Mathematics Notes

Note 3

7 August 1969

LEGEN

A Subroutine for the Generation of Associated Legendre Functions of the First Kind for Real Arguments Along the Cut

Lawrence M. Berg

Air Force Weapons Laboratory

ABSTRACT

A subroutine for the generation of the Legendre polynomial and the Associated Legendre Functions of the first kind is described from the algorithmic and operational points of view. Methods for the verification of accuracy are discussed. The use of the subroutine to generate the Legendre functions and their derivatives is outlined.

LEGEN

A Subroutine for the Generation of Associated
Legendre Functions of the First Kind for Real Arguments Along the Cut

INTRODUCTION

This note describes a computer subroutine entitled LEGEN, written in FORTRAN IV for the CDC 6600 computer, which calculates the associated Legendre function $P_n^m(x)$. The subroutine accepts as inputs the degree (n), the order (m), and the argument (x) of the function. The subroutine will return values for $P_n^m(x)$ and its derivative, $P_n^{m\prime}(x)$. The order and degree of the function must be TYPE INTEGER. The order must be greater than or equal to zero ($m \geq 0$), and the degree must be greater than zero ($n > 0$). The argument (x) must be TYPE REAL and range along the cut ($-1.0 \leq x \leq 1.0$). All returned parameters should be TYPE DOUBLE to insure accuracy at large m and n.

Several error messages and an optional accuracy check is contained within the subroutine to enable the user to verify returned values.

OPERATION

Generation

LEGEN must be used with a calling routine which specifies the order, and argument of the function and accepts the results from the subroutine.

Calling Sequence

Subroutine LEGEN may be called by the calling routine by using the following CALL statement:

```
CALL LEGEN (N, M, X, P, PP, DE, IDE)
```

The parameters N, M, X and IDE are supplied to the subroutine by the calling routine. They must have been previously assigned before execution of the CALL statement. The other parameters are returned by LEGEN.

Parameters

1. N — TYPE INTEGER. This is the uppermost degree of the Legendre functions desired. N must be greater than 0.

2. M — TYPE INTEGER. This is the order of the Legendre functions to be calculated. M must be greater than or equal to 0.

3. X — TYPE REAL. This is the argument of the Legendre functions. The range of X must be $-1.0 \leq x \leq 1.0$.

4. P TYPE DOUBLE. This is an array of values of $P_n^m(x)$, returned by the subroutine. The order m is the input value M, and remains constant. The degree n ranges from M through the input value N. If M was input as zero then n will range between 1 and N. The argument of the Legendre function is the input value of X. P must be typed and dimensioned by the calling routine.

5. PP TYPE DOUBLE. This is an array of values for $P_n^{m\prime}(x)$, the first derivative of the Legendre functions. The range of n and the values for the parameters m, and x are the same as that of P. PP must be typed and dimensioned by the calling routine.

6. DE TYPE DOUBLE. This is an array of values calculated as an accuracy check, representing a solution to the differential equation.

$$DE = (1-x^2)\ P_n^{m\prime}(x) + nx\ P_n^m(x) - (n+m)\ P_{n-1}^m(x).$$

DE should be zero for an exact answer. DE must be typed and dimensioned by the calling routine.

7. IDE TYPE INTEGER. This integer parameter is used to return and print the array DE. If IDE = 0, the array is calculated and printed, along with the arrays P and PP. If IDE is any value, other than 0, only P and PP are calculated and returned, without any printing taking place.

The names given to the parameters were chosen for illustrative purposes and are formal parameters in the subroutine. The actual parameters used must agree in type and number, but need not be identical in name.

ERROR MESSAGES

Several printed messages are provided for incorrect or out of range input. For an input of $M < 0$ or $N \leq 0$, the following printout will occur:

NEGATIVE ORDER OR DEGREE NOT ACCEPTED IN THE LEGENDRE FUNCTION ROUTINE.

If the argument is out of range the following printout will occur:

ONLY VALUES ON THE CUT ARE ACCEPTED AS ARGUMENTS FOR THE LEGENDRE FUNCTION.

If M exceeds N the following message will be printed:

BY DEFINITION M CANNOT EXCEED N.

If any of the preceding messages are printed, LEGEN will force termination of the program.

