NAG Library Routine Document

D02JAF

Note: before using this routine, please read the Users' Note for your implementation to check the interpretation of ***bold italicised*** terms and other implementation-dependent details.

1 Purpose

D02JAF solves a regular linear two-point boundary value problem for a single nth-order ordinary differential equation by Chebyshev series using collocation and least-squares.

2 Specification

```
SUBROUTINE D02JAF(N, CF, BC, X0, X1, K1, KP, C, W, LW, IW, IFAIL)

INTEGER            N, K1, KP, LW, IW(K1), IFAIL
double precision   CF, X0, X1, C(K1), W(LW)
EXTERNAL           CF, BC
```

3 Description

D02JAF calculates the solution of a regular two-point boundary value problem for a single nth-order linear ordinary differential equation as a Chebyshev series in the interval (x_0, x_1). The differential equation

$$f_{n+1}(x)y^{(n)}(x) + f_n(x)y^{(n-1)}(x) + \cdots + f_1(x)y(x) = f_0(x)$$

is defined by CF, and the boundary conditions at the points x_0 and x_1 are defined by BC.

You specify the degree of Chebyshev series required, K1 $-$ 1, and the number of collocation points, KP. The routine sets up a system of linear equations for the Chebyshev coefficients, one equation for each collocation point and one for each boundary condition. The boundary conditions are solved exactly, and the remaining equations are then solved by a least-squares method. The result produced is a set of coefficients for a Chebyshev series solution of the differential equation on an interval normalized to $(-1, 1)$.

E02AKF can be used to evaluate the solution at any point on the interval (x_0, x_1) – see Section 9 for an example. E02AHF followed by E02AKF can be used to evaluate its derivatives.

4 References

Picken S M (1970) Algorithms for the solution of differential equations in Chebyshev-series by the selected points method *Report Math. 94* National Physical Laboratory

5 Parameters

1: N – INTEGER *Input*

On entry: n, the order of the differential equation.

Constraint: N $\geq$ 1.

2: CF – ***double precision*** FUNCTION, supplied by the user. *External Procedure*

CF defines the differential equation (see Section 3). It must return the value of a function $f_j(x)$ at a given point x, where, for $1 \leq j \leq n+1$, $f_j(x)$ is the coefficient of $y^{(j-1)}(x)$ in the equation, and $f_0(x)$ is the right-hand side.

The specification of CF is:

```
double precision FUNCTION CF(J, X)
INTEGER                    J
double precision           X
```

1: J – INTEGER *Input*

On entry: the index of the function f_j to be evaluated.

2: X – ***double precision*** *Input*

On entry: the point at which f_j is to be evaluated.

CF must be declared as EXTERNAL in the (sub)program from which D02JAF is called. Parameters denoted as *Input* must **not** be changed by this procedure.

3: BC – SUBROUTINE, supplied by the user. *External Procedure*

BC defines the boundary conditions, each of which has the form $y^{(k-1)}(x_1) = s_k$ or $y^{(k-1)}(x_0) = s_k$. The boundary conditions may be specified in any order.

The specification of BC is:

```
SUBROUTINE BC(I, J, RHS)
INTEGER          I, J
double precision RHS
```

1: I – INTEGER *Input*

On entry: the index of the boundary condition to be defined.

2: J – INTEGER *Output*

On exit: must be set to $-k$ if the boundary condition is $y^{(k-1)}(x_0) = s_k$, and to $+k$ if it is $y^{(k-1)}(x_1) = s_k$.

J must not be set to the same value k for two different values of I.

3: RHS – ***double precision*** *Output*

On exit: must be set to the value s_k.

BC must be declared as EXTERNAL in the (sub)program from which D02JAF is called. Parameters denoted as *Input* must **not** be changed by this procedure.

4: X0 – ***double precision*** *Input*

5: X1 – ***double precision*** *Input*

On entry: the left- and right-hand boundaries, x_0 and x_1, respectively.

Constraint: $\mathrm{X1} > \mathrm{X0}$.

6: K1 – INTEGER *Input*

On entry: the number of coefficients to be returned in the Chebyshev series representation of the solution (hence the degree of the polynomial approximation is $\mathrm{K1} - 1$).

Constraint: $\mathrm{K1} \geq \mathrm{N} + 1$.

7: KP – INTEGER *Input*

On entry: the number of collocation points to be used.

Constraint: $\mathrm{KP} \geq \mathrm{K1} - \mathrm{N}$.

