NAG Fortran Library Routine Document

C05PDF/C05PDA

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

C05PDF/C05PDA is a comprehensive reverse communication routine to find a solution of a system of nonlinear equations by a modification of the Powell hybrid method. The user must provide the Jacobian.

C05PDA is a version of C05PDF that has additional parameters in order to make it safe for use in multithreaded applications (see Section 5 below).

2 Specifications

2.1 Specification for C05PDF

```
      SUBROUTINE C05PDF(IREVCM, N, X, FVEC, FJAC, LDFJAC, XTOL, DIAG, MODE,
     1                  FACTOR, R, LR, QTF, W, IFAIL)
      INTEGER           IREVCM, N, LDFJAC, MODE, LR, IFAIL
      real              X(N), FVEC(N), FJAC(LDFJAC,N), XTOL, DIAG(N), FACTOR,
     1                  R(LR), QTF(N), W(N,4)
```

2.2 Specification for C05PDA

```
      SUBROUTINE C05PDA(IREVCM, N, X, FVEC, FJAC, LDFJAC, XTOL, DIAG, MODE,
     1                  FACTOR, R, LR, QTF, W, LWSAV, IWSAV, RWSAV, IFAIL)
      INTEGER           IREVCM, N, LDFJAC, MODE, LR, IWSAV(15), IFAIL
      real              X(N), FVEC(N), FJAC(LDFJAC,N), XTOL, DIAG(N), FACTOR,
     1                  R(LR), QTF(N), W(N,4), RWSAV(10)
      LOGICAL           LWSAV(5)
```

3 Description

The system of equations is defined as:

$$f_i(x_1, x_2, \ldots, x_n) = 0, \quad \text{for} \quad i = 1, 2, \ldots, n.$$

C05PDF/C05PDA is based upon the MINPACK routine HYBRJ (Moré *et al.* (1980)). It chooses the correction at each step as a convex combination of the Newton and scaled gradient directions. Under reasonable conditions this guarantees global convergence from starting points far from the solution and a fast rate of convergence. The Jacobian is updated by the rank-1 method of Broyden. The Jacobian is requested to be supplied at the start of the computations, but it is not requested again. For more details see Powell (1970).

4 References

Moré J J, Garbow B S and Hillstrom K E (1980) User guide for MINPACK-1 *Technical Report ANL-80-74* Argonne National Laboratory

Powell M J D (1970) A hybrid method for nonlinear algebraic equations *Numerical Methods for Nonlinear Algebraic Equations* (ed P Rabinowitz) Gordon and Breach

5 Parameters

Note: this routine uses **reverse communication.** Its use involves an initial entry, intermediate exits and re-entries, and a final exit, as indicated by the **parameter IREVCM**. Between intermediate exits and re-entries, **all parameters other than FVEC and FJAC must remain unchanged**.

1: IREVCM – INTEGER *Input/Output*

On initial entry: IREVCM must have the value 0.

On intermediate exit: IREVCM specifies what action the user must take before re-entering C05PDF/C05PDA **with** IREVCM **unchanged**. The value of IREVCM should be interpreted as follows:

IREVCM = 1

indicates the start of a new iteration. No action is required by the user but X and FVEC are available for printing.

IREVCM = 2

indicates that before re-entry to C05PDF/C05PDA, FVEC must contain the function value $f_i(x)$.

IREVCM = 3

indicates that before re-entry to C05PDF/C05PDA, FJAC$(i)j$ must contain the value of $\frac{\partial f_i}{\partial x_j}$ at the point x, for $i, j = 1, 2, \ldots, n$.

On final exit: IREVCM = 0, and the algorithm has terminated.

Constraint: IREVCM = 0, 1, 2 or 3.

2: N – INTEGER *Input*

On initial entry: the number of equations, n.

Constraint: N > 0.

3: X(N) – ***real*** array *Input/Output*

On initial entry: an initial guess at the solution vector.

On intermediate exit: X contains the current point.

On final exit: the final estimate of the solution vector.

4: FVEC(N) – ***real*** array *Input/Output*

On initial entry: FVEC need not be set.

On intermediate re-entry: if IREVCM $\neq$ 2, FVEC must not be changed. If IREVCM = 2, FVEC must be set to the values of the functions computed at the current point X.

On final exit: the function values at the final point, X.

5: FJAC(LDFJAC,N) – ***real*** array *Input/Output*

On initial entry: FJAC must be set to the values of the Jacobian evaluated at the initial point X.

On intermediate re-entry: if IREVCM $\neq$ 3, FJAC must not be changed. If IREVCM = 3, FJAC must be set to the value of the Jacobian computed at the current point X.

On final exit: the orthogonal matrix Q produced by the QR factorization of the final approximate Jacobian.

6: LDFJAC – INTEGER *Input*

On initial entry: the first dimension of the array FJAC as declared in the (sub)program from which C05PDF/C05PDA is called.

Constraint: LDFJAC $\geq$ N.

7: XTOL – ***real*** *Input*

On initial entry: the accuracy in X to which the solution is required.

Suggested value: the square root of the ***machine precision***.

