# NAG 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 that finds a solution of a system of nonlinear equations by a modification of the Powell hybrid method. You 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).

## 2 Specification

### 2.1 Specification for C05PDF

```
SUBROUTINE C05PDF (IREVCM, N, X, FVEC, FJAC, LDFJAC, XTOL, DIAG, MODE,         &
                   FACTOR, R, LR, QTF, W, IFAIL)
INTEGER            IREVCM, N, LDFJAC, MODE, LR, IFAIL
REAL (KIND=nag_wp) X(N), FVEC(N), FJAC(LDFJAC,N), XTOL, DIAG(N), FACTOR,       &
                   R(N*(N+1)/2), QTF(N), W(N,4)
```

### 2.2 Specification for C05PDA

```
SUBROUTINE C05PDA (IREVCM, N, X, FVEC, FJAC, LDFJAC, XTOL, DIAG, MODE,         &
                   FACTOR, R, LR, QTF, W, LWSAV, IWSAV, RWSAV, IFAIL)
INTEGER            IREVCM, N, LDFJAC, MODE, LR, IWSAV(15), IFAIL
REAL (KIND=nag_wp) X(N), FVEC(N), FJAC(LDFJAC,N), XTOL, DIAG(N), FACTOR,       &
                   R(N*(N+1)/2), QTF(N), W(N,4), RWSAV(10)
LOGICAL            LWSAV(2)
```

## 3 Description

The system of equations is defined as:

$$f_i(x_1, x_2, \ldots, x_n) = 0, \qquad i = 1, 2, \ldots, n.$$

C05PDF/C05PDA is based on the MINPACK routine HYBRJ (see Moré *et al.* (1980)). It chooses the correction at each step as a convex combination of the Newton and scaled gradient directions. The Jacobian is updated by the rank-1 method of Broyden. 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*: must have the value 0.

*On intermediate exit*: specifies what action you 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 you, but X and FVEC are available for printing.

IREVCM = 2
: Indicates that before re-entry to C05PDF/C05PDA, FVEC must contain the function values $f_i(x)$.

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 $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, n$.

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

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

2: N – INTEGER *Input*

*On initial entry*: $n$, the number of equations.

*Constraint*: N > 0.

3: X(N) – REAL (KIND=nag_wp) array *Input/Output*

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

*On intermediate exit*: contains the current point.

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

4: FVEC(N) – REAL (KIND=nag_wp) array *Input/Output*

*On initial entry*: need not be set.

*On intermediate re-entry*: if IREVCM = 1, 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 (KIND=nag_wp) array *Input/Output*

*On initial entry*: 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 (KIND=nag_wp) *Input*

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

*Suggested value*: $\sqrt{\epsilon}$, where $\epsilon$ is the ***machine precision*** returned by X02AJF.

*Constraint*: XTOL $\geq 0.0$.

*On initial entry*: indicates whether or not you have provided scaling factors in DIAG.

If MODE $= 2$ the scaling must have been supplied in DIAG.

Otherwise, the variables will be scaled internally.

8: DIAG(N) – REAL (KIND=nag_wp) array *Input/Output*

*On initial entry*: if MODE $= 2$, 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 you have provided scaling factors in DIAG.

If MODE $= 2$ the scaling must have been supplied in DIAG.

Otherwise, the variables will be scaled internally.

10: FACTOR – REAL (KIND=nag_wp) *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 nonzero; otherwise the bound is FACTOR.)

*Suggested value*: FACTOR $= 100.0$.

*Constraint*: FACTOR $> 0.0$.

11: R(N $\times$ (N $+ 1)/2$) – REAL (KIND=nag_wp) array *Input/Output*

*On initial entry*: need not be set.

*On intermediate exit*: must 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 *Dummy*

This parameter is no longer accessed by C05PDF/C05PDA.

13: QTF(N) – REAL (KIND=nag_wp) array *Input/Output*

*On initial entry*: need not be set.

*On intermediate exit*: must not be changed.

*On final exit*: the vector $Q^{\mathrm{T}} f$.

14: W(N,4) – REAL (KIND=nag_wp) array *Communication Array*

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 initial 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.