ACCURACY CHECK

The parameter IDE when set to zero will cause subroutine CHECK to be called. In CHECK the array DE is calculated, to provide a means of checking the accuracy of the Legendre function and their derivatives. The solution to the equation calculated should be zero for an exact answer. However, caution should be used for large M; because of the magnitude of $P_n^m(x)$ machine roundoff error can occur. Subroutine CHECK is called by using the FORTRAN CALL statement.

```
CALL CHECK (N, M, X, P, PP, DE).
```

The formal parameters are the same as described for subroutine LEGEN. If CHECK is called the subroutine will provide a printout similar to the one illustrated below:

---DIFFERENTIAL EQUATION CHECK---

THE EQUATION USED IS $(1-x*x)\,\frac{dP_n^m(x)}{dX} + nxP_n^m(x) - (n+m)P_{n-1}^m(x) = 0$

THE ORDER OF THE FUNCTION IS 0 AND THE ARGUMENT IS X = 1.000D+00

N	P	DP/DX	DIFF. EQ. CHECK
1	1.0000000000 D+00	1.0000000000	0.
2	1.0000000000 D+00	1.0000000000	1.5146129380 D-28

Using CHECK one has a quick, qualitative method of checking the values of $P_n^m(x)$ and $P_n^{m\prime}(x)$.

ORGANIZATION

LEGEN, as a program, utilizes the subroutine CHECK and various external functions contained in the CDC 6600 FUNCTION library.

External Functions

FLOAT, DSQRT, DABS, MOD

I/O Files

An output file is required by the subroutine.

Timing

Execution of LEGEN depends primarily on the magnitude of the degree N. However when compared to other methods of computing the Legendre functions LEGEN proved to be extremely fast. For example, for M = 0, N = 1000, X = 0.0 (.1) 1.0, the central processor time used on the CDC 6600 was 12.8 seconds.

ALGORITHMS

The Legendre functions of the first kind are computed from forward recurrence relationships. Several different relations are used, depending upon the order M.

Legendre Polynomial, M = 0

Initial values as defined in the Handbook of Mathematical Functions, AMS 55, were used to begin generation of the arrays P and PP.

$$P_1(x) = x,\ P_2(x) = \frac{1}{2}(3x^2-1),\ \frac{dP_1(x)}{dx} = 1.0$$

From these initial values, the forward recurrence relations

$$Pn{+}1(x) = \frac{(2n+1)\ x\ Pn(x) - nPn{-}1(x)}{n+1}$$

and

$$\frac{dPn{+}1(x)}{dx} = (2n+1)\ Pn(x) + \frac{dPn(x)}{dx}$$

are used.

Associated Legendre Functions, M > 0

The initial value to begin the recurrence technique is as follows:

From AMS 55

$$P_n^m(x) = (-1)^m\ (1-x^2)^{1/2m}\ \frac{d^m Pn(x)}{dx^m}$$

therefore

$$P_1^1(x) = -\ (1-x^2)^{1/2};$$

The recurrence relations to generate the associated Legendre functions are as follows:

For equal order and degree, N = M

$$P_m^m(x) = -(2m-1)(1-x^2)^{1/2}\ P_{n-1}^{m-1}(x)$$

For $m \neq n$

$$P_{n+1}^m(x) = \frac{(2n+1)\ x\ P_n^m(x) - (n+m)\ P_{n-1}^m(x)}{n-m+1}$$

$$\frac{dP_n^m(x)}{dx} = \frac{-(nxP_n^m(x) - (n+m)\ P_{n-1}^m(x))}{1-x^2}$$

And, for the special case, at $|x| = 1$

$$\frac{dP^m_{n-1}(x)}{dx} = \frac{(n+m+1)}{(n-m+1)} \frac{dP^m_n(x)}{dx} .$$

When the subroutine CHECKS is called, the differential equation

$$(1-x^2) \frac{dP^m_n(x)}{dx} + nxP^m_n(x) - (n+m) P^m_{n-1}(x) = 0$$

is calculated to verify the values of $P^m_n(x)$ and $\frac{dP^m_n(x)}{dx}$.

SUMMARY

The subroutine LEGEN provides values for the Legendre polynomial, the associated Legendre function and their derivatives for order M, degree M through N and argument x. Accuracy has been verified through the use of tables contained in the references at the end of this note. It should be noted here that the definition of the associated Legendre functions used in this note includes the factor $(-1)^m$, which is omitted in some texts. This is in keeping with the definition as stated in the National Bureau of Standards AMS 55.

ACKNOWLEDGEMENTS

The suggestions and assistance of Mr. Joe P. Martinez of The Dikewood Corporation in the writing of the program and the preparation of this note is gratefully acknowledged.

REFERENCES

1. Handbook of Mathematical Functions, AMS 55, M. Abramowitz and I.A. Stegun, ed., National Bureau of Standards, 1964.

2. Tables of Functions, E. Jahnke and F. Emde, Chapter VII, Dover Publications, New York, Fourth Edition, 1945.

3. Methods of Theoretical Physics, P. Morse and H. Feshbach, Chapter 10, Part II, McGraw-Hill, 1953.

4. Associated Legendre Polynomials, CIC Report C3.4-001, R. L. Pexton, University of California, Computer Information Center, Lawrence Radiation Laboratory, 1965.

