establish such regions the following conditions must be met:

(a) The applied algorithms must be efficient and yield the desired accuracy over such regions.

(b) It is necessary that $J \leq .9$, where J has the value $I_x(a,b)$ or $I_y(b,a)$. J is computed by the proper choice, acccording to (a), of one or more of the subprograms based on (12)−(16). The complement is then obtained as $1-J$.

Even though analysis was carried out to predict the efficiency and achievable accuracy of the basic relations over various domains, (a) was established with exhaustive testing by Morris using double precision versions of the subprograms.

Since J is to be computed to the greatest possible accuracy, the relative tolerance to be satisfied will be $\epsilon = \max\{\epsilon_0, 10^{-15}\}$ where ϵ_0 is the smallest number for which $1+\epsilon_0 > 1$ for the floating point *arithmetic being* used. The restriction that $\epsilon \geq 10^{-15}$, thereby limiting the maximum precision to 14−15 significant digits, is made since many of the supporting subprograms are accurate to a maximum of 14 digits (see section IV).

Arguments are presented near the end of this section showing that condition (b) is always satisfied. For easy reference we have:

BPSER(a,b,x,ϵ)	Relation(12)
BUP(a,b,x,y,N,ϵ)	Relation(13)
BGRAT$(a,b,x,y,w,15\epsilon,IERR)$	Relation(14)
BFRAC$(a,b,x,y,\lambda,15\epsilon)$	Relation(15)
BASYM$(a,b,x,y,\lambda,100\epsilon)$	Relation(16)

There are two main domains to consider, namely $\min(a,b) \leq 1$ and $\min(a,b) > 1$. It should be recalled that if a and b, and x and y are interchanged in one of the above subprograms, then the subprogram, except for BUP, yields $J = I_y(b,a) = 1 - I_x(a,b)$ with $1 - J = I_x(a,b)$. BUP gives on the interchange $I_y(b,a) - I_y(b+N,a)$.

$\underline{\min(a,b)\leq 1}$

If $x>1/2$, then a and b, and x and y are interchanged. For $x\leq 1/2$, (17) is used for computing $J=I_x(a,b)$ and (18)−(20) are used for computing $J=I_y(a,b)$. In (19) and (20), w_0 is the initial value of w and J is the final value of w.

BPSER(a,b,x,ϵ)	$\max(a,b)\leq 1,\ a\geq\min(0.2,b)$	(17)
	$\max(a,b)\leq 1,\ a<\min(0.2,b),\ x^a\leq 0.9$	(17.1)
	$\max(a,b)>1,\ b\leq 1$	(17.2)
	$\max(a,b)>1,\ b>1,\ x<0.1,\ (bx)^a\leq 0.7$	(17.3)
BPSER(b,a,y,ϵ)	$\max(a,b)\leq 1,\ a<\min(0.2,b),\ x^a>0.9,\ x\geq 0.3$	(18)
	$\max(a,b)>1,\ b>1,\ x\geq 0.3$	(18.1)

BGRAT(b,a,y,x,w,15ϵ,IERR), $w_0=0$

	$\max(a,b)>1,\ b>15,\ 0.1\leq x<0.3$	(19)
	$\max(a,b)>1,\ b>15,\ x<0.1,\ (bx)^a>0.7$	(19.1)

BGRAT(b+N,a,y,x,w,15ϵ,IERR), $w_0=$BUP(b,a,y,x,N,ϵ), $N=20$

	$\max(a,b)>1,\ b>1,\ 0.1\leq x<0.3,\ b\leq 15$	(20)
	$\max(a,b)>1,\ b>1,\ x<0.1,\ (bx)^a>0.7,\ b\leq 15$	(20.1)
	$\max(a,b)\leq 1,\ a<\min(0.2,b),\ x^a>0.9,\ x<0.3$	(20.2)

$\underline{\min(a,b)>1}$

If $x>p$ then a and b, and x and y are interchanged. For $x\leq p$, (21)−(26) are used for computing $J=I_x(a,b)$. In (22)−(24) N is the largest integer less than b and $\bar{b}=b-N$. Also, in (23) and (24) w_0 is the initial value of w and J is the final value of w.

BPSER(a,b,x,ϵ)	$b<40,\quad bx\leq .7$	(21)

BUP($\bar{b}$,a,y,x,N,ϵ) + BPSER(a,$\bar{b}$,x,ϵ)

	$b<40,\quad bx>.7,\quad x\leq .7$	(22)

BGRAT(a,$\bar{b}$,x,y,w,15ϵ,IERR), $w_0=$BUP($\bar{b}$,a,y,x,N,ϵ)

	$b<40,\quad x>.7,\quad a>15$	(23)

$BGRAT(a+M,\bar{b},x,y,w,15\epsilon,IERR)$, $\quad M=20$

$w_0=BUP(\bar{b},a,y,x,N,\epsilon)+BUP(a,\bar{b},x,y,M,\epsilon)$

$$b<40, \quad x>.7, \quad a\leq 15 \tag{24}$$

$BASYM(a,b,x,y,\lambda,100\epsilon)$

$$b\geq 40, \quad 100<a\leq b, \quad x\geq .97p \tag{25}$$

$$b\geq 40, \quad 100<b<a, \quad y\leq 1.03q \tag{25.1}$$

$BFRAC(a,b,x,y,\lambda,15\epsilon)$

$$b\geq 40, \quad a\leq b, \quad a\leq 100 \tag{26}$$

$$b\geq 40, \quad 100<a\leq b, \quad x<.97p \tag{26.1}$$

$$b\geq 40, \quad a>b, \quad b\leq 100 \tag{26.2}$$

$$b\geq 40, \quad 100<b<a, \quad y>1.03q \tag{26.3}$$

These statements are summarized in the flowchart for BRATIO in Fig.1. Proofs are now given which verify that $J\leq 0.9$ is always satisfied (requirement b above). The arguments use the facts that $\ln\Gamma(t)$ is strictly convex for $t>0$, [1; 6.4.10], and that $I_x(a,b)$ is a decreasing function of a and an increasing function of b. The latter result is proven in Appendix B. Since $\ln\Gamma(t)$ is strictly convex for $t>0$, we also note that $\psi(t)=\frac{d}{dt}\{\ln\Gamma(t)\}$, [1; 6.3.1], is an increasing function of t.

For (17): $a\leq 1$, $b\leq 1$, $x\leq 1/2$, $a\geq \min(.2,b)$

If $a\geq 0.2$ then

$$J=I_x(a,b)\leq I_x(a,1)=x^a\leq (1/2)^{.2}=0.8706.$$

If $a\geq b$ then

$$J=I_x(a,b)\leq I_x(a,a)\leq I_{1/2}(a,a).$$

Also $I_{1/2}(a,a)=0.5$ from (4).

For (17.1): $a\leq 1$, $b\leq 1$, $x\leq 1/2$, $a<\min(.2,b)$, $x^a\leq 0.9$

$$J=I_x(a,b)\leq I_x(a,1)=x^a\leq 0.9$$

For (17.2): $a>1$, $b\leq 1$, $x\leq 1/2$

$$J=I_x(a,b)\leq I_x(1,1)=x\leq 1/2$$

For (17.3): $a\leq 1$, $b>1$, $x<0.1$, $(bx)^a\leq 0.7$

Integrating (1) by parts gives

$$I_x(a,b)=G(a,b)\left\{\frac{x^a}{a}(1-x)^{b-1}+\frac{b-1}{a}\int_0^x t^a(1-t)^{b-2}dt\right\}$$

$$\leq G(a,b)\left\{\frac{x^a}{a}(1-x)^{b-1}+\frac{b-1}{a}x^a\int_0^x (1-t)^{b-2}dt\right\}=G(a,b)\frac{x^a}{a}.$$

Let

$$F(a,b)=\Gamma(a+b)/[\Gamma(b)b^a], \quad 0\le a\le 1, \quad b\ge 1.$$

Since $\partial/\partial a[\ln F]=\psi(a+b)-\ln b$ is an increasing function of a for $a>0$, lnF is strictly convex and hence F is strictly convex for $a>0$. Thus $F(a,b)\le 1$ for $0\le a\le 1$ since $F(0,b)=F(1,b)=1$. Hence

$$J=I_x(a,b)\le (bx)^a/\Gamma(a+1)\le 0.7/\Gamma(1.46163\cdots)<0.791.$$

For (18): $a\le 1$, $b\le 1$, $0.3\le x\le 0.5$, $a<\min(.2,b)$, $x^a>0.9$

From (1), with

$$(1-t)^{b-1}\ge 1, \quad 0\le t\le x,$$

$$I_x(a,b)\ge H(a)\frac{b}{a+b}x^a\ge H(a)(0.9)/2\ge .45, \qquad (a<b),$$

where

$$H(a)=\Gamma(a+b+1)/[\Gamma(a+1)\Gamma(b+1)] \quad (0\le a\le 1).$$

The last inequality on I_x follows from the fact that H(a) is increasing and that $H(0)=1$. Hence

$$J=I_y(b,a)=1-I_x(a,b)\le 0.55.$$

For (18.1): $a\le 1$, $b>1$, $0.3\le x\le 0.5$

$$J=I_y(b,a)<I_y(1,1)=1-x\le 0.7$$

For (19): $a\le 1$, $b>15$, $0.1\le x<0.3$

$$J=I_y(b,a)\le I_y(b,1)=y^b=(1-x)^b\le (0.9)^{15}\approx 0.2059$$

For (19.1): $a\le 1$, $b>15$, $x<0.1$, $(bx)^a>0.7$

Then

$$J=I_y(b,a)\le I_y(b,1)=y^b=(1-x)^b.$$

From(1), using $(1-t)^{b-1}\ge (1-x)^{b-1}$ and recalling F(a,b) in the proof of (17.3),

$$I_x(a,b)\ge \frac{F(a,b)}{\Gamma(a)}\frac{(1-x)^b}{1-x}\frac{(bx)^a}{a}\ge F(a,b)(bx)^a J/\Gamma(a+1).$$

Now $\partial F/\partial b=g(a,b)F$ where $g(a,b)=\psi(a+b)-\psi(b)-a/b$. Since $g_{aa}<0$ [1; 6.4.10], g is strictly concave for $0<a<1$. Thus, since $g(0,b)=g(1,b)=0$ [1; 6.3.5], $g(a,b)>0$ for $0<a<1$ and $\partial F/\partial b>0$. Hence, F is increasing in b for $0<a<1$ and

$$F(a,b)>F(a,1)=\Gamma(a+1).$$

Thus $I_x(a,b)\ge 0.7J$ for $0\le a\le 1$, which implies that $J\le 1/1.7\approx .59$.

For (20): $a \leq 1, \ 1 < b \leq 15, \ 0.1 \leq x < 0.3$

$$J = I_y(b,a) \leq I_y(b,1) = y^b = (1-x)^b \leq (0.9)^b \leq 0.9$$

For (20.1): $a \leq 1, \ 1 < b \leq 15, \ x < 0.1, \ (bx)^a > 0.7$

Proof is the same as that given for (19.1).

For (20.2): $b \leq 1, \ x < 0.3, \ a < \min(.2, b), \ x^a > 0.9$

$$I_x(a,b) \geq I_x(a,a) = K(a) \frac{a^2}{2a} \int_0^x t^{a-1} (1-t)^{a-1} dt,$$

where

$$K(a) = \Gamma(2a+1) / [\Gamma(a+1)\Gamma(a+1)], \quad (0 \leq a \leq 1).$$

From (i), using

$$(1-t)^{a-1} \geq 1, \qquad 0 \leq t \leq x,$$

it follows that

$$I_x(a,b) > K(a)\, x^a / 2 > 0.45\, K(a).$$

Since $\partial K(a)/\partial a = 2\,K(a)[\psi(2a+1) - \psi(a+1)] > 0$, $K(a)$ is an increasing function. Hence $K(a) \geq 1$ and

$$J = I_y(b,a) \leq 0.55.$$

For (21)–(26): $a > 1, \quad b > 1$

In this case it can be shown that if $x \leq p$ then

$$I_x(a,b) \leq \begin{cases} 1 - 1/e & a < b \\ 1/2 & a \geq b. \end{cases} \tag{27}$$

A proof of this result is given in Appendix A. Hence

$$J = \begin{cases} I_x(a,b) & x \leq p \\ I_y(b,a) & x > p. \end{cases} \tag{28}$$

Notation for the flowchart in Fig.1

12 Refers to (12) for computing $I_x(a,b)$ (BPSER)

$\overline{12}$ Refers to (12) for computing $I_y(b,a)$ (BPSER)

13 Refers to (13) for computing $I_x(a,b)$ (BUP)

$\overline{13}$ Refers to (13) for computing $I_y(b,a)$ (BUP)

14 Refers to (14) for computing $I_x(a,b)$ (BGRAT)

$\overline{14}$ Refers to (14) for computing $I_y(b,a)$ (BGRAT)

15 Refers to (15) for computing $I_x(a,b)$ (BFRAC)

$\overline{15}$ Refers to (15) for computing $I_y(b,a)$ (BFRAC)

16 Refers to (16) for computing $I_x(a,b)$ (BASYM)

$\overline{16}$ Refers to (16) for computing $I_y(b,a)$ (BASYM)

$p=a/(a+b)$, $q=b/(a+b)$

I.C.$\Rightarrow$Interchange a and b; x and y.

$[b]\equiv$largest integer $< b$.

$\lambda\equiv a-(a+b)x$, $a\leq b$; $\lambda\equiv(a+b)y-b$, $a>b$.

Numbers above some of the flowchart boxes refer to the labels in the Fortran listing of BRATIO.

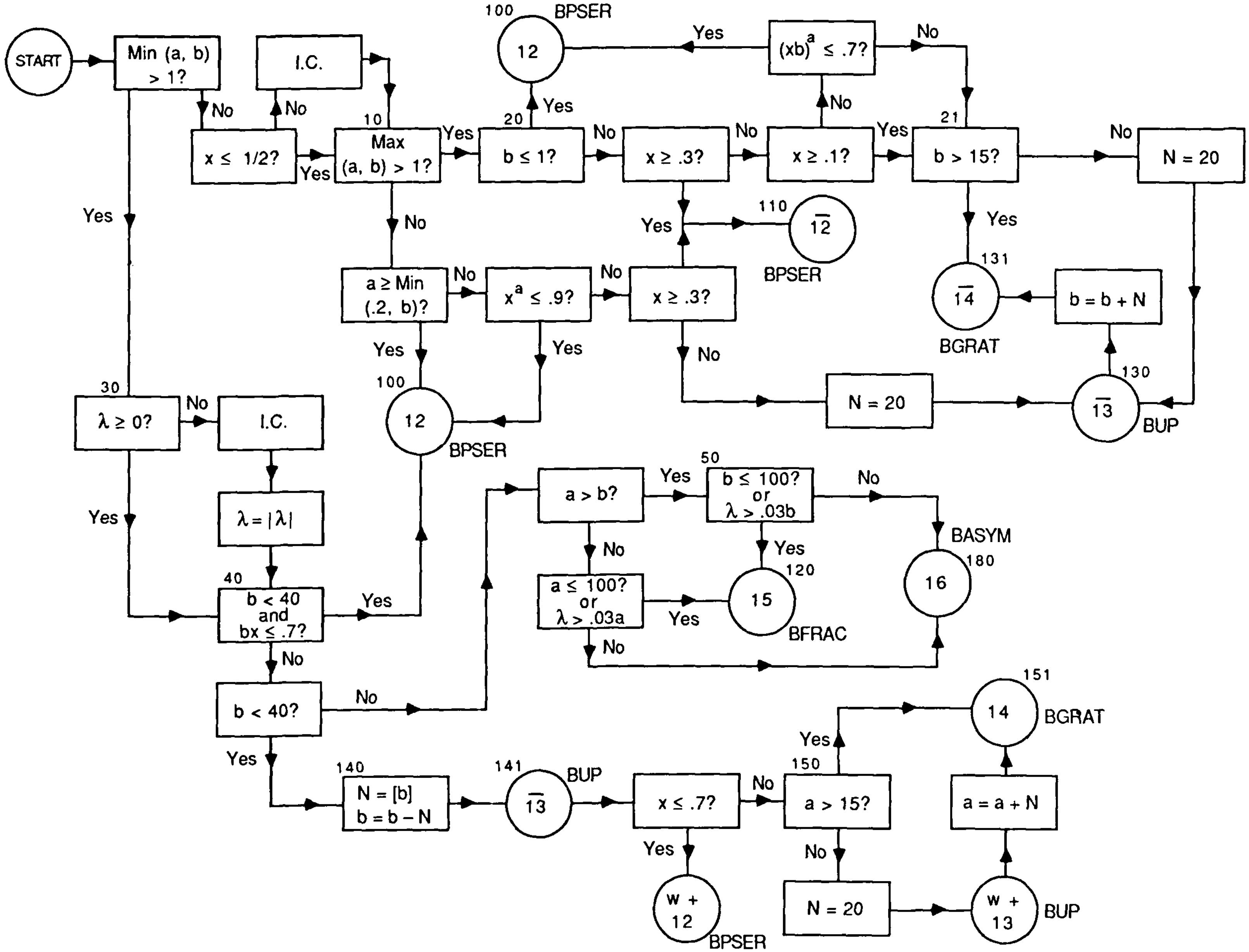

Fig. 1 Flowchart for BRATIO (Notation on p. 16)

