



1 SUMMARY

To find the minimum of a function $F(\mathbf{x})$ of many variables x_1, x_2, \dots, x_n , given that the gradient vector $(\partial F/\partial x_1, \partial F/\partial x_2, \dots, \partial F/\partial x_n)_x$ can be calculated, the routine is for use in large problems where storage is at a premium.

ATTRIBUTES — **Version:** 1.0.0. **Types:** Real (single, double). **Calls:** SDOT/DDOT. **Original date:** January 1972. **Origin:** R. Fletcher, Harwell.

2 HOW TO USE THE PACKAGE

2.1 The argument list and calling sequence

The single precision version

```
CALL VA08A(FUNCT, N, X, F, G, S, DFN, EPS, MAXFN, IPRINT)
```

The double precision version

```
CALL VA08AD(FUNCT, N, X, F, G, S, DFN, EPS, MAXFN, IPRINT)
```

FUNCT the identifier of a user subroutine – see section 3.

N an INTEGER which must be set to the number of independent variables n .

X a REAL (DOUBLE PRECISION in the D version) array of N elements in which must be set an initial approximation to the variables x which minimize $F(\mathbf{x})$. The best approximation to the solution will be returned in X on exit.

F a REAL (DOUBLE PRECISION in the D version) number in which the best value of $F(\mathbf{x})$ corresponding to X above will be returned.

G a REAL (DOUBLE PRECISION in the D version) array of N elements in which the gradient vector evaluated at X above will be returned.

S a REAL (DOUBLE PRECISION in the D version) array of N elements used as working space.

DFN a REAL (DOUBLE PRECISION in the D version) number which must be set so as to give VA08A an estimate of the likely reduction to be obtained in $F(\mathbf{x})$. DFN is used only on the first iteration so an order of magnitude estimate will suffice. The information can be provided in different ways, depending upon the sign of DFN which should be set in one of the following ways:

DFN > 0: the setting of DFN itself will be taken as the likely reduction to be obtained in $F(\mathbf{x})$.

DFN=0: it will be assumed that an estimate of the minimum value of $F(\mathbf{x})$ has been set in the argument F , and the likely reduction in $F(\mathbf{x})$ will be computed according to the initial function value.

DFN < 0: a multiple $|DFN|$ of the modulus of the initial function value will be taken as a likely estimate of the reduction.

EPS a real (DOUBLE PRECISION in the D version) number giving the accuracy required to every element of X . The method is such that the ultimate rate of convergence may not be very rapid, so EPS should be set on the small side – say $\frac{1}{10}$ of that desired.

MAXFN an INTEGER giving the maximum number of calls of **FUNCT** allowed.

IPRINT an INTEGER controlling printing. Printing occurs every $|IPRINT|$ iterations, and also on exit, in the form

Iteration No, No. of calls of **FUNCT**,

Function value

$X(1), X(2), \dots, X(N)$ 8 to a line (5 in VA08AD)

$G(1), G(2), \dots, G(N)$ 8 to a line (5 in VA08AD)

The values of X and G can be suppressed on intermediate iterations by setting `IPRINT < 0`. All intermediate printing can be suppressed by setting `IPRINT=MAXFN+1`. All printing can be suppressed by setting `IPRINT=0`.

3 USER SUBROUTINE

The user must provide a subroutine headed

```
SUBROUTINE XXX(N,X,F,G)
REAL X(1),G(1)
```

where `XXX` is any identifier which must be passed as the first argument to `VA08`, see section 2, and appear in an `EXTERNAL` statement in the calling program. The subroutine should use the variables $X(1), X(2), \dots, X(N)$ to evaluate the function and gradient vector and place them in F and $G(1), G(2), \dots, G(N)$ respectively.

4 GENERAL

Use of common: None.

Workspace: N words provided by the user in `S`.

Input/output: controlled by the user through `IPRINT`. All printing is done on stream 6 (line printer).

Restrictions: none.

5 METHOD

The method of conjugate gradients (Fletcher, R. and Reeves, C.M. Function minimization by conjugate gradients, *Computer Journal* **7**, p.149) is used. The program differs from the ALGOL program given by Fletcher and Reeves in that a more efficient linear search is used requiring on average about two evaluations of $F(\mathbf{x})$ per iteration. See A.E.R.E. Report R-7073 (1972).

6 EXAMPLE OF USE

We consider the test problem

$$\text{minimize } 100(x_1^2 - x_2)^2 + (x_1 - 1)^2.$$

The solution to this problem is at the point (1,1) and the optimal function value is 0. We may solve this problem using the following code:

```
DOUBLE PRECISION F
DOUBLE PRECISION G(2),S(2),X(2)
EXTERNAL FUNCT,VA08AD
DATA X/-1.2D0,1.D0/
CALL VA08AD(FUNCT,2,X,F,G,S,-1.0D+0,1.0D-6,200,0)
WRITE( 6, 2000 ) F, X(1), X(2)
STOP
2000 FORMAT( ' Final function value = ', 1P, D12.4, /,
*          ' Solution components = ', 2D12.4 )
END
```

where the required external function evaluation subroutine is:

```
SUBROUTINE FUNCT(N,X,F,G)
DOUBLE PRECISION F
INTEGER N
DOUBLE PRECISION G(*),X(*)
DOUBLE PRECISION R,S
R = 10.D0* (X(1)**2-X(N))
S = X(1) - 1.D0
F = R**2 + S**2
G(1) = 40.D0*X(1)*R + 2.D0*S
G(N) = -20.D0*R
RETURN
END
```

The output is as follows:

```
Final function value = 2.0059D-12
Solution components = 1.0000D+00 1.0000D+00
```