FORTRAN SUBROUTINES FOR THE EVALUATION OF THE
CONFLUENT HYPERGEOMETRIC FUNCTIONS

WILLIAM GRAGG
BENY NETA

August 1989

Approved for public release; distribution unlimited
Prepared for:  Naval Postgraduate School
Monterey, CA 93943
FORTRAN SUBROUTINES FOR THE EVALUATION OF THE CONFLUENT HYPERGEOMETRIC FUNCTIONS

Personal Authors:
William Gragg and Benx Neta

Type of Report
Technical Report

Date of Report
89 August 14

Page Count
12

Supplementary Notation

Abstract
In this report we list the Fortran subroutines for evaluating the confluent hypergeometric functions $M(a,b;x)$ and $U(a,b;x)$. These subroutines use the stable recurrence relations given e.g. in Wimp.
Fortran Subroutines for the Evaluation of the Confluent Hypergeometric Functions

W. Gragg
and
B. Neta

Naval Postgraduate School
Department of Mathematics
Monterey, CA  93943
Abstract

In this report we list the Fortran subroutines for evaluating the confluent hypergeometric functions $M(a,b;x)$ and $U(a,b;x)$. These subroutines use the stable recurrence relations given e.g. in Wimp.

Key words:
confluent hypergeometric functions
stable algorithm
Fortran subroutine
recurrence relation
Introduction

It is well known that the ordinary differential equation

\[ x \frac{d^2y}{dx^2} + (1-x) \frac{dy}{dx} - ay = 0 \]

has a solution

\[ y(x) = AM(a,1;x) + BU(a,1;x) \]

if \( a \) is not a negative integer.

This problem arises e.g. when solving the linearized shallow water equations with the full linear variation in depth included (see Williams, Staniforth and Neta, [1]).

The computation of the confluent hypergeometric functions is based on the Miller algorithm (see e.g. Wimp. [2]). In general, one has a second order difference equation

\[ z(n) + a(n)z(n+1) + b(n)z(n+2) = 0, \quad n \geq 0, \quad b(n) \neq 0. \]

If \( b(n) = 0 \) for some \( n \), in some cases one can make a change of variable \( Y(n) = \lambda(n)z(n) \) which will produce an equation of the desired type. Let \( w(n) \) be a nontrivial solution and the sum of the normalizing series

\[ S = \sum_{k=0}^{\infty} c(k)w(k) \neq 0 \]
is known. Let $N$ be a large integer and define $z_N(n), 0 \leq n \leq N+1$, by

$$z_N(n) = \begin{cases} 0 & n = N+1 \\ 1 & n = N \end{cases}$$

$$z_N(n) + a(n)z_N(n+1) + b(n)z_N(n+2) = 0, \quad n = N-1, \ldots, 1, 0.$$

One can approximate $w(n)$ by $w_N(n)$

$$w_N(n) = Sz_N(n)/S_N$$

where

$$S_N = \sum_{k=0}^{N} c(k)z_N(k).$$

The algorithm is said to converge if

$$w_N(n) \to w(n) \text{ as } N \to \infty.$$

The function $M(a,b;x)$ satisfies the recurrence relation

$$(2n+b+2)(n+a)z(n) - (2n+b+1)\left\{ (2a-b) + \frac{(2n+b)(2n+b+2)}{x} \right\} z(n+1)$$

$$- (2n+b)(n+b+1-a)z(n+2) = 0.$$
where

\[(c)_n = \frac{\Gamma(n+c)}{\Gamma(c)} .\]

The normalization relationship used in our subroutine is

\[S = b-1 = \sum_{k=0}^{\infty} \frac{(-1)^k}{k!} (b-1)_k (b+2k-1)w(k) .\]

An obvious modification must be made if \(b = 1\). The algorithm is not defined if \(b, b+1-a, a\) are negative integers or zero.

The function \(U(a,b;x)\) satisfies the relationship

\[(n+a)(n+a+1-b)z(n) - (n+1)[2(n+a+1)+x-b]z(n+1) + (n+1)(n+2)z(n+2) = 0 .\]

The minimal solution is

\[w(n) = \frac{x^n(a)_{n(a+1-b)}}{n!} U(a+n,b;x)\]

for \(|\text{arg } x| < \pi\). A normalization relation is

\[1 = \sum_{k=0}^{\infty} w(n,k) .\]

In the next section we give a listing of the Fortran subroutines.
**Subroutine Miller**