IV. AUXILIARY FUNCTIONS

In order to compute $I_x(a,b)$ and $1-I_x(a,b)$, procedures are needed for evaluating $\Gamma(a)$, $\ln\Gamma(a)$, the error function erf x, $\exp(x^2)\,\mathrm{erfc}\,x$, the incomplete gamma function $Q(a,x)$, (14.3), for $a\leq 1$, and the functions

$$
\begin{array}{ll}
e^x-1 & \\
\ln(1+x) & (|x|\leq .375) \\
\ln\Gamma(1+x) & (-0.2\leq a\leq 1.25) \\
1/\Gamma(1+a)-1 & (-0.5\leq a\leq 1.5) \\
e^{-x}x^a/\Gamma(a) & (a>0,\ \ x\geq 0) \\
\phi(x)=x-1-\ln x & (x>0).
\end{array}
\tag{29}
$$

These functions are discussed in [6]. Also, procedures are needed for computing

$$
\begin{array}{ll}
\Delta(a)=\ln\,\Gamma(a)-(a-.5)\ln\,a+a-.5\,\ln(2\pi) & (a\geq 8) \\
\mathrm{ALGDIV}(a,b)=\ln[\Gamma(b)/\Gamma(a+b)], & (a\geq 0, b\geq 8) \\
\mathrm{BCORR}(a,b)=\Delta(a)+\Delta(b)-\Delta(a+b) & (a,b\geq 8) \\
\mathrm{BETALN}(a,b)=\ln\,B(a,b) & (a,b>0) \\
\mathrm{BRCOMP}(a,b,x,y)=x^a y^b/B(a,b) & (a,b>0,\ \ 0<x<1,\ \ y=1-x).
\end{array}
\tag{30}
$$

Rational minimax approximations are used for the functions given in (29). Experience indicates that such approximations normally generate less error and can be considerably more efficient than the standard expansions. However, minimax approximations have the disadvantage of being limited to a fixed maximum precision. The mimimax approximations used are designed to achieve a maximum precision of 14 significant digits.

If $\Delta(a)$ is needed only for $a\geq 20$, then the sum

$$1/(12a)-1/(360a^3)+\cdots$$

in the asymptotic expansion of $\ln\Gamma(a)$ [1; 6.1.41] may be used. If $a\geq 15$ then the minimax approximation

$$\Delta(a)=\sum_{n=0}^{4} c_n/a^{2n+1}$$

$c_0=\ .83333\ 33333\ 33333E-01$

$c_1=-.27777\ 77777\ 70481E-02$

$c_2=\ .79365\ 06631\ 83693E-03$

$c_3=-.59515\ 63364\ 28591E-03$

$c_4=\ .82075\ 63703\ 53826E-03$

can be applied, and if $a \geq 8$ the minimax approximation

$$\Delta(a)=\sum_{n=0}^{5} d_n/a^{2n+1}$$

$$d_0= \ .83333\ 33333\ 33333E-01 \tag{32}$$

$$d_1=-.27777\ 77777\ 60991E-02$$

$$d_2= \ .79365\ 06668\ 25390E-03$$

$$d_3=-.59520\ 29313\ 51870E-03$$

$$d_4= \ .83730\ 80340\ 31215E-03$$

$$d_5=-.16532\ 29627\ 80713E-02$$

can be used. These approximations were obtained by Morris [9]. On the CDC 6000-7000 series computers, they are accurate to within 1 unit of the 14^{th} significant digit.

Expansions for ALGDIV(a,b) and BCORR(a,b) use (32). From the definition of Δ

$$\text{ALGDIV}(a,b)=w-(a+b-.5)\ln(1+a/b)-a(\ln b-1) \tag{33}$$

$$w=\Delta(b)-\Delta(a+b).$$

Let

$$p=a/(a+b),\quad q=b/(a+b),\quad S_m=1+q+\cdots+q^{m-1}\quad (m\geq 1).$$

Then

$$1-q^m=(1-q)S_m=pS_m,\qquad \frac{pS_m}{b^m}=\frac{1}{b^m}-\frac{1}{(a+b)^m}. \tag{34}$$

Thus, from (32) we obtain

$$w=\frac{p}{b}\sum_{n=0}^{5} d_n\frac{S_{2n+1}}{b^{2n}}, \tag{35}$$

which completes the algorithm for ALGDIV(a,b). Also

$$\text{BCORR}(a,b)=\Delta(a_0)+[\Delta(b_0)-\Delta(a_0+b_0)]$$

$$a_0=\min\{a,b\},\quad b_0=\max\{a,b\},$$

where (32) and (35) are applied.

If $a \leq b$, then BETALN(a,b) can be accurately computed when $a \geq 1$. If $a \geq 8$ then

$$\text{BETALN}(a,b)=(.5\ln(2\pi)-.5\ln b)+\text{BCORR}(a,b)-u-v$$

$$u=-(a-.5)\ln\left[a/(a+b)\right],\qquad v=b\ln(1+a/b)$$

is applied. If $2<a<8$ then a is reduced to the interval [1,2] by

$$B(a,b)=\frac{a-1}{a+b-1}B(a-1,b).$$

Consequently, it can be assumed that $a\le 2$.

If $b\ge 8$ then

$$\ln B(a,b)=\ln \Gamma(a)+\mathrm{ALGDIV}(a,b)$$

is applied. If $2<b<8$ then b is also reduced to the interval [1,2] when $a\ge 1$. Thus, we need only consider the cases: $1\le a\le 2$, $1\le b\le 2$, or $a<1$ and $b<8$. If $a\ge 1$ then

$$\ln B(a,b)=\ln \Gamma(a)+\ln \Gamma(b)-\ln \Gamma(a+b)$$

is appropriate. No loss of accuracy due to subtraction can occur since $\ln \Gamma(a)$, $\ln \Gamma(b)$, and $-\ln \Gamma(a+b)$ are nonpositive. However, subtraction does occur when $a<1$. It currently is not clear how loss of accuracy due to subtraction can be avoided when $a<1$. Therefore, in this case BETALN is not used in BRATIO.

If $\min\{a,b\}<8$ then BRCOMP(a,b,x,y) can be computed directly from its definition. Otherwise,

$$\mathrm{BRCOMP}(a,b,x,y)=\sqrt{\frac{ab}{2\pi(a+b)}}\, e^{-z}$$

$$z=\left[a\phi(1-\lambda/a)+b\phi(1+\lambda/b)\right]+\mathrm{BCORR}(a,b)$$

is used, where λ is given in (15.1).

V. CONCLUDING REMARKS

Formulas (12), (13), (14), (15), and (16) for $I_x(a,b)$ are of the form γS, where S is a series. For example, for (15) $\gamma=x^a y^b/B(a,b)$ and S is a continued fraction. On the CDC 6000−7000 series computers, almost no error is generated in computing the series S over the domains specified in section III. The series is normally accurate to within 1 or 2 units of the 14^{th} significant digit. However, the precision of the factor γ is restricted by the inherent error of $I_x(a,b)$. Extensive testing on the CDC 6000−7000 series computers comparing the results obtained by BRATIO with results from double precision code, indicates that the precisions of the values obtained for $I_x(a,b)$ and $1-I_x(a,b)$ by BRATIO approximate the inherent errors of these functions up to a maximum of 14 significant digits. On any computer, accuracy is restricted to 14 digits because of the algorithms used for the auxiliary functions in section IV.

On the CDC 6000−7000 series computers, a maximum of 7 terms of the series (14) for BGRAT, and a maximum of 11 terms of (16) for BASYM were observed for the domains specified in section III. Frequently, 40 or fewer terms of (12) for BPSER suffice, but a maximum of 92 terms has been observed when a is small, b is large, and $x \approx .3$. Also, 40 or fewer terms generally suffice for the continued fraction (15), BFRAC, but a maximum of 58 terms has been observed when a or b is exceedingly large and $x \approx a/(a+b)$.

In practice, BRATIO has been found to be a reliable and efficient subroutine. As was noted in the previous sections, in order to develop such a subroutine new formulas were needed for $I_x(a,b)$, a surprisingly elaborate specification of the domains of usage for the various formulas had to be given, and a number of auxiliary functions had to be treated with extreme care. Thus, the development of BRATIO for efficiently computing $I_x(a,b)$ and $1-I_x(a,b)$ to high relative accuracy was not a simple task.

VI. NUMERICAL EXAMPLES

A collection of 16 examples using BRATIO is given in Fig.2. The results were obtained using the CDC 6000-7000 series single precision truncation floating arithmetic. As was noted in section III, the function BUP is used only with BPSER or BGRAT and appears in (20), (22), (23), and (24). The following three cases illustrate its use. The quantity ϵ below is set to approximately .710E-14 (see p.11).

<u>Case (1)</u>. a=.10, b=14.5, x=.29, y=.71

From (20) $\quad w_0 = \text{BUP}(14.5, .10, .71, .29, 20, \epsilon)$

$$= I_{.71}(14.5, .10) - I_{.71}(34.5, .10)$$

$$= .17776\ 09989\ 0838E{-}3.$$

Hence, if w is assigned the initial value w_0 then a call to BGRAT(34.5, .10, .71, .29, w, 15ϵ, IERR) yields the value

$$w = w_0 + I_{.71}(34.5, .10) = I_{.71}(14.5, .10)$$

$$= .17785\ 31648\ 7898E{-}3.$$

Also

$$I_{.29}(.10, 14.5) = .99982\ 21468\ 3512.$$

<u>Case (2)</u>. a=1.5, b=20.5, x=.065, y=.935

(Note that bx>.70, $\lambda = a-(a+b)x = .07 > 0$)

From(22) $\quad w_0 = \text{BUP}(.50, 1.5, .935, .065, 20, \epsilon)$

$$= I_{.935}(.50, 1.5) - I_{.935}(20.5, 1.5)$$

$$= .56745\ 07805\ 9439.$$

Then

$$I_{.065}(1.5, 20.5) = w_0 + \text{BPSER}(1.5, .50, .065)$$

$$= w_0 + .71754\ 32115\ 7741E{-}3$$

$$= .57462\ 62127\ 1016,$$

$$I_{.935}(20.5, 1.5) = .42537\ 37872\ 8984.$$

<u>Case (3)</u>. a=10.5, b=1.5, x=.80, y=.20, (λ>0).

From (24) $\quad \overline{w}_0 = \text{BUP}(.50, 10.5, .20, .80, 1, \epsilon)$

$$= I_{.20}(.50, 10.5) - I_{.20}(1.5, 10.5)$$

$$= .15518\ 20005\ 6352,$$

$$w_0 = \overline{w}_0 + \text{BUP}(10.5, .50, .80, .20, 20, \epsilon)$$

$$= \overline{w}_0 + I_{.80}(10.5, .50) - I_{.80}(30.5, .50)$$

$$= \overline{w}_0 + .32149\ 19363\ 8971E{-}1$$

$$= .18733\ 11942\ 0245.$$

Hence, if w is assigned the initial value w_0 then a call to BGRAT(30.5, .50, .80, .20, w, 15ϵ ,IERR) yields the value

$$w = w_0 + I_{.80}(30.5, .50)$$

or

$$I_{.80}(10.5, 1.5) = .18756\ 94122\ 3880.$$

Also,

$$I_{.20}(1.5, 10.5) = .81243\ 05877\ 6120.$$

a	b	x	y	$I_x(a,b)$	$1-I_x(a,b)$
.1	.8	.40	.60	.88776 70523 5302E+00	.11223 29476 4698E+00
.1	.8	.60	.40	.92957 83432 6833E+00	.70421 65673 1668E-01
.1	2.3	.40	.60	.97448 97683 7361E+00	.25510 23162 6386E-01
.1	2.3	.60	.40	.99196 58486 2884E+00	.80341 51371 1598E-02
5.0	40.0	.99	.01	.10000 00000 0000E+01	.13053 04681 1410E-74
5.0	10.0	.99	.01	.10000 00000 0000E+01	.96509 74271 4997E-17
10.0	38.0	.02	.98	.26944 43561 3309E-07	.99999 99730 5556E+00
70.0	10.0	.85	.15	.23472 44941 6827E+00	.76527 55058 3173E+00
* 70.0	50.0	.99	.01	.10000 00000 0000E+01	.54279 07073 1686E-66
* 70.0	50.0	.10	.90	.47438 77486 2163E-38	.10000 00000 0000E+01
* 75.0	50.0	.10	.90	.61550 21193 1591E-42	.10000 00000 0000E+01
* 500.0	501.0	.60	.40	.99999 99999 3299E+00	.67009 77013 4757E-10
500.0	501.0	.40	.60	.10148 03038 4399E-09	.99999 99998 9852E+00
1000.0	1001.0	.49	.51	.19153 11043 9543E+00	.80846 88956 0457E+00
1001.0	1000.0	.49	.51	.17957 42144 6754E+00	.82042 57855 3246E+00

(a=5.0E+20, b=5 9E+3, x=1.0, y=1.0E-17
$I_x(a,b)$=.49811 93659 6617, $I_y(b,a)$=.50188 06340 3383)

Due to inherent error, the 4 starred examples are correct to within 1 unit of the 12th significant digit. All other cases are correct to within 5 units of the 14th significant digit.

Fig. 2 16 Examples of BRATIO

REFERENCES

1. Abramowitz,M. and Stegun,I.(EDS.), *Handbook of Mathematical Functions,* Dover, N.Y., 1965.
2. Amos, D.E. and Daniel, S.L., Significant Digit Incomplete Beta Ratios Development, Report SC-DR-69-591, Sandia Laboratories, Albuquerque, N.M., 1969.
3. Aroian, L.A, Continued Fractions for the Incomplete Beta Function, *Am. Math. Statist.*, 12, 1941, 218-223.
4. DiDonato, A.R. and Jarnagin, M.P. Jr., A Method for Computing the Incomplete Beta Function Ratio, NWL Report No. 1949, U.S. Naval Weapons Laboratory, Dahlgren, Va. 22448, 1966.
5. DiDonato, A.R. and Jarnagin, M.P., The Efficient Calculation of the Incomplete Beta Function Ratio for Half - Integer Values of the Parameters a,b, *Math. of Comp.*, 21, 1967, 652-662.
6. DiDonato, A.R. and Morris, A.H., Computation of the Incomplete Gamma Function Ratios and Their Inverse, *ACM Trans. of Math. Software,* 12, 1986, 377-393.
7. Dutka, J., The Incomplete Beta Function - A Historical Profile, *Archive for History of Exact Sciences,* 24, 1981, 11-29.
8. Henrici P., *Applied and Computational Complex Analysis (Vol. 1),* John Wiley and Sons, 1974.
9. Morris A.H., NSWC Library of Mathematical Subroutines, Rep. NSWC TR 86-251. Naval Surface Weapons Center, Dahlgren, Va. 22448, 1987.
10. Olver F.W.J., *Asymptotics and Special Functions,* Academic Press, 1974.
11. Pourahmadi, M., Taylor Expansion of $\text{Exp}(\sum_{k=0}^{\infty} a_k z^k)$ and Some Applications, *American Math Monthly*, 1984, 303-307.
12. Temme, N.M., Uniform Asymptotic Expansions of the Incomplete Gamma Functions and the Incomplete Beta Function, *Math. of Comp.* 29, 1975, 1109-1114.
13. Tretter, M.J. and Walster, G.W., Continued Fractions for the Incomplete Beta Function: Additions and Corrections, *Ann. of Stat.*, 7, 1979, 462-465.
14. Wall, H.S., *Analytic Theory of Continued Fractions.* Van Nostrand Co., N.Y., 1948.
15. Wise, M.E., The Incomplete Beta Function as a Contour Integral and a Quickly Converging Series for Its Inverse, *Biometrika,* 37, 1950, 208-218.