8: C(K1) – ***double precision*** array *Output*

On exit: the computed Chebyshev coefficients; that is, the computed solution is:

$$\sum_{i=1}^{\mathrm{K1}}{}' \mathrm{C}(i) T_{i-1}(x)$$

where $T_i(x)$ is the ith Chebyshev polynomial of the first kind, and $\sum'$ denotes that the first coefficient, C(1), is halved.

9: W(LW) – ***double precision*** array *Workspace*

10: LW – INTEGER *Input*

On entry: the dimension of the array W as declared in the (sub)program from which D02JAF is called.

Constraint: $\mathrm{LW} \geq 2 \times (\mathrm{KP} + \mathrm{N}) \times (\mathrm{K1} + 1) + 7 \times \mathrm{K1}$.

11: IW(K1) – INTEGER array *Workspace*

12: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this parameter you should refer to Section 3.3 in the Essential Introduction for details.

On exit: IFAIL $= 0$ unless the routine detects an error (see Section 6).

For environments where it might be inappropriate to halt program execution when an error is detected, the value -1 or 1 is recommended. If the output of error messages is undesirable, then the value 1 is recommended. Otherwise, if you are not familiar with this parameter, the recommended value is 0. **When the value -1 or 1 is used it is essential to test the value of IFAIL on exit.**

6 Error Indicators and Warnings

If on entry IFAIL $= 0$ or -1, explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors or warnings detected by the routine:

IFAIL $= 1$

On entry, $\mathrm{N} < 1$,
or $\mathrm{X0} \geq \mathrm{X1}$,
or $\mathrm{K1} < \mathrm{N} + 1$,
or $\mathrm{KP} < \mathrm{K1} - \mathrm{N}$.

IFAIL $= 2$

On entry, $\mathrm{LW} < 2 \times (\mathrm{KP} + \mathrm{N}) \times (\mathrm{K1} + 1) + 7 \times \mathrm{K1}$ (insufficient workspace).

IFAIL $= 3$

Either the boundary conditions are not linearly independent (that is, in BC the variable J is set to the same value k for two different values of I), or the rank of the matrix of equations for the coefficients is less than the number of unknowns. Increasing KP may overcome this latter problem.

IFAIL $= 4$

The least-squares routine F04AMF has failed to correct the first approximate solution (see F04AMF).

7 Accuracy

The Chebyshev coefficients are determined by a stable numerical method. The accuracy of the approximate solution may be checked by varying the degree of the polynomial and the number of collocation points (see Section 8).

8 Further Comments

The time taken by D02JAF depends on the complexity of the differential equation, the degree of the polynomial solution, and the number of matching points.

The collocation points in the interval (x_0, x_1) are chosen to be the extrema of the appropriate shifted Chebyshev polynomial. If KP $=$ K1 $-$ N, then the least-squares solution reduces to the solution of a system of linear equations, and true collocation results.

The accuracy of the solution may be checked by repeating the calculation with different values of K1 and with KP fixed but KP $\gg$ K1 $-$ N. If the Chebyshev coefficients decrease rapidly (and consistently for various K1 and KP), the size of the last two or three gives an indication of the error. If the Chebyshev coefficients do not decay rapidly, it is likely that the solution cannot be well-represented by Chebyshev series. Note that the Chebyshev coefficients are calculated for the interval $(-1, 1)$.

Systems of regular linear differential equations can be solved using D02JBF. It is necessary before using D02JBF to write the differential equations as a first-order system. Linear systems of high-order equations in their original form, singular problems, and, indirectly, nonlinear problems can be solved using D02TGF.

9 Example

This example solves the equation

$$y'' + y = 1$$

with boundary conditions

$$y(-1) = y(1) = 0.$$

We use K1 $= 4$, 6 and 8, and KP $= 10$ and 15, so that the different Chebyshev series may be compared. The solution for K1 $= 8$ and KP $= 15$ is evaluated by E02AKF at nine equally spaced points over the interval $(-1, 1)$.