SUBROUTINE MILLER(N, ALPHA, BETA, X, S, SS, COEFF)
INTEGER N
REAL*8 ALPHA, BETA, X, SS
REAL*8 S(0:1000)
EXTERNAL COEFF
C USES THE J.C.P. MILLER ALGORITHM TO COMPUTE
C S(0:N).
C BEGIN
    INTEGER NN, K
    REAL*8 T, D, EPS, A, B, C
    REAL*8 OLDS(0:1000)
    EPS = 0.000000001
    C INITIALIZE OLDS.
    DO 1 K = 0, 1000
        OLDS(K) = 0
    1 CONTINUE
C CHOOSE INITIAL NN.
    NN = N + 2
    C INITIALIZE K, S AND T.
    2 K = NN
    S(K) = 1
    CALL COEFF(K, ALPHA, BETA, X, A, B, C)
    T = 2*C*S(K)
C TAKE A BACKWARD RECURRENCE STEP AND UPDATE IT.
    3 K = K - 1
    CALL COEFF(K, ALPHA, BETA, X, A, B, C)
    S(K) = A*S(K+1) + B*S(K+2)
    C CHECK FOR OVERFLOW AND RESCALE IF NECESSARY.
    D = DABS(S(K))
    IF (D .GT. 1.D30) THEN
    C BEGIN
        CALL SCALE(K, NN, S, T, D)
        END IF
        IF (K .GT. 0) THEN
        C BEGIN
            T = T + 2*C*S(K)
            GO TO 3
        END IF
        T = T + C*S(0)
        DO 4 K = 0, N
            S(K) = S(K)/T
        4 CONTINUE
    C TEMPORARY PRINT STATEMENT.
    C PRINT = S(0)
    C TEST FOR CONVERGENCE.
    D = 0
    DO 5 K = 0, N
    5 D = D + S(K)**2
    CONTINUE
    D = DSQRT(D)
    T = 0
DO 6 K = 0, N
   T = T + (S(K) - OLDS(K))**2
6 CONTINUE
   T = DSQRT(T)
C TAKE ANOTHER STEP IF NO CONVERGENCE.
   IF (T .GT. EPS-D) THEN
C BEGIN
      NN = 2*NN
   DO 7 K = 0, N
      OLDS(K) = S(K)
7 CONTINUE
   IF (NN .LE. 1000) GO TO 2
   PRINT 999, NN, ALPHABET, X, T
999 FORMAT('** NO CONVERGENCE ** NN AP CP X T ', 15, 4E14, 7)
   END IF
   SS = S(0)
   RETURN
END
SUBROUTINE COEFF(N,ALPHA,BETA,X,A,B,C)
INTEGER N
REAL=8 ALPHA,BETA,X,A,B,C
C COMPUTES COEFFICIENTS USED BY J.C.P. MILLER ALGORITHM FOR
C A CONFLUENT HYPERGEOMETRIC FUNCTION M(a,b;x)
C SEE JET WIMP. COMPUTATION WITH RECURRENCE RELATIONS.
C PITMAN 1984 PP. 61-62
C BEGIN
INTEGER M,K
REAL=8 T,U,V,W
S = 2*ALPHA - BETA
T = N + ALPHA
M = 2*N
U = M + BETA
V = U + 1
W = V + 1
A = (S/W + U/X)*V/T
B = (N + BETA - ALPHA + 1)*U/T/W
T = 1
IF (N .GT. 0) THEN
C BEGIN
S = BETA - 1
DO 1 K = 1, N-1
T = -T*(1+S/K)
1 CONTINUE
T = -T*(1+S/M)
END IF
C = T
RETURN
END

SUBROUTINE SCALE(K,N,S,T,D)
INTEGER N,K
REAL=8 T,D
REAL=8 S(0:1000)
C BEGIN
INTEGER J
T = T/D
DO 1 J = K, N
S(J) = S(J)/D
1 CONTINUE
RETURN
END
SUBROUTINE COEFU(N, ALPHA, BETA, X, A, B, C)
INTEGER N
REAL*8 ALPHA, BETA, X, A, B, C
C COMPUTES COEFFICIENTS USED BY J.C.P. MILLER ALGORITHM FOR
C A CONFLUENT HYPERGEOMETRIC FUNCTION \( U(a, b; x) \)
C SEE JET WIMP, COMPUTATION WITH RECURRENCE RELATIONS.
C PITMAN 1984 PP. 63-64
C BEGIN
    INTEGER M, K
    REAL*8 S, T, U, V, W
    S = ALPHA + QFLOAT(N)
    T = S + 1.0D0
    U = S*(T - BETA)
    V = QFLOAT(N + 1)
    W = V + 1.0D0
    A = (2*T + X - BETA)*V/U
    B = - V*W/U
    C = 1
    RETURN
END

Remark: The program that calls Miller must supply as a last parameter either COEFF (for M) or COEFU (for U).
The subroutines are available on a diskette from either author upon request. These subroutines were tested extensively for various values of a, b and x.