APPENDIX A

BOUNDS ON $I_x(a,b)$ WHEN $x=a/(a+b)$, $MIN\{a, b\} \geq 1$

BOUNDS ON $I_x(a,b)$ WHEN $x=a/(a+b)$, $\text{MIN}\{a,b\} \geq 1$

The purpose of this appendix is to show that

$$\begin{aligned} &1/2 \leq I_p(a,b) < 1-e^{-1} \qquad \text{if } 1 \leq a \leq b \\ &1/e < I_p(a,b) \leq 1/2 \qquad \text{if } 1 \leq b \leq a \end{aligned} \tag{A-1}$$

where $p=a/(a+b)$ and $q=1-p=b/(a+b)$. Since $I_{.5}(a,a)=.5$ from (4), we shall assume that $a \neq b$.

Lemma (A-1): If $\lambda(h)=(1+1/h)^{-h}$ for $h>0$ then λ is a decreasing function. Also, $\lambda \to 1$ when $h \to 0$ and $\lambda \to 1/e$ when $h \to \infty$.

The lemma is given for reference. It is a well known result.

Corollary (A-1): If $a>1$ then $1/e < I_p(a,1) < 1/2$.

Proof: From (1) we obtain $I_p(a,1)=p^a=(1+1/a)^{-a}$. Then from the lemma the corollary follows.

Corollary (A-2): If $b>1$ then $1/2 < I_p(1,b) < 1-1/e$.

Proof: Immediate from (4) and corollary (A-1).

Since the above corollaries hold, in order to verify (A-1) it suffices to assume that a, b>1. The following reasoning is due to James C. Perry (NSWC). Let

$$B \equiv \int_0^1 t^{a-1}(1-t)^{b-1}dt \tag{A-2}$$

$$B_p \equiv \int_0^p t^{a-1}(1-t)^{b-1}dt \tag{A-3}$$

$$\overline{B}p \equiv \int_p^1 t^{a-1}(1-t)^{b-1}dt. \tag{A-4}$$

If $x=(1-t)^{b/a}$ then

$$B_p=\frac{a}{b}\int_\lambda^1 \left(x-x^{1+a/b}\right)^{a-1}dx \tag{A-5}$$

$$\overline{B}_p=\frac{a}{b}\int_0^\lambda \left(x-x^{1+a/b}\right)^{a-1}dx \tag{A-6}$$

where

$$\lambda=q^{b/a}=(1+a/b)^{-b/a}. \tag{A-7}$$

Also $1/e < \lambda < 1$ by lemma (A-1), and we now consider the functions

$$f(x)=x-x^{1+a/b} \text{ and } F(x)=f(x)^{a-1} \tag{A-8}$$

for $0 \leq x \leq 1$.

From (A-7) and (A-8) we note that

$f(0) = F(0) = 0, \quad f(1) = F(1) = 0,$

$f(x)$ and $F(x)$ have a unique maximum at $x = \lambda$, and

$f'(x)$ and $F'(x)$ are positive (negative) for $0 < x < \lambda$ ($\lambda < x < 1$).

For any x_2 such that $\lambda < x_2 \leq 1$ let

$$r(x_2) = \frac{x_2 - \lambda}{\lambda - x_1} \tag{A-9}$$

where x_1 is the unique value in $[0,\lambda)$ where $f(x_1) = f(x_2)$ [see Fig. 3].

We now examine $r(x_2)$.

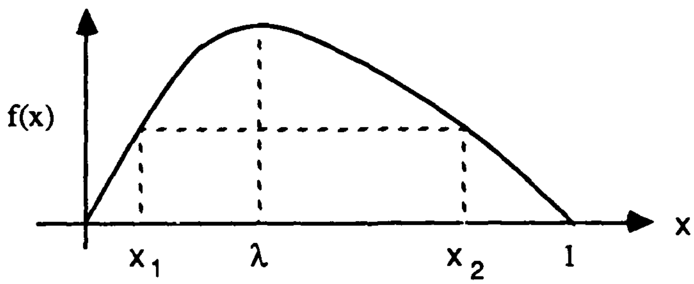

Fig. 3 Graph of f(x)

Lemma (A-2): If $g(x)$ is twice continuously differentiable on $[0,d]$, $g(0) = g'(0) = 0$, and g'' is increasing (decreasing) on $(0,d]$, then h is continuous and increasing (decreasing) on $[0,d]$ where

$$h(x) \equiv \begin{cases} g(x)/x^2 & (x>0) \\ g''(0)/2 & (x=0). \end{cases}$$

Proof: Assume that g'' is increasing on $(0,d]$. Since $h'(x) = (xg' - 2g)/x^3$ for $x > 0$, let $k(x) = xg' - 2g$ for $x \geq 0$. If $x > 0$ then there exists $0 < \xi < x$ such that $g'(x) = g''(\xi)x$ by the Mean-value theorem, so that

$$k'(x) = xg''(x) - g'(x) = x[g''(x) - g''(\xi)] > 0.$$

Therefore, $k(x)$ is increasing for $x > 0$. Since $k(0) = 0$, $h'(x) = k(x)/x^3 > 0$, which proves that h is increasing on $(0,d]$. Also, since $g(0) = g'(0) = 0$, $h(x) \rightarrow g''(0)/2$ when $x \rightarrow 0$ by L'Hopital's rule. If $g''(x)$ is decreasing on $(0,d]$ then apply the lemma to $-g$.

Theorem (A-1): Let $r(x)$ denote the function defined by (A-9). If $a < b$ ($a > b$) then $r(x)$ is increasing (decreasing) for $\lambda < x \leq 1$. Also, $r \rightarrow 1$ when $x \rightarrow \lambda$.

Proof: If $\phi(h) = f(\lambda) - f(\lambda + h)$ for $0 \leq h \leq 1 - \lambda$ then $\phi(0) = 0$, $\phi'(0) = -f'(\lambda) = 0$, and $\phi''(h) = (a/bq)(\lambda + h)^{-1+a/b}$ for $h \geq 0$. Hence, if $a<b$ ($a>b$) then ϕ'' is decreasing (increasing), so that

$\bar{\phi}(h)=\phi(h)/h^2$ is decreasing (increasing) by lemma (A-2). Also, $\bar{\phi}(h)\to\phi''(0)/2=a/(2b\lambda)$ when $h\to 0$ by the lemma and $\lambda=q^{b/a}$.

Similarly, if $\psi(h)=f(\lambda)-f(\lambda-h)$ for $0\le h\le\lambda$ then $\bar{\psi}(h)=\psi(h)/h^2$ is increasing (decreasing) when $a<b$ ($a>b$), and $\bar{\psi}(h)\to\psi''(0)/2=a/(2b\lambda)$ when $h\to 0$.

Now if $\lambda<x\le 1$ and $x_1<\lambda$ where $f(x_1)=f(x)$, then

$$r^2=\left(\frac{x-\lambda}{\lambda-x_1}\right)^2=\frac{f(\lambda)-f(x_1)}{(\lambda-x_1)^2}\,\frac{(x-\lambda)^2}{f(\lambda)-f(x)}=\frac{\bar{\psi}(\lambda-x_1)}{\bar{\phi}(x-\lambda)}$$

so that $r^2\to\dfrac{a/(2b\lambda)}{a/(2b\lambda)}=1$ when $x\to\lambda$. If $a<b$ then when x increases, $\lambda-x_1$ increases, $\bar{\psi}(\lambda-x_1)$ increases, $\bar{\phi}(x-\lambda)$ decreases, and hence r^2 increases. Similarly, r^2 decreases when $a>b$. Consequently, since $r>0$ the theorem follows.

Theorem (A-2): If $1\le a<b$ then $I_p(a,b)<1-1/e$.

Proof: Since $a<b$, $r(x)\le r(1)=(1-\lambda)/\lambda$ for $\lambda<x\le 1$ by theorem (A-1). Also, since $I_p(1,b)<1-1/e$ by corollary (A-2), we shall assume that $a>1$. Now let

$$t=\frac{\lambda}{1-\lambda}(1-x)\qquad \lambda\le x\le 1 \tag{A-10}$$

so that $0\le t\le\lambda$. For $x>\lambda$ let $x_1<\lambda$ where $f(x_1)=f(x)$. Then $t<\lambda$, and from

$r(x)=\dfrac{x-\lambda}{\lambda-x_1}\le\dfrac{1-\lambda}{\lambda}$ we have $x_1\le t$. Therefore,

$$F\left(1-\frac{1-\lambda}{\lambda}t\right)=F(x)=F(x_1)\le F(t)$$

so that

$$B_p=\frac{a}{b}\int_\lambda^1 F(x)dx=\frac{a}{b}\,\frac{1-\lambda}{\lambda}\int_0^\lambda F\left(1-\frac{1-\lambda}{\lambda}t\right)dt$$

$$\le\frac{a}{b}\,\frac{1-\lambda}{\lambda}\int_0^\lambda F(t)dt=\left(\frac{1}{\lambda}-1\right)\bar{B}_p<(e-1)\bar{B}_p$$

from (A-5) and (A-6). Therefore, from (A-2)$-$(A-4), $B_p<(e-1)\bar{B}_p=(e-1)(B-B_p)$ so that $I_p(a,b)=B_p/B<1-1/e$.

Theorem (A-3): If $1\le b<a$ then $I_p(a,b)<1/2$.

Proof: Since $a>b$, $r(x)<1$ for $\lambda<x\le 1$ by theorem (A-1). Also, since $I_p(a,1)<1/2$ by corollary (A-1), we shall assume that $b>1$. Now let

$$t=2\lambda-x\qquad \lambda\le x\le 1. \tag{A-11}$$

Since $b/a<1$, $.5<\lambda<1$ from (A-7) and lemma (A-1). Therefore, $0<2\lambda-1<\lambda$ and $2\lambda-1\le t\le\lambda$.

For $x > \lambda$ let $x_1 < \lambda$ where $f(x_1) = f(x)$. Then $t < \lambda$, and from $r(x) = \frac{x-\lambda}{\lambda - x_1} < 1$ we have $x_1 < t$. Therefore,

$$F(2\lambda - t) = F(x) = F(x_1) < F(t)$$

so that

$$B_P = \frac{a}{b}\int_\lambda^1 F(x)dx = \frac{a}{b}\int_{2\lambda-1}^{\lambda} F(2\lambda - t)dt$$

$$< \frac{a}{b}\int_{2\lambda-1}^{\lambda} F(t)dt \leq \frac{a}{b}\int_0^\lambda F(t)dt = \overline{B}_P$$

from (A-5) and (A-6). Therefore, from (A-2)$-$(A-4) $B_P < \overline{B}_P = B - B_P$ so that $I_P(a,b) = B_P/B < 1/2$.

Theorem (A-4): If $a,b \geq 1$ then

$$1/2 < I_P(a,b) < 1 - 1/e \qquad \text{when } a < b$$

$$1/e < I_P(a,b) < 1/2 \qquad \text{when } a > b.$$

Proof: Immediate from (4) and theorems (A-2) and (A-3).

APPENDIX B

MONOTONIC PROPERTIES OF $I_x(a,b)$

MONOTONIC PROPERTIES OF $I_x(a,b)$

The purpose of this appendix is to show that $I_x(a,b)$ is a decreasing function of a and an increasing function of b.

<u>Theorem B</u>: If $a>0$, $b>0$, and $0 \le x \le 1$, then $\dfrac{\partial I_x(a,b)}{\partial a} \le 0$.

<u>Proof</u>: Let

$$f(t;a,b) \equiv t^{a-1}(1-t)^{b-1},$$

hereafter simply denoted by $f(t)$. We have

$$I_x(a,b) \equiv B_x(a,b)/B(a,b)$$

$$B_x(a,b) \equiv \int_0^x t^{a-1}(1-t)^{b-1}\,dt, \qquad B(a,b) \equiv \int_0^1 t^{a-1}(1-t)^{b-1}\,dt.$$

Then

$$\frac{\partial I_x(a,b)}{\partial a} = \frac{B(a,b)\int_0^x u^{a-1}(1-u)^{b-1}\ln u\,du - B_x(a,b)\int_0^1 t^{a-1}(1-t)^{b-1}\ln t\,dt}{B(a,b)\,B(a,b)}$$

$$\left[B(a,b)\right]^2 \frac{\partial I_x(a,b)}{\partial a} = \int_0^1 \left[\int_0^x t^{a-1}(1-t)^{b-1}\; u^{a-1}(1-u)^{b-1}(\ln u - \ln t)\,du\right]dt$$

$$= \int_0^x \int_0^x \left[f(t)f(u)(\ln u - \ln t)\,du\right]dt + \int_x^1 \int_0^x \left[f(t)f(u)(\ln u - \ln t)\,du\right]dt.$$

But the first double integral on the right is zero. Indeed, with

$$T_{u,t} \equiv \int_0^x \int_0^x \left[f(t)f(u)(\ln u - \ln t)\,du\right]dt,$$

where the subscripts are used to indicate the order of integration (u then t), we have, by renaming the variables,

$$T_{u,t} = T_{t,u}.$$

Interchanging the order of integration then gives

$$T_{u,t} = T_{t,u} = -T_{u,t},$$

which implies that $T_{u,t}=0$. Therefore,

$$\left[B(a,b)\right]^2 \frac{\partial I_x(a,b)}{\partial a} = \int_x^1 \int_0^x f(t)\,f(u)(\ln u - \ln t)\,du\,dt, \tag{B-1}$$

which is negative since $\ln u - \ln t \le 0$ for $u \le t$. Thus,

$$\frac{\partial I_x(a,b)}{\partial a} \le 0 \text{ for all } a>0, \quad b>0, \quad 0 \le x \le 1.$$

Remarks. A more direct proof of (B-1) can be obtained by differentiating $1/[1+I_y(b,a)/I_x(a,b)]$. Also, since $\ln u - \ln t \leq u - t \leq 0$ for $0 < u \leq t \leq 1$, from (B-1) we obtain the slightly stronger inequality

$$\frac{\partial I_x(a,b)}{\partial a} \leq \int_x^1 \int_0^x \left[f(t)f(u)(u-t)\,du\right]dt / \left[B(a,b)\right]^2$$

$$= \frac{B(a,b)\,B_x(a+1,b)}{\left[B(a,b)\right]^2} - \frac{B(a+1,b)\,B_x(a,b)}{\left[B(a,b)\right]^2}$$

$$= \frac{a}{a+b}\left[I_x(a+1,b) - I_x(a,b)\right].$$

Corollary: If $a_1 \leq a_2$ then $I_x(a_2,b) \leq I_x(a_1,b)$.

Corollary: If $b_1 \leq b_2$ then $I_x(a,b_1) \leq I_x(a,b_2)$.

Proof: Since $I_y(b,a)$ decreases as b increases, $I_x(a,b) = 1 - I_y(b,a)$ increases as b increases.

We close by noting that precisely the same technique used in Theorem B establishes a corresponding result for the incomplete gamma function. If

$$P(a,x) = \frac{1}{\Gamma(a)} \int_0^x e^{-t} t^{a-1} dt, \quad a > 0, \quad x \geq 0,$$

then $\dfrac{\partial P(a,x)}{\partial a} \leq 0$, and hence if $a_1 \leq a_2$ then $P(a_2,x) \leq P(a_1,x)$.

APPENDIX C

UNIFORM ASYMPTOTIC BEHAVIOR OF (16)

UNIFORM ASYMPTOTIC BEHAVIOR OF (16)

The purpose of this appendix is to show that (16) has uniform asymptotic behavior for $a \leq b$ and $a(1+a/b) \to \infty$. The proof will depend on showing that the n^{th} derivative of the function $f(v)=v/s$ $(v \geq 0)$ has a bound not depending on $h=a/b$ when $a \leq b$ and n is odd.