9.1 Program Text

```
*     D02JAF Example Program Text
*     Mark 15 Revised. NAG Copyright 1991.
*     .. Parameters ..
      INTEGER          N, K1MAX, KPMAX, LW
      PARAMETER        (N=2,K1MAX=8,KPMAX=15,LW=2*(KPMAX+N)*(K1MAX+1)
     +                 +7*K1MAX)
      INTEGER          NOUT
      PARAMETER        (NOUT=6)
*     .. Local Scalars ..
      DOUBLE PRECISION X, X0, X1, Y
      INTEGER          I, IA1, IFAIL, K1, KP, M
*     .. Local Arrays ..
      DOUBLE PRECISION C(K1MAX), W(LW)
      INTEGER          IW(K1MAX)
*     .. External Functions ..
      DOUBLE PRECISION CF
      EXTERNAL         CF
*     .. External Subroutines ..
      EXTERNAL         BC, D02JAF, E02AKF
```

```
*     .. Intrinsic Functions ..
      INTRINSIC        DBLE
*     .. Executable Statements ..
      WRITE (NOUT,*) 'D02JAF Example Program Results'
      X0 = -1.0D0
      X1 = 1.0D0
      WRITE (NOUT,*)
      WRITE (NOUT,*) ' KP  K1   Chebyshev coefficients'
      DO 40 KP = 10, KPMAX, 5
         DO 20 K1 = 4, K1MAX, 2
            IFAIL = 1
*
            CALL D02JAF(N,CF,BC,X0,X1,K1,KP,C,W,LW,IW,IFAIL)
*
            IF (IFAIL.EQ.0) THEN
               WRITE (NOUT,99998) KP, K1, (C(I),I=1,K1)
            ELSE
               IF (IFAIL.GT.0) THEN
                  WRITE (NOUT,99999) KP, K1, IFAIL
               ELSE
                  WRITE (NOUT,99995) IFAIL
               END IF
               GO TO 80
            END IF
   20    CONTINUE
   40 CONTINUE
      K1 = 8
      M = 9
      IA1 = 1
      WRITE (NOUT,*)
      WRITE (NOUT,99997) 'Last computed solution evaluated at', M,
     +  ' equally spaced points'
      WRITE (NOUT,*)
      WRITE (NOUT,*) '      X         Y'
      DO 60 I = 1, M
         X = (X0*DBLE(M-I)+X1*DBLE(I-1))/DBLE(M-1)
         IFAIL = 1
*
         CALL E02AKF(K1,X0,X1,C,IA1,K1MAX,X,Y,IFAIL)
*
         WRITE (NOUT,99996) X, Y
   60 CONTINUE
   80 CONTINUE
*
99999 FORMAT (1X,2(I3,1X),'  D02JAF fails with IFAIL =',I4)
99998 FORMAT (1X,2(I3,1X),8F8.4)
99997 FORMAT (1X,A,I5,A)
99996 FORMAT (1X,2F10.4)
99995 FORMAT (1X,/1X,' ** D02JAF returned with IFAIL = ',I5)
      END
*
      DOUBLE PRECISION FUNCTION CF(J,X)
*     .. Scalar Arguments ..
      DOUBLE PRECISION X
      INTEGER          J
*     .. Executable Statements ..
      IF (J.EQ.2) THEN
         CF = 0.0D0
      ELSE
         CF = 1.0D0
      END IF
      RETURN
      END
*
      SUBROUTINE BC(I,J,RHS)
*     .. Scalar Arguments ..
      DOUBLE PRECISION RHS
      INTEGER          I, J
*     .. Executable Statements ..
      RHS = 0.0D0
      IF (I.EQ.1) THEN
```

```
      J = 1
   ELSE
      J = -1
   END IF
   RETURN
   END
```

9.2 Program Data

None.

9.3 Program Results

```
D02JAF Example Program Results

 KP  K1   Chebyshev coefficients
 10   4  -0.6108  0.0000  0.3054  0.0000
 10   6  -0.8316  0.0000  0.4246  0.0000 -0.0088  0.0000
 10   8  -0.8325  0.0000  0.4253  0.0000 -0.0092  0.0000  0.0001  0.0000
 15   4  -0.6174  0.0000  0.3087  0.0000
 15   6  -0.8316  0.0000  0.4246  0.0000 -0.0088  0.0000
 15   8  -0.8325  0.0000  0.4253  0.0000 -0.0092  0.0000  0.0001  0.0000

Last computed solution evaluated at    9 equally spaced points

      X          Y
   -1.0000    0.0000
   -0.7500   -0.3542
   -0.5000   -0.6242
   -0.2500   -0.7933
    0.0000   -0.8508
    0.2500   -0.7933
    0.5000   -0.6242
    0.7500   -0.3542
    1.0000    0.0000
```