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.**

*On final exit*: IFAIL $= 0$ unless the routine detects an error or a warning has been flagged (see Section 6).

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

15: LWSAV(2) – LOGICAL array *Communication Array*
16: IWSAV(15) – INTEGER array *Communication Array*
17: RWSAV(10) – REAL (KIND=nag_wp) array *Communication Array*

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 $\leq 0$,
or XTOL $< 0.0$,
or FACTOR $\leq 0.0$,
or LDFJAC $<$ N,
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 five Jacobian evaluations.

IFAIL $= 5$

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

A value of IFAIL $= 4$ or 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 obviates 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:** 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 lower value for XTOL.

# 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

This example determines the values $x_1, \ldots, x_9$ which satisfy the tridiagonal equations:

$$\begin{aligned} (3 - 2x_1)x_1 - 2x_2 &= -1, \\ -x_{i-1} + (3 - 2x_i)x_i - 2x_{i+1} &= -1, \qquad i = 2, 3, \ldots, 8 \\ -x_8 + (3 - 2x_9)x_9 &= -1. \end{aligned}$$

## 9.1 Program Text

```
    Program c05pdfe

!     C05PDF Example Program Text

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: c05pdf, dnrm2, nag_wp, x02ajf
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter                :: n = 9, nout = 6
      Integer, Parameter                :: ldfjac = n
!     .. Local Scalars ..
      Real (Kind=nag_wp)                :: factor, fnorm, xtol
      Integer                           :: icount, ifail, irevcm, j, k, lr, mode
```

```
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: diag(:), fjac(:,:), fvec(:), qtf(:), &
                                          r(:), w(:,:), x(:)
!     .. Intrinsic Procedures ..
      Intrinsic                        :: sqrt
!     .. Executable Statements ..
      Write (nout,*) 'C05PDF Example Program Results'

      Allocate (diag(n),fjac(ldfjac,n),fvec(n),qtf(n),r(n*(n+ &
        1)/2),w(n,4),x(n))

!     The following starting values provide a rough solution.

      x(1:n) = -1.0E0_nag_wp

      xtol = sqrt(x02ajf())
      diag(1:n) = 1.0E0_nag_wp
      mode = 2
      factor = 100.0E0_nag_wp
      icount = 0
      irevcm = 0
      ifail = -1

revcomm: Do

        Call c05pdf(irevcm,n,x,fvec,fjac,ldfjac,xtol,diag,mode,factor,r,lr, &
          qtf,w,ifail)

        Select Case (irevcm)
        Case (1)
          icount = icount + 1

!         Insert print statements here to monitor progess if desired

          Cycle revcomm
        Case (2)

!         Evaluate functions at current point

          fvec(1:n) = (3.0E0_nag_wp-2.0E0_nag_wp*x(1:n))*x(1:n) + 1.0E0_nag_wp
          fvec(2:n) = fvec(2:n) - x(1:(n-1))
          fvec(1:(n-1)) = fvec(1:(n-1)) - 2.0E0_nag_wp*x(2:n)
          Cycle revcomm
        Case (3)

!         Evaluate Jacobian at current point

          fjac(1:n,1:n) = 0.0E0_nag_wp

          Do k = 1, n
            fjac(k,k) = 3.0E0_nag_wp - 4.0E0_nag_wp*x(k)

            If (k/=1) Then
              fjac(k,k-1) = -1.0E0_nag_wp
            End If

            If (k/=n) Then
              fjac(k,k+1) = -2.0E0_nag_wp
            End If

          End Do

          Cycle revcomm
        Case Default
          Exit revcomm
        End Select

      End Do revcomm

      Select Case (ifail)
      Case (0)
```

```
!       The NAG name equivalent of dnrm2 is f06ejf
        fnorm = dnrm2(n,fvec,1)
        Write (nout,*)
        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)
      Case (3:)
        Write (nout,*)
        Write (nout,*) 'Approximate solution'
        Write (nout,99998)(x(j),j=1,n)
      End Select

99999 Format (1X,A,I4,A,E12.4)
99998 Format (5X,3F12.4)
    End Program c05pdfe
```

### 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
```