Theorem (C-1): **If $a \leq b$ then $0 \leq f'(v) < 1$ for $v \geq 0$.**

Proof: Since $\frac{ds}{dv}=\frac{v}{s}(1-s)(1+hs)$ and $f=\frac{v}{s}$ we have

$$f'(v)=\frac{df}{dv}=\frac{1}{s}-\frac{v}{s^2}\frac{ds}{dv}=\frac{1}{s}-\frac{v^2}{s^3}(1-s)(1+hs). \tag{C-1}$$

Also

$$\frac{v^2}{s^2}(1-s)(1+hs)=1-\sum_{n\geq 1}\Delta_n s^n \tag{C-2}$$

$$\frac{\Delta_n}{2q}=\frac{1+h^{n+1}}{(n+1)(n+2)}+\frac{h+h^n}{n(n+1)}>0 \qquad \text{n even} \tag{C-3}$$

$$\frac{\Delta_n}{2q}=\frac{1-h^{n+1}}{(n+1)(n+2)}+\frac{h-h^n}{n(n+1)}\geq 0 \qquad \text{n odd.}$$

Therefore $f'(v)=\sum_{n\geq 1}\Delta_n s^{n-1}>0$ is an increasing function for $s>0$ (and hence for $v>0$). From (C-2) $1-\sum_{n\geq 1}\Delta_n s^n>0$ for $0\leq s<1$, so that $\sum_{n\geq 1}\Delta_n\leq 1$. Consequently, $f'(v)=\sum_{n\geq 1}\Delta_n s^{n-1}<\sum_{n\geq 1}\Delta_n\leq 1$ for $v\geq 0$.

Theorem (C-2): **If $a\leq b$ then $f'(v)\to 1$ when $v\to\infty$.**

Proof: From (C-3) $\sum_{n\geq 1}\Delta_n/(2q)=(1+h)\sum_{n\geq 2}1/[n(n+1)]$, where

$$\sum_{n\geq 1}1/[n(n+1)]=\sum_{n\geq 1}\left[1/n-1/(n+1)\right]=1.$$

From (16.7) we have

$$v^2=\frac{2}{1+h}\left[-\ln(1-s)-\frac{1}{h}\ln(1+hs)\right]. \tag{C-4}$$

Consequently, we now consider v as a function defined for $0<h\leq 1$ and $0\leq s<1$.

Lemma (C-1): **If $n=0,1,2,\ldots$ then $v^n(1-s)$ can be extended to a continuous function $\phi_n(h,s)$ for $0\leq h\leq 1$ and $0\leq s\leq 1$. Also, $\phi_n(h,1)=0$ for $0\leq h\leq 1$.**

Proof: From (C-4) we obtain

$$v^{2n}(1-s)^2 = \frac{2^n}{(1+h)^n} \sum_{k=0}^{n} \binom{n}{k} \alpha_k(s)\, [s\,g(hs)]^{n-k}$$

$$\alpha_k(s) = [-\ln(1-s)]^k (1-s)^2 \qquad 0 \le s < 1$$

$$g(z) = -\ln(1+z)/z \qquad 0 < z \le 1.$$

By L'Hopital's rule $g(z) \to -1$ when $z \to 0$. Hence, if $\bar{g}(z) = g(z)$ for $0 < z \le 1$ and $\bar{g}(0) = -1$, then $\bar{g}(z)$ is continuous for $0 \le z \le 1$. Also if $w = -\ln(1-s)$ then

$$\lim_{s \to 1} \alpha_k(s) = \lim_{w \to \infty} w^k \exp(-2w) = 0.$$

Thus, if $\bar{\alpha}_k(s) = \alpha_k(s)$ for $0 \le s < 1$ and $\bar{\alpha}_k(1) = 0$, then

$$\phi_n(h,s) = \left[\frac{2^n}{(1+h)^n} \sum_{k=0}^{n} \binom{n}{k} \bar{\alpha}_k(s)\, [s\,g(hs)]^{n-k} \right]^{1/2}$$

is a continuous extension of $v^n(1-s)$ for $0 \le h \le 1$ and $0 \le s \le 1$. Also, $\phi_n(h,1) = 0$ since each $\bar{\alpha}_k(1) = 0$.

<u>Theorem (C-3):</u> If $n \ge 2$ then $f^{(n)}(v)$ can be extended to a function that is continuous for $0 \le h \le 1$ and $0 < s \le 1$. Also, $f^{(n)}(v) \to 0$ when $v \to \infty$.

<u>Proof:</u> Given a fixed value $0 < h \le 1$. Then for any polynomial P in s (and h), $k = 0,1,2,\ldots$, and $m = 1,2,\ldots$,

$$\frac{d}{dv}\left[\frac{v^k}{s^m}(1-s)P\right] = k\,\frac{v^{k-1}}{s^m}(1-s)\,P - m\,\frac{v^{k+1}}{s^{m+2}}(1-s)^2(1+hs)\,P$$
$$- \frac{v^{k+1}}{s^{m+1}}(1-s)(1+hs)\,P + \frac{v^{k+1}}{s^{m+1}}(1-s)^2(1+hs)\,P'$$

where P' is the derivative dP/ds. By induction (starting with C-1) it follows for $n \ge 2$ that $f^{(n)}(v)$ is a finite sum of terms of the form $(v^k/s^m)(1-s)P$. Consequently, by Lemma (C-1) $f^{(n)}(v)$ has a continuous extension $\psi_n(h,s)$ for $0 \le h \le 1$ and $0 < s \le 1$. Also, $\psi_n(h,1) = 0$ so that $f^{(n)}(v) \to 0$ when $v \to \infty$.

<u>Remark.</u> For $n \ge 2$, since $f(v) = \sum_{k \ge 0} d_k v^k$ in a neighborhood of 0 and $f^{(n)}(v) \to 0$ when $v \to \infty$, by Theorem (C-3) $f^{(n)}(v)$ is bounded for $v \ge 0$. However, the bound may depend on h. We now prove that a bound exists not depending on h when n is odd.

<u>Definition:</u> Given a series $\sum_{n \ge 0} \alpha_n(h) s^n$ where each $\alpha_n(h)$ is continuous for $0 \le h \le 1$. Let $M_n = \max\{\,|\alpha_n(h)| : 0 \le h \le 1\,\}$. Then the series $\sum \alpha_n s^n$ is said to satisfy condition (*) when $\sum_{n \ge 0} M_n s$ converges for $|s| < 1$.

Lemma (C-2): $f'(v)=\sum_{n\geq 1} \Delta_n s^{n-1}$ **satisfies condition (*).**

Proof: From (C-3) Δ_n is continuous for $0\leq h\leq 1$ and

$$|\Delta_n|\leq\frac{4}{(n+1)(n+2)}+\frac{4}{n(n+1)} .$$

Also $\sum_{n\geq 1} 1/[n(n+1)]=1$.

Theorem (C-4): For k odd, the series $\sum_{n\geq 0}\alpha_n(h)s^n$ for $f^{(k)}(v)$ satisfies condition (*).

Proof: By Lemma (C-2), the theorem is true for $k=1$. Assume inductively that it is true for odd k. If $e_n=\max\{\ |\alpha_n(h)|\ :\ 0\leq h\leq 1\ \}$ then let $g(s)=\sum_{n\geq 0} e_n s^n$. Now

$$f^{(k+1)}(v)=\Big(\sum_{n\geq 1} n\,\alpha_n\, s^{n-1}\Big)\,\tfrac{v}{s}(1-s)(1+hs)=\tfrac{v}{s}\sum_{n\geq 0}\beta_n(h)s^n \tag{C-5}$$

$$\beta_0=\alpha_1,\quad \beta_n=(n+1)\alpha_{n+1}-n(1-h)\alpha_n-h(n-1)\alpha_{n-1}\qquad (n\geq 1).$$

Since $\alpha_n(h)$ is continuous for $0\leq h\leq 1$, therefore $\beta_n(h)$ is continuous for $0\leq h\leq 1$. Also $|\beta_n|\leq M_n$ where $M_0=e_1$ and $M_n=(n+1)e_{n+1}+ne_n+(n-1)e_{n-1}$ $(n\geq 1)$. Since $\sum_{n\geq 0} M_n s^n$ is the series for $g'(s)(1+s+s^2)$, $\sum_{n\geq 0} M_n s^n$ converges for $|s|<1$ and $\sum_{n\geq 0}\beta_n s^n$ satisfies condition (*).

From (C-5) we obtain

$$f^{(k+2)}(v)=f'(v)\sum_{n\geq 0}\beta_n s^n+\sum_{n\geq 1} n\,\beta_n\, s^{n-1}\,\frac{v^2}{s^2}(1-s)(1+hs)$$

$$=\sum_{n\geq 1}\Delta_n s^{n-1}\sum_{n\geq 0}\beta_n s^n+\sum_{n\geq 1} n\,\beta_n\, s^{n-1}\Big(1-\sum_{n\geq 1}\Delta_n s^n\Big).$$

Consequently, $f^{(k+2)}(v)=\sum_{n\geq 0}\delta_n(h)s^n$ where

$$\delta_0=\beta_1+\beta_0\Delta_1$$

$$\delta_n=(n+1)\beta_{n+1}+\beta_0\,\Delta_{n+1}-\sum_{i=0}^{n-1} i\,\beta_{i+1}\,\Delta_{n-i}\quad (n\geq 1). \tag{C-6}$$

Then δ_n is continuous for $0\leq h\leq 1$. Also, if $c_n=\max\{\ \Delta_n\ :\ 0\leq h\leq 1\ \}$ then

$$|\delta_n|\leq (n+1)\,M_{n+1}+M_0\,c_{n+1}+N_n$$

$$N_n=\sum_{i=0}^{n-1} iM_{i+1}c_{n-i}\qquad (n\geq 1),\qquad N_0=0.$$

Now $\sum_{n\geq 0}(n+1)M_{n+1}s^n$ converges for $|s|<1$ since it is the derivative of $\sum_{n\geq 0} M_n s^n$, and $\sum_{n\geq 0} c_{n+1}s^n$ converges for $|s|<1$ by Lemma C-2. Also

$$\sum_{n\geq 0} N_n|s|^n = \sum_{n\geq 0} nM_{n+1}|s|^n \sum_{n\geq 0} c_n|s|^n \leq \sum_{n\geq 0}(n+1)M_{n+1}|s|^n \sum_{n\geq 0} c_n|s|^n,$$

which converges for $|s|<1$. Consequently, $f^{(k+2)}(v)=\sum_{n\geq 0}\delta_n s^n$ satisfies condition $(*)$.

<u>Theorem (C-5):</u> **If k is odd then $f^{(k)}(v)$ can be extended to a function that is continuous for $0\leq h\leq 1$ and $0\leq s\leq 1$. Consequently, $|f^{(k)}(v)|$ is bounded for $v\geq 0$ by a value that does not depend on h.**

<u>Proof:</u> It has already been verified that $f^{(k)}(v)$ can be extended to a function that is continuous for $0\leq h\leq 1$ and $0<s\leq 1$. Let $0<s_0<1$. Since $f^{(k)}(v)=\sum_{n\geq 0}\alpha_n(h)s^n$ satisfies condition $(*)$, $\sum_{n\geq 0} M_n {s_0}^n$ converges for $M_n=\max\{\,|\alpha_n(h)| : 0\leq h\leq 1\}$. Therefore, by the Weierstrass M-test, $\sum_{n\geq 0}\alpha_n s^n$ converges uniformly for $0\leq h\leq 1$ and $0\leq s\leq s_0$. Since each $\alpha_n(h)s^n$ is continuous for $0\leq h\leq 1$ and $0\leq s\leq s_0$, therefore $\sum_{n\geq 0}\alpha_n\, s^n$ is continuous for $0\leq h\leq 1$ and $0\leq s\leq s_0$.

APPENDIX D

FORTRAN LISTING OF BRATIO

FORTRAN LISTING OF BRATIO

In this appendix the code for BRATIO is given. The following functions are used.

$$\text{ERF}(x) = \text{erf } x$$

$$\text{ERFC1}(i,x) = \begin{cases} \text{erfc } x & \text{if } i=0 \\ \exp(x^2)\text{erfc } x & \text{if } i \neq 0 \end{cases}$$

$$\text{REXP}(x) = \exp(x) - 1$$

$\text{ALNREL}(a) = \ln(1+a)$	$a > -1$
$\text{RLOG1}(a) = a - \ln(1+a)$	$a > -1$
$\text{GAM1}(a) = 1/\Gamma(1+a) - 1$	$-.5 \le a \le 1.5$
$\text{GAMLN}(x) = \ln \Gamma(x)$	$x > 0$
$\text{GAMLN1}(a) = \ln \Gamma(1+a)$	$-.2 \le a \le 1.25$
$\text{ALGDIV}(a,b) = \ln[\Gamma(b)/\Gamma(a+b)]$	$a \ge 0, \quad b \ge 8$
$\text{BCORR}(a,b) = \Delta(a) + \Delta(b) - \Delta(a+b)$	$a,b \ge 8$
$\text{BETALN}(a,b) = \ln B(a,b)$	$a,b > 0$
$\text{BRCOMP}(a,b,x,y) = x^a y^b / B(a,b)$	$a,b > 0, \quad 0 < x < 1, \quad y = 1 - x$

These functions, written by A.H.Morris, are part of the NSWC mathematics subroutine library [9].

Machine - Dependent Constants

The function SPMPAR provides the machine-dependent constants needed by BRATIO. It is necessary that SPMPAR be properly defined for the computer arithmetic being used. The constants are defined in the in-line documentation of SPMAR. Values for these constants are given in the in-line documentation. SPMPAR, released by Argonne National Laboratory, is an adaptation of the Bell Laboratories function R1MACH [D1].

Transportability

All coding adheres to the 1966 and 1977 ANSI FORTRAN standards. It is assumed that a floating point arithmetic of 6 or more digits is being used. The codes were designed specifically for k-digit arithmetics where $6 \le k \le 14$. If $k > 14$ then only 14-digit accuracy will normally be obtained.