5. Spherical and Ellipsoidal Harmonics, E. W. Hobson, Chapter III, Chelsea Publishing Company, New York, Second Reprint, 1965.

6. Spherical Harmonics, T.M. MacRobert, Chapters V, VI, and VII, Pergamon Press, London, Third Edition, 1967.

Appendix I

Methods of Accuracy Verification

The numerical results obtained with subroutine LEGEN were compared with tables contained in the preceding references. The accuracy compared favorably with the tables, having no discrepancies with respect to the number of significant places contained in the tables. A difference in sign at odd-numbered orders (m) was noticed, but this is due to the difference in definition of the associated Legendre functions between reference (1) and the other references. This is the $(-1)^m$ term mentioned before. Unfortunately, existing tables are limited to small order and degree. When testing LEGEN for large m and n results were compared by running the subroutine of reference (4). For small m and n = 100 the two subroutines agreed with each other to eight significant digits and were off at the most by 1 in the ninth significant digit. For n > 100 the Lawrence Radiation Laboratory subroutine caused overflow problems in the CDC computer. For n > 100 the equation of subroutine CHECK is used as an accuracy check. For m = 0 and $1 \leq n \leq 1000$ the solution to the differential equation was in the range of 10^{-26} where the exact solution is 0. For larger m accuracy is reduced. However, even at large m the solution to the differential equation is still zero for certain values. When m > 10, the associated Legendre function $|P_n^m(x)|$ is extremely large and machine roundoff error may be introduced in the computations. At m > 10 absolute accuracy cannot be guaranteed.

Another method employed in checking accuracy was by calculating the functions using a backward recurrence technique and comparing values. If M is the order and N the largest degree desired, then $P_N^M(x)$ and $P_{N-1}^M(x)$ are calculated from the following equations, for starting values.

$$P_n^m(x) = (-1)^m \frac{(n+m)!}{(n-m)!} \frac{(1-x^2)^{m/2}}{2^m m!} \left(\frac{1+x}{2}\right)^{n-m} \sum_{k=0}^{n-m} t_k \left(\frac{x-1}{x+1}\right)^k$$

where

$$t_o = 1, \text{ and } t_k = t_{k-1} \left[\frac{(n-k+1)(n-k+1-m)}{k(m+k)}\right]$$

The backward recurrence formula

$$P_n^m(x) = \frac{(2n+3)x\ P_{n+1}^m(x) - (n-m+2)\ P_{n+2}^m(x)}{n+m+1}$$

is then carried to n = m. This recurrence formula produces numbers which differ from the forward recurrence method only in the 9th or 10th significant place. Again at large m computer roundoff errors are introduced due to the large absolute magnitude of the functions, but for m < 10 the results agree with each other to a high degree of accuracy.