Constraint: XTOL ≥ 0.0.

8: DIAG(N) – ***real*** array *Input/Output*

On initial entry: if MODE $= 2$ (see below), DIAG must contain multiplicative scale factors for the variables.

Constraint: DIAG(i) > 0.0 for $i = 1, 2, \ldots, n$.

On intermediate exit: the scale factors actually used (computed internally if MODE $\neq 2$).

9: MODE – INTEGER *Input*

On initial entry: indicates whether or not the user has provided scaling factors in DIAG. If MODE $= 2$ the scale factors must be supplied in DIAG. Otherwise, the variables will be scaled internally.

10: FACTOR – ***real*** *Input*

On initial entry: a quantity to be used in determining the initial step bound. In most cases, FACTOR should lie between 0.1 and 100.0. (The step bound is FACTOR $\times \|$DIAG $\times$ X$\|_2$ if this is non-zero; otherwise the bound is FACTOR.)

Suggested value: FACTOR $= 100.0$.

Constraint: FACTOR > 0.0.

11: R(LR) – ***real*** array *Input/Output*

On intermediate exit: R should not be changed.

On final exit: the upper triangular matrix R produced by the QR factorization of the final approximate Jacobian, stored row-wise.

12: LR – INTEGER *Input*

On initial entry: the dimension of the array R as declared in the (sub)program from which C05PDF/C05PDA is called.

Constraint: LR $\geq$ N $\times$ (N $+ 1)/2$.

13: QTF(N) – ***real*** array *Input/Output*

On intermediate exit: QTF should not be changed.

On final exit: the vector $Q^T f$.

14: W(N,4) – ***real*** array *Workspace*

15: IFAIL – INTEGER *Input/Output*

Note: *for C05PDA, IFAIL does not occur in this position in the parameter list. See the additional parameters described below.*

On entry: IFAIL must be set to 0, -1 or 1. Users who are unfamiliar with this parameter should refer to Chapter P01 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, because for this routine the values of the output parameters may be useful even if IFAIL $\neq 0$ on exit, the recommended value is -1. **When the value -1 or 1 is used it is essential to test the value of IFAIL on exit.**

Note: *the following are additional parameters for specific use with C05PDA. Users of C05PDF therefore need not read the remainder of this section.*

15: LWSAV(5) – LOGICAL array *Workspace*
16: IWSAV(15) – INTEGER array *Workspace*
17: RWSAV(10) – ***real*** array *Workspace*

The arrays LWSAV, IWSAV and RWSAV **must not** be altered between calls to C05PDA.

18: IFAIL – INTEGER *Input/Output*

Note: see the parameter description for IFAIL above.

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, N ≤ 0,
or XTOL < 0.0,
or FACTOR ≤ 0.0,
or LDFJAC $\leq$ N,
or LR $\leq$ N $\times$ (N $+ 1)/2$,
or MODE $= 2$ and DIAG$(i) \leq 0.0$ for some i, $i = 1, 2, \ldots,$ N.

IFAIL $= 2$

On entry, IREVCM < 0 or IREVCM > 3.

IFAIL $= 3$

No further improvement in the approximate solution X is possible; XTOL is too small.

IFAIL $= 4$

The iteration is not making good progress, as measured by the improvement from the last 5 Jacobian evaluations.

IFAIL $= 5$

The iteration is not making good progress, as measured by the improvement from the last 10 iterations.

The values IFAIL $= 4$ and IFAIL $= 5$ may indicate that the system does not have a zero, or that the solution is very close to the origin (see Section 7). Otherwise, rerunning C05PDF/C05PDA from a different starting point may avoid the region of difficulty.

7 Accuracy

If $\hat{x}$ is the true solution and D denotes the diagonal matrix whose entries are defined by the array DIAG then C05PDF/C05PDA tries to ensure that

$$\|D(x - \hat{x})\|_2 \leq \text{XTOL} \times \|D\hat{x}\|_2.$$

If this condition is satisfied with XTOL $= 10^{-k}$, then the larger components of Dx have k significant decimal digits. There is a danger that the smaller components of Dx may have large relative errors, but the fast rate of convergence of C05PDF/C05PDA usually avoids this possibility.

If XTOL is less than ***machine precision*** and the above test is satisfied with the ***machine precision*** in place of XTOL, then the routine exits with IFAIL $= 3$.

Note that this convergence test is based purely on relative error, and may not indicate convergence if the solution is very close to the origin.

The test assumes that the functions and the Jacobian are coded consistently and that the functions are reasonably well behaved. If these conditions are not satisfied then C05PDF/C05PDA may incorrectly indicate convergence. The coding of the Jacobian can be checked using C05ZAF. If the Jacobian is coded correctly, then the validity of the answer can be checked by rerunning C05PDF/C05PDA with a tighter tolerance.

8 Further Comments

The time required by C05PDF/C05PDA to solve a given problem depends on n, the behaviour of the functions, the accuracy requested and the starting point. The number of arithmetic operations executed by C05PDF/C05PDA is about $11.5 \times n^2$ to process each evaluation of the functions and about $1.3 \times n^3$ to process each evaluation of the Jacobian. The timing of C05PDF/C05PDA is strongly influenced by the time spent in the evaluation of the functions and the Jacobian.