Reference

D1. Fox, P.A., Hall, A.D., and Schryer, N.L. The PORT Mathematical Subroutine Library, *ACM Trans. Math Software 4* (1978), pp. 104-126.

```
      SUBROUTINE BRATIO(A, B, X, Y, W, W1, IERR)
C-----------------------------------------------------------------------
C
C            EVALUATION OF THE INCOMPLETE BETA FUNCTION IX(A,B)
C
C                     --------------------
C
C     IT IS ASSUMED THAT A AND B ARE NONNEGATIVE, AND THAT X .LE. 1
C     AND Y = 1 - X.  BRATIO ASSIGNS W AND W1 THE VALUES
C
C                      W  = IX(A,B)
C                      W1 = 1 - IX(A,B)
C
C     IERR IS A VARIABLE THAT REPORTS THE STATUS OF THE RESULTS.
C     IF NO INPUT ERRORS ARE DETECTED THEN IERR IS SET TO 0 AND
C     W AND W1 ARE COMPUTED. OTHERWISE, IF AN ERROR IS DETECTED,
C     THEN W AND W1 ARE ASSIGNED THE VALUE 0 AND IERR IS SET TO
C     ONE OF THE FOLLOWING VALUES ...
C
C        IERR = 1  IF A OR B IS NEGATIVE
C        IERR = 2  IF A = B = 0
C        IERR = 3  IF X .LT. 0 OR X .GT. 1
C        IERR = 4  IF Y .LT. 0 OR Y .GT. 1
C        IERR = 5  IF X + Y .NE. 1
C        IERR = 6  IF X = A = 0
C        IERR = 7  IF Y = B = 0
C
C--------------------
C     WRITTEN BY ALFRED H. MORRIS, JR.
C        NAVAL SURFACE WEAPONS CENTER
C        DAHLGREN, VIRGINIA
C     REVISED ... JUNE 1988
C-----------------------------------------------------------------------
      REAL LAMBDA
C-----------------------------------------------------------------------
C     ****** EPS IS A MACHINE DEPENDENT CONSTANT. EPS IS THE SMALLEST
C            FLOATING POINT NUMBER FOR WHICH 1.0 + EPS .GT. 1.0
C
                       EPS = SPMPAR(1)
C-----------------------------------------------------------------------
      W = 0.0
      W1 = 0.0
      IF (A .LT. 0.0 .OR. B .LT. 0.0) GO TO 300
      IF (A .EQ. 0.0 .AND. B .EQ. 0.0) GO TO 310
      IF (X .LT. 0.0 .OR. X .GT. 1.0) GO TO 320
      IF (Y .LT. 0.0 .OR. Y .GT. 1.0) GO TO 330
      Z = DBLE(X) + DBLE(Y) - 1.D0
      IF (ABS(Z) .GT. EPS) GO TO 340
C
      IERR = 0
      IF (X .EQ. 0.0) GO TO 200
      IF (Y .EQ. 0.0) GO TO 210
      IF (A .EQ. 0.0) GO TO 211
      IF (B .EQ. 0.0) GO TO 201
C
      IND = 0
      A0 = A
```

```
      BO = B
      XO = X
      YO = Y
      EPS = AMAX1(EPS, 1.E-15)
      IF (AMIN1(AO, BO) .GT. 1.0) GO TO 30
C
C              PROCEDURE FOR AO .LE. 1 OR BO .LE. 1
C
      IF (X .LE. 0.5) GO TO 10
      IND = 1
      AO = B
      BO = A
      XO = Y
      YO = X
C
   10 IF (AMAX1(AO, BO) .GT. 1.0) GO TO 20
      IF (AO .GE. AMIN1(0.2, BO)) GO TO 100
      IF (XO**AO .LE. 0.9) GO TO 100
      IF (XO .GE. 0.3) GO TO 110
      N = 20
      GO TO 130
C
   20 IF (BO .LE. 1.0) GO TO 100
      IF (XO .GE. 0.3) GO TO 110
      IF (XO .GE. 0.1) GO TO 21
      IF ((XO*BO)**AO .LE. 0.7) GO TO 100
   21 IF (BO .GT. 15.0) GO TO 131
      N = 20
      GO TO 130
C
C              PROCEDURE FOR AO .GT. 1 AND BO .GT. 1
C
   30 IF (A .GT. B) GO TO 31
         LAMBDA = A - (A + B)*X
         GO TO 32
   31 LAMBDA = (A + B)*Y - B
   32 IF (LAMBDA .GE. 0.0) GO TO 40
      IND = 1
      AO = B
      BO = A
      XO = Y
      YO = X
      LAMBDA = ABS(LAMBDA)
C
   40 IF (BO .LT. 40.0 .AND. BO*XO .LE. 0.7) GO TO 100
      IF (BO .LT. 40.0) GO TO 140
      IF (AO .GT. BO) GO TO 50
         IF (AO .LE. 100.0) GO TO 120
         IF (LAMBDA .GT. 0.03*AO) GO TO 120
         GO TO 180
   50 IF (BO .LE. 100.0) GO TO 120
      IF (LAMBDA .GT. 0.03*BO) GO TO 120
      GO TO 180
C
C             EVALUATION OF THE APPROPRIATE ALGORITHM
C
  100 W = BPSER(AO, BO, XO, EPS)
```

```
      W1 = 0.5 + (0.5 - W)
      GO TO 220
C
  110 W1 = BPSER(BO, AO, YO, EPS)
      W = 0.5 + (0.5 - W1)
      GO TO 220
C
  120 W = BFRAC(AO, BO, XO, YO, LAMBDA, 15.0*EPS)
      W1 = 0.5 + (0.5 - W)
      GO TO 220
C
  130 W1 = BUP(BO, AO, YO, XO, N, EPS)
      BO = BO + N
  131 CALL BGRAT(BO, AO, YO, XO, W1, 15.0*EPS, IERR1)
      W = 0.5 + (0.5 - W1)
      GO TO 220
C
  140 N = BO
      BO = BO - N
      IF (BO .NE. 0.0) GO TO 141
         N = N - 1
         BO = 1.0
  141 W = BUP(BO, AO, YO, XO, N, EPS)
      IF (XO .GT. 0.7) GO TO 150
      W = W + BPSER(AO, BO, XO, EPS)
      W1 = 0.5 + (0.5 - W)
      GO TO 220
C
  150 IF (AO .GT. 15.0) GO TO 151
         N = 20
         W = W + BUP(AO, BO, XO, YO, N, EPS)
         AO = AO + N
  151 CALL BGRAT(AO, BO, XO, YO, W, 15.0*EPS, IERR1)
      W1 = 0.5 + (0.5 - W)
      GO TO 220
C
  180 W = BASYM(AO, BO, XO, YO, LAMBDA, 100.0*EPS)
      W1 = 0.5 + (0.5 - W)
      GO TO 220
C
C                TERMINATION OF THE PROCEDURE
C
  200 IF (A .EQ. 0.0) GO TO 350
  201 W = 0.0
      W1 = 1.0
      RETURN
C
  210 IF (B .EQ. 0.0) GO TO 360
  211 W = 1.0
      W1 = 0.0
      RETURN
C
  220 IF (IND .EQ. 0) RETURN
      T = W
      W = W1
      W1 = T
      RETURN
```

```
C
C                         ERROR RETURN
C
  300 IERR = 1
      RETURN
  310 IERR = 2
      RETURN
  320 IERR = 3
      RETURN
  330 IERR = 4
      RETURN
  340 IERR = 5
      RETURN
  350 IERR = 6
      RETURN
  360 IERR = 7
      RETURN
      END
```

```
      REAL FUNCTION BPSER(A, B, X, EPS)
C-----------------------------------------------------------------------
C     POWER SERIES EXPANSION FOR EVALUATING IX(A,B) WHEN B .LE. 1
C     OR B*X .LE. 0.7.  EPS IS THE TOLERANCE USED.
C-----------------------------------------------------------------------
      REAL N
C
      BPSER = 0.0
      IF (X .EQ. 0.0) RETURN
C-----------------------------------------------------------------------
C              COMPUTE THE FACTOR X**A/(A*BETA(A,B))
C-----------------------------------------------------------------------
      A0 = AMIN1(A,B)
      IF (A0 .LT. 1.0) GO TO 10
         Z = A*ALOG(X) - BETALN(A,B)
         BPSER = EXP(Z)/A
         GO TO 70
   10 B0 = AMAX1(A,B)
      IF (B0 .GE. 8.0) GO TO 60
      IF (B0 .GT. 1.0) GO TO 40
C
C              PROCEDURE FOR A0 .LT. 1 AND B0 .LE. 1
C
      BPSER = X**A
      IF (BPSER .EQ. 0.0) RETURN
C
      APB = A + B
      IF (APB .GT. 1.0) GO TO 20
         Z = 1.0 + GAM1(APB)
         GO TO 30
   20 U = DBLE(A) + DBLE(B) - 1.D0
      Z = (1.0 + GAM1(U))/APB
C
   30 C = (1.0 + GAM1(A))*(1.0 + GAM1(B))/Z
      BPSER = BPSER*C*(B/APB)
      GO TO 70
C
C           PROCEDURE FOR A0 .LT. 1 AND 1 .LT. B0 .LT. 8
C
   40 U = GAMLN1(A0)
      M = B0 - 1.0
      IF (M .LT. 1) GO TO 50
      C = 1.0
      DO 41 I = 1,M
         B0 = B0 - 1.0
   41    C = C*(B0/(A0 + B0))
      U = ALOG(C) + U
C
   50 Z = A*ALOG(X) - U
      B0 = B0 - 1.0
      APB = A0 + B0
      IF (APB .GT. 1.0) GO TO 51
         T = 1.0 + GAM1(APB)
         GO TO 52
   51 U = DBLE(A0) + DBLE(B0) - 1.D0
      T = (1.0 + GAM1(U))/APB
   52 BPSER = EXP(Z)*(A0/A)*(1.0 + GAM1(B0))/T
```

```
      GO TO 70
C
C              PROCEDURE FOR AO .LT. 1 AND BO .GE. 8
C
   60 U = GAMLN1(AO) + ALGDIV(AO,BO)
      Z = A*ALOG(X) - U
      BPSER = (AO/A)*EXP(Z)
   70 IF (BPSER .EQ. 0.0 .OR. A .LE. 0.1*EPS) RETURN
C-----------------------------------------------------------------------
C                         COMPUTE THE SERIES
C-----------------------------------------------------------------------
      SUM = 0.0
      N = 0.0
      C = 1.0
      TOL = EPS/A
  100    N = N + 1.0
         C = C*(0.5 + (0.5 - B/N))*X
         W = C/(A + N)
         SUM = SUM + W
         IF (ABS(W) .GT. TOL) GO TO 100
      BPSER = BPSER*(1.0 + A*SUM)
      RETURN
      END
```

```
      REAL FUNCTION BUP(A, B, X, Y, N, EPS)
C-----------------------------------------------------------------------
C     EVALUATION OF IX(A,B) - IX(A+N,B) WHERE N IS A POSITIVE INTEGER.
C     EPS IS THE TOLERANCE USED.
C-----------------------------------------------------------------------
      REAL L
C
      BUP = BRCOMP(A,B,X,Y)/A
      IF (N .EQ. 1 .OR. BUP .EQ. 0.0) RETURN
      NM1 = N - 1
      APB = A + B
      AP1 = A + 1.0
      SUM = 1.0
      D = 1.0
C
C          LET K BE THE INDEX OF THE MAXIMUM TERM
C
      K = 0
      IF (B .LE. 1.0) GO TO 30
      IF (Y .GT. 1.E-4) GO TO 10
         K = NM1
         GO TO 20
   10 R = (B - 1.0)*X/Y - A
      IF (R .LT. 1.0) GO TO 30
      K = NM1
      IF (R .LT. FLOAT(NM1)) K = R
C
C          ADD THE INCREASING TERMS OF THE SERIES
C
   20 DO 21 I = 1,K
      L = I - 1
      D = ((APB + L)/(AP1 + L))*X*D
   21 SUM = SUM + D
      IF (K .EQ. NM1) GO TO 40
C
C          ADD THE REMAINING TERMS OF THE SERIES
C
   30 KP1 = K + 1
      DO 31 I = KP1,NM1
      L = I - 1
      D = ((APB + L)/(AP1 + L))*X*D
      SUM = SUM + D
      IF (D .LE. EPS*SUM) GO TO 40
   31 CONTINUE
C
C                 TERMINATE THE PROCEDURE
C
   40 BUP = BUP*SUM
      RETURN
      END
```

```
      REAL FUNCTION BFRAC(A, B, X, Y, LAMBDA, EPS)
C-----------------------------------------------------------------------
C     CONTINUED FRACTION EXPANSION FOR IX(A,B) WHEN A,B .GT. 1.
C     IT IS ASSUMED THAT  LAMBDA = (A + B)*Y - B.
C-----------------------------------------------------------------------
      REAL LAMBDA, N
C------------------
      BFRAC = BRCOMP(A,B,X,Y)
      IF (BFRAC .EQ. 0.0) RETURN
C
      C = 1.0 + LAMBDA
      CO = B/A
      C1 = 1.0 + 1.0/A
      YP1 = Y + 1.0
C
      N = 0.0
      P = 1.0
      S = A + 1.0
      AN = 0.0
      BN = 1.0
      ANP1 = 1.0
      BNP1 = C/C1
      R = C1/C
C
C        CONTINUED FRACTION CALCULATION
C
   10    N = N + 1.0
         T = N + N
         W = N*(B - N)*X
         E = A/S
         ALPHA = (P*(P + CO)*E*E)*(W*X)
         E = (1.0 + T)/(C1 + T + T)
         BETA = N + W/S + E*(C + N*YP1)
         P = 1.0 + T
         S = S + 2.0
C
C        UPDATE AN, BN, ANP1, AND BNP1
C
         T = ALPHA*AN + BETA*ANP1
         AN = ANP1
         ANP1 = T
         T = ALPHA*BN + BETA*BNP1
         BN = BNP1
         BNP1 = T
C
         RO = R
         R = ANP1/BNP1
         IF (ABS(R - RO) .LE. EPS*R) GO TO 20
C
C        RESCALE AN, BN, ANP1, AND BNP1
C
         AN = AN/BNP1
         BN = BN/BNP1
         ANP1 = R
         BNP1 = 1.0
         GO TO 10
C
```

```
C                    TERMINATION
C
   20 BFRAC = BFRAC*R
      RETURN
      END
```

```
      REAL FUNCTION BRCOMP(A, B, X, Y)
C-----------------------------------------------------------------------
C     EVALUATION OF X**A*Y**B/BETA(A,B)
C-----------------------------------------------------------------------
      REAL LAMBDA, LNX, LNY
C----------------
C     CONST = 1/SQRT(2*PI)
C----------------
      DATA CONST/.398942280401433/
C
      AO = AMIN1(A,B)
      IF (AO .GE. 8.0) GO TO 100
C
      IF (X .GT. 0.375) GO TO 10
         LNX = ALOG(X)
         LNY = ALNREL(-X)
         GO TO 20
   10 IF (Y .GT. 0.375) GO TO 11
         LNX = ALNREL(-Y)
         LNY = ALOG(Y)
         GO TO 20
   11 LNX = ALOG(X)
      LNY = ALOG(Y)
C
   20 IF (AO .LT. 1.0) GO TO 30
      Z = (A*LNX + B*LNY) - BETALN(A,B)
      BRCOMP = EXP(Z)
      RETURN
C-----------------------------------------------------------------------
C               PROCEDURE FOR A .LT. 1 OR B .LT. 1
C-----------------------------------------------------------------------
   30 BO = AMAX1(A,B)
      IF (BO .GE. 8.0) GO TO 80
      IF (BO .GT. 1.0) GO TO 60
C
C                    ALGORITHM FOR BO .LE. 1
C
      BRCOMP = EXP(A*LNX + B*LNY)
      IF (BRCOMP .EQ. 0.0) RETURN
C
      APB = A + B
      IF (APB .GT. 1.0) GO TO 40
         Z = 1.0 + GAM1(APB)
         GO TO 50
   40 U = DBLE(A) + DBLE(B) - 1.DO
      Z = (1.0 + GAM1(U))/APB
C
   50 C = (1.0 + GAM1(A))*(1.0 + GAM1(B))/Z
      BRCOMP = BRCOMP*(AO*C)/(1.0 + AO/BO)
      RETURN
C
C                ALGORITHM FOR 1 .LT. BO .LT. 8
C
   60 U = GAMLN1(AO)
      N = BO - 1.0
      IF (N .LT. 1) GO TO 70
      C = 1.0
```

```
      DO 61 I = 1,N
         BO = BO - 1.0
   61    C = C*(BO/(AO + BO))
      U = ALOG(C) + U
C
   70 Z = (A*LNX + B*LNY) - U
      BO = BO - 1.0
      APB = AO + BO
      IF (APB .GT. 1.0) GO TO 71
         T = 1.0 + GAM1(APB)
         GO TO 72
   71 U = DBLE(AO) + DBLE(BO) - 1.DO
      T = (1.0 + GAM1(U))/APB
   72 BRCOMP = AO*EXP(Z)*(1.0 + GAM1(BO))/T
      RETURN
C
C                      ALGORITHM FOR BO .GE. 8
C
   80 U = GAMLN1(AO) + ALGDIV(AO,BO)
      Z = (A*LNX + B*LNY) - U
      BRCOMP = AO*EXP(Z)
      RETURN
C-----------------------------------------------------------------------
C                PROCEDURE FOR A .GE. 8 AND B .GE. 8
C-----------------------------------------------------------------------
  100 IF (A .GT. B) GO TO 101
         H = A/B
         XO = H/(1.0 + H)
         YO = 1.0/(1.0 + H)
         LAMBDA = A - (A + B)*X
         GO TO 110
  101 H = B/A
      XO = 1.0/(1.0 + H)
      YO = H/(1.0 + H)
      LAMBDA = (A + B)*Y - B
C
  110 E = -LAMBDA/A
      IF (ABS(E) .GT. 0.6) GO TO 111
         U = RLOG1(E)
         GO TO 120
  111 U = E - ALOG(X/XO)
C
  120 E = LAMBDA/B
      IF (ABS(E) .GT. 0.6) GO TO 121
         V = RLOG1(E)
         GO TO 130
  121 V = E - ALOG(Y/YO)
C
  130 Z = EXP(-(A*U + B*V))
      BRCOMP = CONST*SQRT(B*XO)*Z*EXP(-BCORR(A,B))
      RETURN
      END
```

```
      SUBROUTINE BGRAT(A, B, X, Y, W, EPS, IERR)
C-----------------------------------------------------------------------
C     ASYMPTOTIC EXPANSION FOR IX(A,B) WHEN A IS LARGER THAN B.
C     THE RESULT OF THE EXPANSION IS ADDED TO W. IT IS ASSUMED
C     THAT A .GE. 15 AND B .LE. 1.  EPS IS THE TOLERANCE USED.
C     IERR IS A VARIABLE THAT REPORTS THE STATUS OF THE RESULTS.
C-----------------------------------------------------------------------
      REAL J, L, LNX, NU, N2
      REAL C(30), D(30)
C
      BM1 = (B - 0.5) - 0.5
      NU = A + 0.5*BM1
      IF (Y .GT. 0.375) GO TO 10
         LNX = ALNREL(-Y)
         GO TO 11
   10 LNX = ALOG(X)
   11 Z = -NU*LNX
      IF (B*Z .EQ. 0.0) GO TO 100
C
C                 COMPUTATION OF THE EXPANSION
C                 SET R = EXP(-Z)*Z**B/GAMMA(B)
C
      R = B*(1.0 + GAM1(B))*EXP(B*ALOG(Z))
      R = R*EXP(A*LNX)*EXP(0.5*BM1*LNX)
      U = ALGDIV(B,A) + B*ALOG(NU)
      U = R*EXP(-U)
      IF (U .EQ. 0.0) GO TO 100
      CALL GRAT1(B,Z,R,P,Q,EPS)
C
      V = 0.25*(1.0/NU)**2
      T2 = 0.25*LNX*LNX
      L = W/U
      J = Q/R
      SUM = J
      T = 1.0
      CN = 1.0
      N2 = 0.0
      DO 22 N = 1,30
         BP2N = B + N2
         J = (BP2N*(BP2N + 1.0)*J + (Z + BP2N + 1.0)*T)*V
         N2 = N2 + 2.0
         T = T*T2
         CN = CN/(N2*(N2 + 1.0))
         C(N) = CN
         S = 0.0
         IF (N .EQ. 1) GO TO 21
            NM1 = N - 1
            COEF = B - N
            DO 20 I = 1,NM1
               S = S + COEF*C(I)*D(N-I)