**Remark:** If the parameter is a negative integer, the solution of the differential equation is

\[ y = A L_n(x) + B \{ \ln |x| L_n(x) + \sum_{m=0}^{\infty} \beta_m x^m \} \]

where \( n = -a \).

\( L_n(x) \) are Laguerre polynomials whose coefficients \( a_i \) satisfy

\[ a_i = \frac{i-n-1}{i^2} a_{i-1}, \quad i = 2, \ldots, n. \]

\[ a_1 = -n. \]

The coefficients \( \beta_m \) satisfy

\[ \beta_{m+1} = \frac{(m-n) \beta_n + \left( 1 - \frac{2(m-n)}{m+1} a_m \right)}{(m+1)^2} \quad m = 1, \ldots, n-1 \]

\[ \beta_m = \frac{1}{(n-1)^2} a_n \quad m = n \]

\[ \beta_m = \frac{n(n-1)}{m^2} \beta_{m-1} \quad m = n+1, n+2, \ldots. \]
Acknowledgement:

This research was conducted for the Office of Naval Research and was funded by the Naval Postgraduate School.

References


<table>
<thead>
<tr>
<th>Name</th>
<th>No. of Copies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Director</td>
<td>2</td>
</tr>
<tr>
<td>Defense Tech. Information Center</td>
<td></td>
</tr>
<tr>
<td>Cameron Station</td>
<td></td>
</tr>
<tr>
<td>Alexandria, VA 22314</td>
<td></td>
</tr>
<tr>
<td>Director of Research Administration</td>
<td>1</td>
</tr>
<tr>
<td>Code 012</td>
<td></td>
</tr>
<tr>
<td>Naval Postgraduate School</td>
<td></td>
</tr>
<tr>
<td>Monterey, CA 93943</td>
<td></td>
</tr>
<tr>
<td>Library</td>
<td>2</td>
</tr>
<tr>
<td>Code 0142</td>
<td></td>
</tr>
<tr>
<td>Naval Postgraduate School</td>
<td></td>
</tr>
<tr>
<td>Monterey, CA 93943</td>
<td></td>
</tr>
<tr>
<td>Department of Mathematics</td>
<td>1</td>
</tr>
<tr>
<td>Code 53</td>
<td></td>
</tr>
<tr>
<td>Naval Postgraduate School</td>
<td></td>
</tr>
<tr>
<td>Monterey, CA 93943</td>
<td></td>
</tr>
<tr>
<td>Center for Naval Analyses</td>
<td>1</td>
</tr>
<tr>
<td>4401 Ford Avenue</td>
<td></td>
</tr>
<tr>
<td>Alexandria, VA 22302-0268</td>
<td></td>
</tr>
<tr>
<td>Professor Beny Neta</td>
<td>6</td>
</tr>
<tr>
<td>Code 53Nd</td>
<td></td>
</tr>
<tr>
<td>Department of Mathematics</td>
<td></td>
</tr>
<tr>
<td>Naval Postgraduate School</td>
<td></td>
</tr>
<tr>
<td>Monterey, CA 93943</td>
<td></td>
</tr>
<tr>
<td>Dr. C.P. Katti</td>
<td>5</td>
</tr>
<tr>
<td>J. Nehru University</td>
<td></td>
</tr>
<tr>
<td>School of Computer &amp; Systems Sciences</td>
<td></td>
</tr>
<tr>
<td>New Delhi 110067</td>
<td></td>
</tr>
<tr>
<td>India</td>
<td></td>
</tr>
<tr>
<td>Professor Paul Nelson</td>
<td>1</td>
</tr>
<tr>
<td>Texas A&amp;M University</td>
<td></td>
</tr>
<tr>
<td>Department of Nuclear Engineering and Mathematics</td>
<td></td>
</tr>
<tr>
<td>College Station, TX 77843-3133</td>
<td></td>
</tr>
<tr>
<td>Professor H.B. Keller</td>
<td>1</td>
</tr>
<tr>
<td>Department of Applied Mathematics</td>
<td></td>
</tr>
<tr>
<td>California Institute of Technology</td>
<td></td>
</tr>
<tr>
<td>Pasadena, CA 91125</td>
<td></td>
</tr>
</tbody>
</table>