Ideally the problem should be scaled so that at the solution the function values are of comparable magnitude.

9 Example

To determine the values $x_1, \ldots, x_9$ which satisfy the tridiagonal equations:

$$(3 - 2x_1)x_1 - 2x_2 = -1,$$
$$-x_{i-1} + (3 - 2x_i)x_i - 2x_{i+1} = -1, \quad i = 2, 3, \ldots, 8$$
$$-x_8 + (3 - 2x_9)x_9 = -1.$$

9.1 Program Text

Note: the listing of the example program presented below uses ***bold italicised*** terms to denote precision-dependent details. Please read the Users' Note for your implementation to check the interpretation of these terms. As explained in the Essential Introduction to this manual, the results produced may not be identical for all implementations.

```
*     C05PDF Example Program Text
*     Mark 14 Release.  NAG Copyright 1989.
*     .. Parameters ..
      INTEGER          N, LDFJAC, LR
      PARAMETER        (N=9,LDFJAC=N,LR=(N*(N+1))/2)
      INTEGER          NOUT
      PARAMETER        (NOUT=6)
      real             ZERO, ONE, TWO, THREE, FOUR
      PARAMETER        (ZERO=0.0e0,ONE=1.0e0,TWO=2.0e0,THREE=3.0e0,
     +                 FOUR=4.0e0)
*     .. Local Scalars ..
      real             FACTOR, FNORM, XTOL
      INTEGER          ICOUNT, IFAIL, IREVCM, J, K, MODE
*     .. Local Arrays ..
      real             DIAG(N), FJAC(LDFJAC,N), FVEC(N), QTF(N), R(LR),
     +                 W(N,4), X(N)
*     .. External Functions ..
      real             F06EJF, X02AJF
      EXTERNAL         F06EJF, X02AJF
*     .. External Subroutines ..
      EXTERNAL         C05PDF
*     .. Intrinsic Functions ..
      INTRINSIC        SQRT
*     .. Executable Statements ..
      WRITE (NOUT,*) 'C05PDF Example Program Results'
*     The following starting values provide a rough solution.
      DO 20 J = 1, N
         X(J) = -1.0e0
   20 CONTINUE
      XTOL = SQRT(X02AJF())
      DO 40 J = 1, N
```

```
            DIAG(J) = 1.0e0
   40 CONTINUE
      MODE = 2
      FACTOR = 100.0e0
      ICOUNT = 0
      IFAIL = 1
      IREVCM = 0
*
   60 CALL C05PDF(IREVCM,N,X,FVEC,FJAC,LDFJAC,XTOL,DIAG,MODE,FACTOR,R,
     +            LR,QTF,W,IFAIL)
*
      IF (IREVCM.EQ.1) THEN
         ICOUNT = ICOUNT + 1
*        Insert print statements here to monitor progess if desired
         GO TO 60
      ELSE IF (IREVCM.EQ.2) THEN
*        Evaluate functions at current point
         DO 80 K = 1, N
            FVEC(K) = (THREE-TWO*X(K))*X(K) + ONE
            IF (K.GT.1) FVEC(K) = FVEC(K) - X(K-1)
            IF (K.LT.N) FVEC(K) = FVEC(K) - TWO*X(K+1)
   80    CONTINUE
         GO TO 60
      ELSE IF (IREVCM.EQ.3) THEN
*        Evaluate Jacobian at current point
         DO 120 K = 1, N
            DO 100 J = 1, N
               FJAC(K,J) = ZERO
  100       CONTINUE
            FJAC(K,K) = THREE - FOUR*X(K)
            IF (K.NE.1) FJAC(K,K-1) = -ONE
            IF (K.NE.N) FJAC(K,K+1) = -TWO
  120    CONTINUE
         GO TO 60
      END IF
*
      WRITE (NOUT,*)
      IF (IFAIL.EQ.0) THEN
         FNORM = F06EJF(N,FVEC,1)
         WRITE (NOUT,99999) 'Final 2 norm of the residuals after',
     +     ICOUNT, ' iterations is ', FNORM
         WRITE (NOUT,*)
         WRITE (NOUT,*) 'Final approximate solution'
         WRITE (NOUT,99998) (X(J),J=1,N)
      ELSE
         WRITE (NOUT,99999) 'IFAIL =', IFAIL
         IF (IFAIL.GT.2) THEN
            WRITE (NOUT,*) 'Approximate solution'
            WRITE (NOUT,99998) (X(J),J=1,N)
         END IF
      END IF
      STOP
*
99999 FORMAT (1X,A,I4,A,e12.4)
99998 FORMAT (5X,3F12.4)
      END
```

9.2 Program Data

None.

9.3 Program Results

```
C05PDF Example Program Results

Final 2 norm of the residuals after  11 iterations is   0.1193E-07

Final approximate solution
         -0.5707     -0.6816     -0.7017
         -0.7042     -0.7014     -0.6919
         -0.6658     -0.5960     -0.4164
```