   20          COEF = COEF + B
   21    D(N) = BM1*CN + S/N
         DJ = D(N)*J
         SUM = SUM + DJ
         IF (SUM .LE. 0.0) GO TO 100
         IF (ABS(DJ) .LE. EPS*(SUM + L)) GO TO 30
   22 CONTINUE
```

```
C
C                          ADD THE RESULTS TO W
C
   30 IERR = 0
      W = W + U*SUM
      RETURN
C
C                    THE EXPANSION CANNOT BE COMPUTED
C
  100 IERR = 1
      RETURN
      END
```

```
      SUBROUTINE GRAT1 (A,X,R,P,Q,EPS)
      REAL J, L
C--------------------------------------------------------------------
C         EVALUATION OF THE INCOMPLETE GAMMA RATIO FUNCTIONS
C                         P(A,X) AND Q(A,X)
C
C     IT IS ASSUMED THAT A .LE. 1.  EPS IS THE TOLERANCE TO BE USED.
C     THE INPUT ARGUMENT R HAS THE VALUE E**(-X)*X**A/GAMMA(A).
C--------------------------------------------------------------------
      IF (A*X .EQ. 0.0) GO TO 130
      IF (A .EQ. 0.5) GO TO 120
      IF (X .LT. 1.1) GO TO 10
      GO TO 50
C
C              TAYLOR SERIES FOR P(A,X)/X**A
C
   10 AN = 3.0
      C = X
      SUM = X/(A + 3.0)
      TOL = 0.1*EPS/(A + 1.0)
   11    AN = AN + 1.0
         C = -C*(X/AN)
         T = C/(A + AN)
         SUM = SUM + T
         IF (ABS(T) .GT. TOL) GO TO 11
      J = A*X*((SUM/6.0 - 0.5/(A + 2.0))*X + 1.0/(A + 1.0))
C
      Z = A*ALOG(X)
      H = GAM1(A)
      G = 1.0 + H
      IF (X .LT. 0.25) GO TO 20
         IF (A .LT. X/2.59) GO TO 40
         GO TO 30
   20 IF (Z .GT. -.13394) GO TO 40
C
   30 W = EXP(Z)
      P = W*G*(0.5 + (0.5 - J))
      Q = 0.5 + (0.5 - P)
      RETURN
C
   40 L = REXP(Z)
      W = 0.5 + (0.5 + L)
      Q = (W*J - L)*G - H
      IF (Q .LT. 0.0) GO TO 110
      P = 0.5 + (0.5 - Q)
      RETURN
C
C              CONTINUED FRACTION EXPANSION
C
   50 A2NM1 = 1.0
      A2N = 1.0
      B2NM1 = X
      B2N = X + (1.0 - A)
      C = 1.0
   51    A2NM1 = X*A2N + C*A2NM1
         B2NM1 = X*B2N + C*B2NM1
         AMO = A2NM1/B2NM1
```

```
      C = C + 1.0
      CMA = C - A
      A2N = A2NM1 + CMA*A2N
      B2N = B2NM1 + CMA*B2N
      ANO = A2N/B2N
      IF (ABS(ANO - AMO) .GE. EPS*ANO) GO TO 51
   Q = R*ANO
   P = 0.5 + (0.5 - Q)
   RETURN
C
C              SPECIAL CASES
C
100 P = 0.0
    Q = 1.0
    RETURN
C
110 P = 1.0
    Q = 0.0
    RETURN
C
120 IF (X .GE. 0.25) GO TO 121
    P = ERF(SQRT(X))
    Q = 0.5 + (0.5 - P)
    RETURN
121 Q = ERFC1(0,SQRT(X))
    P = 0.5 + (0.5 - Q)
    RETURN
C
130 IF (X .LE. A) GO TO 100
    GO TO 110
    END
```

```
      REAL FUNCTION BASYM(A, B, X, Y, LAMBDA, EPS)
C-----------------------------------------------------------------------
C     ASYMPTOTIC EXPANSION FOR IX(A,B) FOR LARGE A AND B.
C     LAMBDA = (A + B)*Y - B  AND EPS IS THE TOLERANCE USED.
C     IT IS ASSUMED THAT LAMBDA IS NONNEGATIVE AND THAT
C     A AND B ARE GREATER THAN OR EQUAL TO 15.
C-----------------------------------------------------------------------
      REAL JO, J1, LAMBDA
      REAL AO(21), BO(21), C(21), D(21)
C------------------------
C     ****** NUM IS THE MAXIMUM VALUE THAT N CAN TAKE IN THE DO LOOP
C            ENDING AT STATEMENT 50. IT IS REQUIRED THAT NUM BE EVEN.
C            THE ARRAYS AO, BO, C, D HAVE DIMENSION NUM + 1.
C
                       DATA NUM/20/
C------------------------
C     EO = 2/SQRT(PI)
C     E1 = 2**(-3/2)
C------------------------
      DATA EO/1.12837916709551/, E1/.353553390593274/
C------------------------
      BASYM = 0.0
      IF (A .GE. B) GO TO 10
         H = A/B
         RO = 1.0/(1.0 + H)
         R1 = (B - A)/B
         WO = 1.0/SQRT(A*(1.0 + H))
         GO TO 20
   10 H = B/A
      RO = 1.0/(1.0 + H)
      R1 = (B - A)/A
      WO = 1.0/SQRT(B*(1.0 + H))
C
   20 F = A*RLOG1(-LAMBDA/A) + B*RLOG1(LAMBDA/B)
      T = EXP(-F)
      IF (T .EQ. 0.0) RETURN
      ZO = SQRT(F)
      Z = 0.5*(ZO/E1)
      Z2 = F + F
C
      AO(1) = (2.0/3.0)*R1
      C(1) = - 0.5*AO(1)
      D(1) = - C(1)
      JO = (0.5/EO)*ERFC1(1,ZO)
      J1 = E1
      SUM = JO + D(1)*WO*J1
C
      S = 1.0
      H2 = H*H
      HN = 1.0
      W = WO
      ZNM1 = Z
      ZN = Z2
      DO 50 N = 2, NUM, 2
         HN = H2*HN
         AO(N) = 2.0*RO*(1.0 + H*HN)/(N + 2.0)
         NP1 = N + 1
```

```
         S = S + HN
         AO(NP1) = 2.0*R1*S/(N + 3.0)
C
         DO 41 I = N, NP1
         R = -0.5*(I + 1.0)
         BO(1) = R*AO(1)
         DO 31 M = 2, I
            BSUM = 0.0
            MM1 = M - 1
            DO 30 J = 1, MM1
               MMJ = M - J
   30          BSUM = BSUM + (J*R - MMJ)*AO(J)*BO(MMJ)
   31       BO(M) = R*AO(M) + BSUM/M
         C(I) = BO(I)/(I + 1.0)
C
         DSUM = 0.0
         IM1 = I - 1
         DO 40 J = 1, IM1
            IMJ = I - J
   40       DSUM = DSUM + D(IMJ)*C(J)
   41    D(I) = -(DSUM + C(I))
C
         JO = E1*ZNM1 + (N - 1.0)*JO
         J1 = E1*ZN + N*J1
         ZNM1 = Z2*ZNM1
         ZN = Z2*ZN
         W = WO*W
         TO = D(N)*W*JO
         W = WO*W
         T1 = D(NP1)*W*J1
         SUM = SUM + (TO + T1)
         IF ((ABS(TO) + ABS(T1)) .LE. EPS*SUM) GO TO 60
   50    CONTINUE
C
   60 U = EXP(-BCORR(A,B))
      BASYM = EO*T*U*SUM
      RETURN
      END
```

```
      REAL FUNCTION REXP(X)
C     ------------------------------------------------------------------
C     COMPUTATION OF EXP(X) - 1
C     ------------------------------------------------------------------
      DATA P1/ .914041914819518E-09/, P2/ .238082361044469E-01/,
     *     Q1/-.499999999085958E+00/, Q2/ .107141568980644E+00/,
     *     Q3/-.119041179760821E-01/, Q4/ .595130811860248E-03/
C     ------------------
      IF (ABS(X) .GT. 0.15) GO TO 10
      REXP = X*(((P2*X + P1)*X + 1.0)/((((Q4*X + Q3)*X + Q2)*X
     *                 + Q1)*X + 1.0))
      RETURN
C
   10 W = EXP(X)
      IF (X .GT. 0.0) GO TO 20
         REXP = (W - 0.5) - 0.5
         RETURN
   20 REXP = W*(0.5 + (0.5 - 1.0/W))
      RETURN
      END
```

```
      REAL FUNCTION ALNREL(A)
C-----------------------------------------------------------------------
C            EVALUATION OF THE FUNCTION LN(1 + A)
C-----------------------------------------------------------------------
      DATA P1/-.129418923021993E+01/, P2/.405303492862024E+00/,
     *     P3/-.178874546012214E-01/
      DATA Q1/-.162752256355323E+01/, Q2/.747811014037616E+00/,
     *     Q3/-.845104217945565E-01/
C--------------------------
      IF (ABS(A) .GT. 0.375) GO TO 10
      T = A/(A + 2.0)
      T2 = T*T
      W = (((P3*T2 + P2)*T2 + P1)*T2 + 1.0)/
     *    (((Q3*T2 + Q2)*T2 + Q1)*T2 + 1.0)
      ALNREL = 2.0*T*W
      RETURN
C
   10 X = 1.D0 + DBLE(A)
      ALNREL = ALOG(X)
      RETURN
      END
```

```
      REAL FUNCTION RLOG1(X)
C     ------------------
C     COMPUTATION OF  X - LN(1 + X)
C     ------------------
      DATA A/.566749439387324E-01/
      DATA B/.456512608815524E-01/
C     ------------------
      DATA PO/ .333333333333333E+00/, P1/-.224696413112536E+00/,
     *     P2/ .620886815375787E-02/
      DATA Q1/-.127408923933623E+01/, Q2/ .354508718369557E+00/
C     ------------------
      IF (X .LT. -0.39 .OR. X .GT. 0.57) GO TO 100
      IF (X .LT. -0.18) GO TO 10
      IF (X .GT.  0.18) GO TO 20
C
C              ARGUMENT REDUCTION
C
      H = X
      W1 = 0.0
      GO TO 30
C
   10 H = DBLE(X) + 0.3D0
      H = H/0.7
      W1 = A - H*0.3
      GO TO 30
C
   20 H = 0.75D0*DBLE(X) - 0.25D0
      W1 = B + H/3.0
C
C               SERIES EXPANSION
C
   30 R = H/(H + 2.0)
      T = R*R
      W = ((P2*T + P1)*T + PO)/((Q2*T + Q1)*T + 1.0)
      RLOG1 = 2.0*T*(1.0/(1.0 - R) - R*W) + W1
      RETURN
C
C
  100 W = (X + 0.5) + 0.5
      RLOG1 = X - ALOG(W)
      RETURN
      END
```

```
      REAL FUNCTION ERF (X)
C-----------------------------------------------------------------------
C              EVALUATION OF THE REAL ERROR FUNCTION
C-----------------------------------------------------------------------
      REAL A(4),B(4),P(8),Q(8),R(5),S(5)
C------------------------
      DATA C/.564189583547756/
C------------------------
      DATA A(1)/-1.65581836870402E-4/, A(2)/3.25324098357738E-2/,
     *     A(3)/1.02201136918406E-1/,  A(4)/1.12837916709552E00/
      DATA B(1)/4.64988945913179E-3/,  B(2)/7.01333417158511E-2/,
     *     B(3)/4.23906732683201E-1/,  B(4)/1.00000000000000E00/
      DATA P(1)/-1.36864857382717E-7/, P(2)/5.64195517478974E-1/,
     *     P(3)/7.21175825088309E00/,  P(4)/4.31622272220567E01/,
     *     P(5)/1.52989285046940E02/,  P(6)/3.39320816734344E02/,
     *     P(7)/4.51918953711873E02/,  P(8)/3.00459261020162E02/
      DATA Q(1)/1.00000000000000E00/,  Q(2)/1.27827273196294E01/,
     *     Q(3)/7.70001529352295E01/,  Q(4)/2.77585444743988E02/,
     *     Q(5)/6.38980264465631E02/,  Q(6)/9.31354094850610E02/,
     *     Q(7)/7.90950925327898E02/,  Q(8)/3.00459260956983E02/
      DATA R(1)/2.10144126479064E00/,  R(2)/2.62370141675169E01/,
     *     R(3)/2.13688200555087E01/,  R(4)/4.65807828718470E00/,
     *     R(5)/2.82094791773523E-1/
      DATA S(1)/9.41537750555460E01/,  S(2)/1.87114811799590E02/,
     *     S(3)/9.90191814623914E01/,  S(4)/1.80124575948747E01/,
     *     S(5)/1.00000000000000E00/
C------------------------
      AX = ABS(X)
      IF (AX .GE. 0.5) GO TO 10
      T = X*X
      TOP = ((A(1)*T + A(2))*T + A(3))*T + A(4)
      BOT = ((B(1)*T + B(2))*T + B(3))*T + B(4)
      ERF = X*TOP/BOT
      RETURN
C
   10 IF (AX .GT. 4.0) GO TO 20
      TOP = ((((((P(1)*AX + P(2))*AX + P(3))*AX + P(4))*AX + P(5))*AX
     *                    + P(6))*AX + P(7))*AX + P(8)
      BOT = ((((((Q(1)*AX + Q(2))*AX + Q(3))*AX + Q(4))*AX + Q(5))*AX
     *                    + Q(6))*AX + Q(7))*AX + Q(8)
      ERF = 0.5 + (0.5 - EXP(-X*X)*TOP/BOT)
      IF (X .LT. 0.0) ERF = -ERF
      RETURN
C
   20 ERF = 1.0
      IF (AX .GE. 5.6) GO TO 21
      X2 = X*X
      T = 1.0/X2
      TOP = (((R(1)*T + R(2))*T + R(3))*T + R(4))*T + R(5)
      BOT = (((S(1)*T + S(2))*T + S(3))*T + S(4))*T + S(5)
      ERF = (C - TOP/(X2*BOT)) / AX
      ERF = 0.5 + (0.5 - EXP(-X2)*ERF)
   21 IF (X .LT. 0.0) ERF = -ERF
      RETURN
      END
```

```
      REAL FUNCTION ERFC1 (IND, X)
C-----------------------------------------------------------------------
C         EVALUATION OF THE COMPLEMENTARY ERROR FUNCTION
C
C          ERFC1(IND,X) = ERFC(X)            IF IND = 0
C          ERFC1(IND,X) = EXP(X*X)*ERFC(X)   OTHERWISE
C-----------------------------------------------------------------------
      REAL A(4),B(4),P(8),Q(8),R(5),S(5)
      DOUBLE PRECISION W
C------------------------
      DATA C/.564189583547756/
C------------------------
      DATA A(1)/-1.65581836870402E-4/, A(2)/3.25324098357738E-2/,
     *     A(3)/1.02201136918406E-1/,  A(4)/1.12837916709552E00/
      DATA B(1)/4.64988945913179E-3/,  B(2)/7.01333417158511E-2/,
     *     B(3)/4.23906732683201E-1/,  B(4)/1.00000000000000E00/
      DATA P(1)/-1.36864857382717E-7/, P(2)/5.64195517478974E-1/,
     *     P(3)/7.21175825088309E00/,  P(4)/4.31622272220567E01/,
     *     P(5)/1.52989285046940E02/,  P(6)/3.39320816734344E02/,
     *     P(7)/4.51918953711873E02/,  P(8)/3.00459261020162E02/
      DATA Q(1)/1.00000000000000E00/,  Q(2)/1.27827273196294E01/,
     *     Q(3)/7.70001529352295E01/,  Q(4)/2.77585444743988E02/,
     *     Q(5)/6.38980264465631E02/,  Q(6)/9.31354094850610E02/,
     *     Q(7)/7.90950925327898E02/,  Q(8)/3.00459260956983E02/
      DATA R(1)/2.10144126479064E00/,  R(2)/2.62370141675169E01/,
     *     R(3)/2.13688200555087E01/,  R(4)/4.65807828718470E00/,
     *     R(5)/2.82094791773523E-1/
      DATA S(1)/9.41537750555460E01/,  S(2)/1.87114811799590E02/,
     *     S(3)/9.90191814623914E01/,  S(4)/1.80124575948747E01/,
     *     S(5)/1.00000000000000E00/
C------------------------
C
C                     ABS(X) .LT. 0.47
C
      AX = ABS(X)
      IF (AX .GE. 0.47) GO TO 10
      T = X*X
      TOP = ((A(1)*T + A(2))*T + A(3))*T + A(4)
      BOT = ((B(1)*T + B(2))*T + B(3))*T + B(4)
      ERFC1 = 0.5 + (0.5 - X*TOP/BOT)
      IF (IND .NE. 0) ERFC1 = EXP(T) * ERFC1
      RETURN
C
C                  0.47 .LE. ABS(X) .LE. 4
C
   10 IF (AX .GT. 4.0) GO TO 30
      TOP = ((((((P(1)*AX + P(2))*AX + P(3))*AX + P(4))*AX + P(5))*AX
     *                    + P(6))*AX + P(7))*AX + P(8)
      BOT = ((((((Q(1)*AX + Q(2))*AX + Q(3))*AX + Q(4))*AX + Q(5))*AX
     *                    + Q(6))*AX + Q(7))*AX + Q(8)
      ERFC1 = TOP/BOT
C
   20 IF (IND .EQ. 0.0) GO TO 21
         IF (X .LT. 0.0) ERFC1 = 2.0*EXP(X*X) - ERFC1
         RETURN
   21 W = DBLE(X)*DBLE(X)
      T = W
```

```
      E = W - DBLE(T)
      ERFC1 = ((0.5 + (0.5 - E)) * EXP(-T)) * ERFC1
      IF (X .LT. 0.0) ERFC1 = 2.0 - ERFC1
      RETURN
C
C                        ABS(X) .GT. 4
C
   30 IF (X .LE. -5.5) GO TO 40
      IF (IND .EQ. 0.0 .AND. X .GT. 50.0) GO TO 50
      T = (1.0/X)**2
      TOP = (((R(1)*T + R(2))*T + R(3))*T + R(4))*T + R(5)
      BOT = (((S(1)*T + S(2))*T + S(3))*T + S(4))*T + S(5)
      ERFC1 = (C - T*TOP/BOT) / AX
      GO TO 20
C
C                LIMIT VALUE FOR LARGE NEGATIVE X
C
   40 ERFC1 = 2.0
      IF (IND .NE. 0) ERFC1 = 2.0*EXP(X*X)
      RETURN
C
C                LIMIT VALUE FOR LARGE POSITIVE X
C                         WHEN IND = 0
C
   50 ERFC1 = 0.0
      RETURN
      END
```

```
      REAL FUNCTION GAM1(A)
C     -----------------------------------------------------------------------
C     COMPUTATION OF 1/GAMMA(A+1) - 1  FOR -0.5 .LE. A .LE. 1.5
C     -----------------------------------------------------------------------
      REAL P(7), Q(5), R(9)
C     ------------------
      DATA P(1)/ .577215664901533E+00/, P(2)/-.409078193005776E+00/,
     *     P(3)/-.230975380857675E+00/, P(4)/ .597275330452234E-01/,
     *     P(5)/ .766968181649490E-02/, P(6)/-.514889771323592E-02/,
     *     P(7)/ .589597428611429E-03/
C     ------------------
      DATA Q(1)/ .100000000000000E+01/, Q(2)/ .427569613095214E+00/,
     *     Q(3)/ .158451672430138E+00/, Q(4)/ .261132021441447E-01/,
     *     Q(5)/ .423244297896961E-02/
C     ------------------
      DATA R(1)/-.422784335098468E+00/, R(2)/-.771330383816272E+00/,
     *     R(3)/-.244757765222226E+00/, R(4)/ .118378989872749E+00/,
     *     R(5)/ .930357293360349E-03/, R(6)/-.118290993445146E-01/,
     *     R(7)/ .223047661158249E-02/, R(8)/ .266505979058923E-03/,
     *     R(9)/-.132674909766242E-03/
C     ------------------
      DATA S1  / .273076135303957E+00/, S2  / .559398236957378E-01/
C     ------------------
      T = A
      D = A - 0.5
      IF (D .GT  0.0) T = D - 0.5
      IF (T) 30,10,20
C