Appendix II. Program Listing

```
      SUBROUTINE LEGEN (N,M,XX,P,PP,DE,IDE)
C
C     THIS ROUTINE CALCULATES THE ASSOCIATED LEGENDRE FUNCTION AND ITS
C     DERIVATIVE FOR ARGUMENTS ON THE CUT.
C
      INTEGER N,M,IDE
      DOUBLE PRECISION PP(N),P(N),OMX2,X,SQT,V,W,EM,G,DE(N)
      X=XX
      IF (-1..GT.X.OR.X.GT.1.) GO TO 350
      IF (N.LE.0.OR.M.LT.0) GO TO 340
      SIGN=1.
      IF (X.LT.0.) SIGN=-1.
      W=FLOAT(M)
      OMX2=(1.-X*X)
      SQT=DSQRT(OMX2)
      N1=N-1
      IF (M.GT.N) GO TO 360
      IF (M.EQ.0) GO TO 200
      IF (.99999998-DABS(X)) 190,10,10
10    P(1)=-SQT
      IF (N.EQ.1.AND.M.EQ.1) GO TO 100
      IF (M.EQ.1) GO TO 30
      DO 20 I=2,M
      EM=FLOAT(I)
      P(I)=-(2.*EM-1.)*SQT*P(I-1)
20    CONTINUE
30    DO 90 J=M,N1
      V=FLOAT(J)
      IF (.00000001-DABS(X)) 50,40,40
40    NM=J+1+M
      IF (MOD(NM,2).EQ.1) GO TO 80
50    IF ((J-1)-M) 70,60,60
60    P(J+1)=((2.*V+1.)*X*P(J)-(V+W)*P(J-1))/(V-W+1.)
      GO TO 90
70    P(J+1)=((2.*V+1.)*X*P(J))/(V-W+1.)
      GO TO 90
80    P(J+1)=0.0
90    CONTINUE
100   DO 160 I=M,N
      V=FLOAT(I)
      IF (.00000001-DABS(X)) 120,110,110
110   NM=I+M
      IF (MOD(NM,2).EQ.0) GO TO 150
120   IF ((I-1)-M) 130,140,140
130   PP(I)=-(V*X*P(I))/OMX2
      GO TO 160
140   PP(I)=-((V*X*P(I)-(V+W)*P(I-1))/OMX2)
      GO TO 160
150   PP(I)=0.
160   CONTINUE
      IF (.99999-DABS(X)) 170,330,330
170   DO 180 K=M,N1
      G=FLOAT(K)
```

```
      P(K)=0.
180   PP(K+1)=((G+W+1.)/(G-W+1.))*PP(K)
      P(N)=0.
      GO TO 330
190   X=.999999*SIGN
      IF (M.LT.10) X=.9999999999*SIGN
      OMX2=(1.-X*X)
      SQT=DSQRT(OMX2)
      GO TO 10
C
C     THE FOLLOWING CALCULATES THE LEGENDRE POLYNOMIAL (ORDER = 0)
C
200   IF (.00000001-DABS(X)) 210,230,230
210   P(1)=X
      PP(2)=3.*X
220   P(2)=.5*(-1.+3.*X*X)
      PP(1)=1.
      GO TO 240
230   P(1)=0.
      PP(2)=0.
      GO TO 220
240   IF (N.EQ.1) GO TO 330
      IF (.99999998-DABS(X)) 310,250,250
250   DO 300 NN=2,N1
      G=FLOAT(NN)
      IF (.00000001-DABS(X)) 270,260,260
260   IF (MOD(NN,2).EQ.0) GO TO 280
      PP(NN+1)=0.
270   P(NN+1)=((2.*G+1.)*X*P(NN)-G*P(NN-1))/(G+1.)
      IF (.00000001-DABS(X)) 290,300,300
280   P(NN+1)=0.
290   PP(NN+1)=(2.*G+1.)*P(NN)+PP(NN-1)
300   CONTINUE
      GO TO 330
310   DO 320 KA=2,N
      P(KA)=1.
      IF (X.LT.0.00) P(KA)=(-1.)**KA
      G=FLOAT(KA)
320   PP(KA+1)=(2.*G+1.)*P(KA)+PP(KA-1)
330   IF (DABS(X).GT..999999) X=SIGN
      IF (IDE.EQ.0) CALL CHECK (N,M,X,P,PP,DE)
      RETURN
340   PRINT 370
      STOP
350   PRINT 380
      STOP
360   PRINT 390
      STOP
C
370   FORMAT (68H NEGATIVE ORDER OR DEGREE NOT ACCEPTED IN LEGENDRE FUNC
     1TION ROUTINE.)
380   FORMAT (76H ONLY VALUES ON THE CUT ARE ACCEPTED AS ARGUMENTS FOR T
     1HE LEGENDRE FUNCTION.)
390   FORMAT (32H BY DEFINITION M CANNOT EXCEED N)
      END
```

```
      SUBROUTINE CHECK (N,M,X,P,PP,DE)
      INTEGER N,M
      DOUBLE PRECISION DE(N),P(N),PP(N),X,G,EM,C,SIGN
      SIGN=1.
      IF (X.LT.0.) SIGN=-SIGN
      EM=FLOAT(M)
      K=M
      C=0.
      IF (M.EQ.0) K=1
      DO 30 I=K,N
      G=FLOAT(I)
      IF (M.EQ.0.AND.I.EQ.1) C=SIGN
      IF(M.EQ.I.OR.(M.EQ.0.AND.I.EQ.1))GO TO 20
10    DE(I)=(1.-X*X)*PP(I)+G*X*P(I)-(G+EM)*P(I-1)
      GO TO 30
20    DE(I)=(1.-X*X)*PP(I)+G*X*P(I)-C
30    CONTINUE
      PRINT 40, M,X
      PRINT 50, (J,P(J),PP(J),DE(J),J=K,N)
      RETURN
C
40    FORMAT (1H1,50X,31H--DIFFERENTIAL EQUATION CHECK--//38X,60H THE EQ
     1UATION USED IS (1.-X*X)*DPN/DX+N*X*PN-(N+M)*PN-1=0.00//28X,31HTHE
     2ORDER OF THE FUNCTION IS,  ,I3,25H  AND THE ARGUMENT IS X =,D12.4
     3//28X,1HN,15X,1HP,24X,5HDP/DX,17X,15HDIFF. EQ. CHECK//)
50    FORMAT (22X,I6,6X,D20.10,6X,D20.10,7X,D20.10)
      END
```