   10 GAM1 = 0.0
      RETURN
C
   20 TOP = (((((P(7)*T + P(6))*T + P(5))*T + P(4))*T + P(3))*T
     *                     + P(2))*T + P(1)
      BOT = (((Q(5)*T + Q(4))*T + Q(3))*T + Q(2))*T + 1.0
      W = TOP/BOT
      IF (D .GT. 0.0) GO TO 21
         GAM1 = A*W
         RETURN
   21 GAM1 = (T/A)*((W - 0.5) - 0.5)
      RETURN
C
   30 TOP = (((((((R(9)*T + R(8))*T + R(7))*T + R(6))*T + R(5))*T
     *                       + R(4))*T + R(3))*T + R(2))*T + R(1)
      BOT = (S2*T + S1)*T + 1.0
      W = TOP/BOT
      IF (D .GT. 0.0) GO TO 31
         GAM1 = A*((W + 0.5) + 0.5)
         RETURN
   31 GAM1 = T*W/A
      RETURN
      END
```

```
      REAL FUNCTION GAMLN(A)
C     ****************************************************************
C     EVALUATION OF LN(GAMMA(A)) FOR POSITIVE A
C     ****************************************************************
C     WRITTEN BY ALFRED H. MORRIS
C          NAVAL SURFACE WEAPONS CENTER
C          DAHLGREN, VIRGINIA
C     --------------------
C     D = 0.5*(LN(2*PI) - 1)
C     --------------------
      DATA D/.418938533204673/
C     --------------------
      DATA C0/.833333333333333E-01/, C1/-.277777777770481E-02/,
     1     C2/.793650663183693E-03/, C3/-.595156336428591E-03/,
     2     C4/.820756370353826E-03/
C     ----------------------------------------------------------------
      IF (A .GT. 0.8) GO TO 10
         GAMLN = GAMLN1(A) - ALOG(A)
         RETURN
   10 IF (A .GT. 2.25) GO TO 20
         X = DBLE(A) - 1.D0
         GAMLN = GAMLN1(X)
         RETURN
C
   20 IF (A .GE. 15.0) GO TO 30
      N = A - 1.25
      X = A
      W = 1.0
      DO 21 I = 1,N
         X = X - 1.0
   21    W = X*W
      GAMLN = GAMLN1(X - 1.0) + ALOG(W)
      RETURN
C
   30 X = (1.0/A)**2
      W = ((((C4*X + C3)*X + C2)*X + C1)*X + C0)/A
      GAMLN = (D + W) + (A - 0.5)*(ALOG(A) - 1.0)
      END
```

```
      REAL FUNCTION GAMLN1(A)
C     -----------------------------------------------------------------------
C     EVALUATION OF LN(GAMMA(1 + A)) FOR -0.2 .LE. A .LE. 1.25
C     -----------------------------------------------------------------------
      DATA P0/ .577215664901533E+00/, P1/ .844203922187225E+00/,
     *     P2/-.168860593646662E+00/, P3/-.780427615533591E+00/,
     *     P4/-.402055799310489E+00/, P5/-.673562214325671E-01/,
     *     P6/-.271935708322958E-02/                              .
      DATA Q1/ .288743195473681E+01/, Q2/ .312755088914843E+01/,
     *     Q3/ .156875193295039E+01/, Q4/ .361951990101499E+00/,
     *     Q5/ .325038868253937E-01/, Q6/ .667465618796164E-03/
C     ----------------
      DATA R0/.422784335098467E+00/,  R1/.848044614534529E+00/,
     *     R2/.565221050691933E+00/,  R3/.156513060486551E+00/,
     *     R4/.170502484022650E-01/,  R5/.497958207639485E-03/
      DATA S1/.124313399877507E+01/,  S2/.548042109832463E+00/,
     *     S3/.101552187439830E+00/,  S4/.713309612391000E-02/,
     *     S5/.116165475989616E-03/
C     ----------------
      IF (A .GE. 0.6) GO TO 10
      W = ((((((P6*A + P5)*A + P4)*A + P3)*A + P2)*A + P1)*A + P0)/
     *    ((((((Q6*A + Q5)*A + Q4)*A + Q3)*A + Q2)*A + Q1)*A + 1.0)
      GAMLN1 = -A*W
      RETURN
C
   10 X = DBLE(A) - 1.D0
      W = (((((R5*X + R4)*X + R3)*X + R2)*X + R1)*X + R0)/
     *    (((((S5*X + S4)*X + S3)*X + S2)*X + S1)*X + 1.0)
      GAMLN1 = X*W
      RETURN
      END
```

```
      REAL FUNCTION BETALN(A0, B0)
C-----------------------------------------------------------------------
C     EVALUATION OF THE LOGARITHM OF THE BETA FUNCTION
C-----------------------------------------------------------------------
C     E = 0.5*LN(2*PI)
C--------------------------
      DATA E /.918938533204673/
C--------------------------
      A = AMIN1(A0,B0)
      B = AMAX1(A0,B0)
      IF (A .GE. 8.0) GO TO 60
      IF (A .GE. 1.0) GO TO 20
C-----------------------------------------------------------------------
C                        PROCEDURE WHEN A .LT. 1
C-----------------------------------------------------------------------
      IF (B .GE. 8.0) GO TO 10
         BETALN = GAMLN(A) + (GAMLN(B) - GAMLN(A + B))
         RETURN
   10 BETALN = GAMLN(A) + ALGDIV(A,B)
      RETURN
C-----------------------------------------------------------------------
C                   PROCEDURE WHEN 1 .LE. A .LT. 8
C-----------------------------------------------------------------------
   20 IF (A .GT. 2.0) GO TO 30
      IF (B .GT. 2.0) GO TO 21
         BETALN = GAMLN(A) + GAMLN(B) - GSUMLN(A,B)
         RETURN
   21 W = 0.0
      IF (B .LT. 8.0) GO TO 40
         BETALN = GAMLN(A) + ALGDIV(A,B)
         RETURN
C
C                     REDUCTION OF A WHEN A .GT. 2
C
   30 N = A - 1.0
      W = 0.0
      DO 31 I = 1,N
         A = A - 1.0
         H = A/B
         W = W + ALOG(H/(1.0 + H))
   31 CONTINUE
      IF (B .LT. 8.0) GO TO 40
      BETALN = W + GAMLN(A) + ALGDIV(A,B)
      RETURN
C
C                     REDUCTION OF B WHEN B .LT. 8
C
   40 N = B - 1.0
      Z = 1.0
      DO 41 I = 1,N
         B = B - 1.0
   41    Z = Z*(B/(A + B))
      BETALN = W + ALOG(Z) + (GAMLN(A) + (GAMLN(B) - GSUMLN(A,B)))
      RETURN
C-----------------------------------------------------------------------
C                        PROCEDURE WHEN A .GE. 8
C-----------------------------------------------------------------------
```

```
60 W = BCORR(A,B)
   H = A/B
   C = H/(1.0 + H)
   U = -(A - 0.5)*ALOG(C)
   V = B*ALNREL(H)
   IF (U .LE. V) GO TO 61
      BETALN = (((-0.5*ALOG(B) + E) + W) - V) - U
      RETURN
61 BETALN = (((-0.5*ALOG(B) + E) + W) - U) - V
   RETURN
   END
```

```
      REAL FUNCTION GSUMLN(A,B)
C     -----------------------------------------------------------------
C     EVALUATION OF THE FUNCTION LN(GAMMA(A + B))
C     FOR 1 .LE. A .LE. 2  AND  1 .LE. B .LE. 2
C     -----------------------------------------------------------------
      X = DBLE(A) + DBLE(B) - 2.D0
      IF (X .GT. 0.25) GO TO 10
         GSUMLN = GAMLN1(1.0 + X)
         RETURN
   10 IF (X .GT. 1.25) GO TO 20
         GSUMLN = GAMLN1(X) + ALNREL(X)
         RETURN
   20 GSUMLN = GAMLN1(X - 1.0) + ALOG(X*(1.0 + X))
      RETURN
      END
```

```fortran
      REAL FUNCTION BCORR(AO, BO)
C     ------------------------------------------------------------------
C     EVALUATION OF  DEL(AO) + DEL(BO) - DEL(AO + BO)  WHERE
C     LN(GAMMA(A)) = (A - 0.5)*LN(A) - A + 0.5*LN(2*PI) + DEL(A).
C     IT IS ASSUMED THAT AO .GE. 8 AND BO .GE. 8.
C     ------------------------------------------------------------------
      DATA CO/.833333333333333E-01/, C1/-.277777777760991E-02/,
     *     C2/.793650666825390E-03/, C3/-.595202931351870E-03/,
     *     C4/.837308034031215E-03/, C5/-.165322962780713E-02/
C     ------------------
      A = AMIN1(AO, BO)
      B = AMAX1(AO, BO)
C
      H = A/B
      C = H/(1.0 + H)
      X = 1.0/(1.0 + H)
      X2 = X*X
C
C                SET SN = (1 - X**N)/(1 - X)
C
      S3 = 1.0 + (X + X2)
      S5 = 1.0 + (X + X2*S3)
      S7 = 1.0 + (X + X2*S5)
      S9 = 1.0 + (X + X2*S7)
      S11 = 1.0 + (X + X2*S9)
C
C                SET W = DEL(B) - DEL(A + B)
C
      T = (1.0/B)**2
      W = ((((C5*S11*T + C4*S9)*T + C3*S7)*T + C2*S5)*T + C1*S3)*T + CO
      W = W*(C/B)
C
C                   COMPUTE  DEL(A) + W
C
      T = (1.0/A)**2
      BCORR = (((((C5*T + C4)*T + C3)*T + C2)*T + C1)*T + CO)/A + W
      RETURN
      END
```

```
      REAL FUNCTION ALGDIV(A,B)
C     -----------------------------------------------------------------
C     COMPUTATION OF LN(GAMMA(B)/GAMMA(A+B)) WHEN B .GE. 8
C
C     IN THIS CODE DEL(Z) IS THE FUNCTION WHERE
C     LN(GAMMA(Z)) = (Z - 0.5)*LN(Z) - Z + 0.5*LN(2*PI) + DEL(Z)
C     -----------------------------------------------------------------
      DATA C0/.833333333333333E-01/, C1/-.277777777760991E-02/,
     *     C2/.793650666825390E-03/, C3/-.595202931351870E-03/,
     *     C4/.837308034031215E-03/, C5/-.165322962780713E-02/
C     ------------------
      IF (A .LE. B) GO TO 10
         H = B/A
         C = 1.0/(1.0 + H)
         X = H/(1.0 + H)
         D = A + (B - 0.5)
         GO TO 20
   10 H = A/B
      C = H/(1.0 + H)
      X = 1.0/(1.0 + H)
      D = B + (A - 0.5)
C
C                 SET SN = (1 - X**N)/(1 - X)
C
   20 X2 = X*X
      S3 = 1.0 + (X + X2)
      S5 = 1.0 + (X + X2*S3)
      S7 = 1.0 + (X + X2*S5)
      S9 = 1.0 + (X + X2*S7)
      S11 = 1.0 + (X + X2*S9)
C
C                 SET W = DEL(B) - DEL(A + B)
C
      T = (1.0/B)**2
      W = ((((C5*S11*T + C4*S9)*T + C3*S7)*T + C2*S5)*T + C1*S3)*T + C0
      W = W*(C/B)
C
C                      COMBINE THE RESULTS
C
      U = D*ALNREL(A/B)
      V = A*(ALOG(B) - 1.0)
      IF (U .LE. V) GO TO 30
         ALGDIV = (W - V) - U
         RETURN
   30 ALGDIV = (W - U) - V
      RETURN
      END
```

```
      REAL FUNCTION SPMPAR(I)
      INTEGER I
C----------------------------------------------------------------------
C
C     SPMPAR PROVIDES THE SINGLE PRECISION MACHINE PARAMETERS FOR
C     THE COMPUTER BEING USED. IT IS ASSUMED THAT THE ARGUMENT
C     I IS AN INTEGER HAVING ONE OF THE VALUES 1, 2, OR 3. IF THE
C     SINGLE PRECISION ARITHMETIC BEING USED HAS T BASE B DIGITS AND
C     ITS SMALLEST AND LARGEST EXPONENTS ARE EMIN AND EMAX, THEN
C
C         SPMPAR(1) = B**(1 - T), THE MACHINE PRECISION,
C
C         SPMPAR(2) = B**(EMIN - 1), THE SMALLEST MAGNITUDE,
C
C         SPMPAR(3) = B**EMAX*(1 - B**(-T)), THE LARGEST MAGNITUDE.
C
C     TO DEFINE THIS FUNCTION FOR THE COMPUTER BEING USED, ACTIVATE
C     THE DATA STATMENTS FOR THE COMPUTER BY REMOVING THE C FROM
C     COLUMN 1. (ALL OTHER DATA STATEMENTS IN SPMPAR SHOULD HAVE C
C     IN COLUMN 1.)
C
C----------------------------------------------------------------------
C
C     SPMPAR IS AN ADAPTATION OF THE FUNCTION R1MACH, WRITTEN BY P.A.
C     FOX, A.D. HALL, AND N.L. SCHRYER (BELL LABORATORIES). SPMPAR
C     WAS DESIGNED BY B.S. GARBOW, K.E. HILLSTROM, AND J.J. MORE
C     (ARGONNE NATIONAL LABORATORY). THE MAJORITY OF PARAMETER VALUES
C     ARE FROM BELL LABORATORIES.
C
C----------------------------------------------------------------------
      INTEGER MCHEPS(2)
      INTEGER MINMAG(2)
      INTEGER MAXMAG(2)
      REAL RMACH(3)
      EQUIVALENCE (RMACH(1),MCHEPS(1))
      EQUIVALENCE (RMACH(2),MINMAG(1))
      EQUIVALENCE (RMACH(3),MAXMAG(1))
C
C     MACHINE CONSTANTS FOR THE BURROUGHS 1700 SYSTEM.
C
C     DATA RMACH(1) / Z4EA800000 /
C     DATA RMACH(2) / Z400800000 /
C     DATA RMACH(3) / Z5FFFFFFFF /
C
C     MACHINE CONSTANTS FOR THE BURROUGHS 5700/6700/7700 SYSTEMS.
C
C     DATA RMACH(1) / O1301000000000000 /
C     DATA RMACH(2) / O1771000000000000 /
C     DATA RMACH(3) / O0777777777777777 /
C
C     MACHINE CONSTANTS FOR THE CDC 6000/7000 SERIES.
C     (OCTAL FORMAT FOR FORTRAN 4 COMPILERS)
C
C     DATA RMACH(1) / 16414000000000000000B /
C     DATA RMACH(2) / 00014000000000000000B /
C     DATA RMACH(3) / 37767777777777777777B /
C
```

```
C     MACHINE CONSTANTS FOR THE CDC 6000/7000 SERIES.
C     (INTEGER FORMAT FOR FORTRAN 4 AND 5 COMPILERS)
C
      DATA MCHEPS(1) / 2616309908525547 52 /
      DATA MINMAG(1) /    422212465065984 /
      DATA MAXMAG(1) / 5761792773267 12831 /
C
C     MACHINE CONSTANTS FOR THE CRAY-1.
C
C     DATA RMACH(1) / 0377224000000000000000B /
C     DATA RMACH(2) / 0200034000000000000000B /
C     DATA RMACH(3) / 0577777777777777777776B /
C
C     MACHINE CONSTANTS FOR THE DATA GENERAL ECLIPSE S/200.
C
C     NOTE - IT MAY BE APPROPRIATE TO INCLUDE THE FOLLOWING CARD -
C     STATIC RMACH(3)
C
C     DATA MINMAG/20K,0/,MAXMAG/77777K,177777K/
C     DATA MCHEPS/36020K,0/
C
C     MACHINE CONSTANTS FOR THE HARRIS 220.
C
C     DATA MCHEPS(1) / '20000000, '00000353 /
C     DATA MINMAG(1) / '20000000, '00000201 /
C     DATA MAXMAG(1) / '37777777, '00000177 /
C
C     MACHINE CONSTANTS FOR THE HONEYWELL 600/6000 SERIES.
C
C     DATA RMACH(1) / 0716400000000 /
C     DATA RMACH(2) / 0402400000000 /
C     DATA RMACH(3) / 0376777777777 /
C
C     MACHINE CONSTANTS FOR THE HP 2100
C
C     DATA MCHEPS(1), MCHEPS(2) / 40000B,     327B /
C     DATA MINMAG(1), MINMAG(2) / 40000B,        1 /
C     DATA MAXMAG(1), MAXMAG(2) / 77777B,  177776B /
C
C     MACHINE CONSTANTS FOR THE HP 9000
C
C     DATA RMACH(1) / .1192093E-06 /
C     DATA RMACH(2) / .5877472E-38 /
C     DATA RMACH(3) / .3402823E+39 /
C
C     MACHINE CONSTANTS FOR THE IBM 360/370 SERIES,
C     THE AMDAHL 470/V6, THE ICL 2900, THE ITEL AS/6,
C     THE XEROX SIGMA 5/7/9 AND THE SEL SYSTEMS 85/86.
C
C     DATA RMACH(1) / Z3C100000 /
C     DATA RMACH(2) / Z00100000 /
C     DATA RMACH(3) / Z7FFFFFFF /
C
C     MACHINE CONSTANTS FOR THE IBM PC - MICROSOFT FORTRAN
C
C     DATA MCHEPS(1) / #34000000 /
C     DATA MINMAG(1) / #00800000 /
```

```
C     DATA MAXMAG(1) / #7F7FFFFF /
C
C     MACHINE CONSTANTS FOR THE IBM PC - PROFESSIONAL FORTRAN,
C     LAHEY FORTRAN, AND RM FORTRAN
C
C     DATA MCHEPS(1) / Z'34000000' /
C     DATA MINMAG(1) / Z'00800000' /
C     DATA MAXMAG(1) / Z'7F7FFFFF' /
C
C     MACHINE CONSTANTS FOR THE PDP-10 (KA OR KI PROCESSOR).
C
C     DATA RMACH(1) / "147400000000 /
C     DATA RMACH(2) / "000400000000 /
C     DATA RMACH(3) / "377777777777 /
C
C     MACHINE CONSTANTS FOR THE PDP-11 FORTRAN SUPPORTING
C     32-BIT INTEGERS (EXPRESSED IN INTEGER AND OCTAL).
C
C     DATA MCHEPS(1) /  889192448 /
C     DATA MINMAG(1) /    8388608 /
C     DATA MAXMAG(1) / 2147483647 /
C
C     DATA RMACH(1) / O06500000000 /
C     DATA RMACH(2) / O00040000000 /
C     DATA RMACH(3) / O17777777777 /
C
C     MACHINE CONSTANTS FOR THE PDP-11 FORTRAN SUPPORTING
C     16-BIT INTEGERS (EXPRESSED IN INTEGER AND OCTAL).
C
C     DATA MCHEPS(1),MCHEPS(2) / 13568,      0 /
C     DATA MINMAG(1),MINMAG(2) /   128,      0 /
C     DATA MAXMAG(1),MAXMAG(2) / 32767,     -1 /
C
C     DATA MCHEPS(1),MCHEPS(2) / O032400, O000000 /
C     DATA MINMAG(1),MINMAG(2) / O000200, O000000 /
C     DATA MAXMAG(1),MAXMAG(2) / O077777, O177777 /
C
C     MACHINE CONSTANTS FOR THE UNIVAC 1100 SERIES.
C
C     DATA RMACH(1) / O147400000000 /
C     DATA RMACH(2) / O000400000000 /
C     DATA RMACH(3) / O377777777777 /
C
C     MACHINE CONSTANTS FOR THE VAX 11/780
C     (EXPRESSED IN INTEGER AND HEXADECIMAL)
C
C     DATA MCHEPS(1) /      13568 /
C     DATA MINMAG(1) /        128 /
C     DATA MAXMAG(1) /     -32769 /
C
C     DATA MCHEPS(1) / Z00003500 /
C     DATA MINMAG(1) / Z00000080 /
C     DATA MAXMAG(1) / ZFFFF7FFF /
C
      SPMPAR = RMACH(I)
      RETURN
C
```

```
C      LAST CARD OF FUNCTION SPMPAR.
C
       END
```

DISTRIBUTION

	Copies
Chief of Naval Operations Attn: OP-961	1
OP-982	1
OP-982E	1
OP-983	1
OP-987 Washington D.C. 20350	1
Commander, Naval Facilities Engineering Command Department of the Navy Attn: Code 032F 200 Stovall Street Alexandria, VA 22332	1
Office of Naval Research Code 411 800 North Quincy Street Arlington VA 22217	1
Commander, Naval Sea Systems Command Attn: SEA 05R Department of the Navy Washington, DC 20362	1
Fleet Analysis Center Naval Weapons Station Seal Beach Attn: Library Corona, CA 91720	1
Commander, U.S. Naval Weapons Center Attn: Code 38 China Lake, CA 93555	1
Argonne National Laboratory Attn: Dr. A. H. Jaffey Bldg. 200 9700 South Cass Avenue Argonne, IL 60439	1
U.S. Naval Observatory 34th Street and Mass. Ave, N.W. Attn: Library Washington, DC 20390	1
The Library of Congress Attn: Exchange and Gift Division Washington D.C. 20540	1
Director Office of Naval Research Branch Office 1030 East Green Street Pasadena, Ca 91101	1
Commanding Officer Marine Aviation Detachment Pacific Missile Test Center Point Mugu, CA 93042	1
Director U.S. Army Ballistic Research Lab. Attn: DRDAR-TSB-S (STINFO) Aberdeen Proving Ground, MD 21005	1
Director National Security Agency Attn: Library Fort George G. Meade, MD 20755	1
Commanding General White Sands Missile Range Attn:Technical Library, Documents Section White Sands Missile Range, NM 88002	1
Director National Bureau of Standards Attn:Library Gaithersburg, MD 20760	1
Rand Corporation Attn: Library D 1700 Main Street Santa Monica, CA 90406	1
Columbia University Prof. H. Goldstein 520 W. 120th Street, Engter Bldg. New York, NY 10027	1

University of Chicago
Attn: Prof. W. Kruskal 1
Statistics Department
5734 Univ. Ave.
Chicago, IL 60637

Joint Strategic Target
Planning Staff 1
Offut Air Force Base, NB 68113

Massachustts Institute of Technology
Attn: Computation Center 1
Cambridge, MA 02139

President
Naval War College 1
Newport, RI 02840

University of New Orleans
Dr. George Ioup, Physics Dept. 1
Lake Front, New Orleans LA 70148

Willian C. Guenther
Box 3332 University Station 1
Laramie, WY 82071

Sandia National Laboratories
Dr. Donald E. Amos, 5642 1
P.O. Box 5800
Albuquerque, NM 87185

Exploratory Technology
Donald Baker Moore 1
P.O. Box KK
Fairfield, CA 94533

The Johns Hopkins University
Dr. I. Sugai, F4A 1
Applied Physics Laboratory
Johns Hopkins Road
Laurel, MD 20707

Washington State University
Prof. George Marsaglia 1
Computer Science Dept.
Pullman, WA 99164

University of Wisconsin-Madison
Prof. J. Gurland 1
Department of Statistics
1210 West Dayton Street
Madison, WI 53706

Chief of Missile and Projectile
Systems Design
John A. Simpson 1
Norden Systems Inc. P.O. Box 5300
Norwalk, CT 06856

Shering Plough Corp.
M. Miller 1
60 Orange Street
Bloomfield, NJ 07003

Department of Mathematics
Prof. Joo Koo 1
Northeastern Illinois University
Chicago, IL 60625

Trinity College
Harvey S. Picker 1
Physics Department
Hartford, CT 06106

Naval Ocean Systems Center
Attn: Library 1
San Diego, CA 92152

U.S. Naval Oceanographic Office
Navy Lib./NSTL Station 1
Bay St. Louis, MS 39522

General Electric Company
H. Saunders 1
Building 41, Room 304
One River Road
Schenectady, NY 12345

Lockheed Missile & Space Company
R. F. Hausman 1
Department 6212-- Bldg. 104
P.O. Box 504
Sunnyvale, CA 94088

Dr. Harold Crutcher 35 Westall Avenue Asheville, NC 28804	1
Deputy Director Marine Corps Operations Analysis Group Attn: Dr. B. Barfoot 2000 N. Beauregard Street Alexandria, VA 22311	1
LMSC Dr. H. Weingarten 0/8449 B 538 P.O. Box 504 Sunnyvale, CA 94088-3504	1
Harvard University D.C. Hoaglin Dept. of Statistics 1 Oxford Street Cambridge Mass 02138	1
Dr. A.V. Hershey 722 Redwood Lane Pacific Grove, CA 93950	1
Naval Research Laboratory Alvin B. Owens Code 2303 Washington, D.C. 20375	1
Office of Naval Research Attn: Math & Sciences Division Library Washington, DC 20360	1
Commander, David W. Taylor Naval Ship Research & Develop. Center	
Code 1120 (E. O'Neill)	1
Code 1129 (E. Wolfe)	1
Code 1127 (S. Baylin)	1
Code 185	1
Code 1805	1
Code 5220 (Technical Library) Bethesda, MD 20084	1
AVARDA George Cyrcos DAVAA-N-SA Fort Monmouth, NJ 07703	1
U.S. Army Tradoc Systems Analysis Activity Peter Shugart Attn: ATAA-DDD White Sands Missile Range, NM 88002	1
Director Naval Research Laboratory Washington, D.C. 20375	1

<u>Internal Distribution</u>

D	1
D1	1
D2 (D. Snyder)	1
E	1
E211	1
E231	10
F41 (E. Morgan)	1
F411	1
F42 (E. Spooner)	1
G	1
G102 (G. Hornbaker)	1
K	1
K10	1
K104D	20
K106 (Dr. M. Thomas)	2
K106 (Dr. J. Crigler)	1
K106 (G. Gemmill)	1
K107 (R. Mulliken)	1
K33 (A. Morris)	50
K40	1
K43 (R. Coppola)	1
N	1
N302 (S. Vittoria)	1
R44	1
R44 (Dr. A.H. Van Tuyl)	1