



## 1 SUMMARY

To find the minimum of a function  $F(\mathbf{x})$  of several variables, given that the gradient vector  $(\partial F/\partial x_1, \partial F/\partial x_2, \dots, \partial F/\partial x_n)$  can be calculated. The subroutine replaces VA01 to which it is superior in various ways (see Section 5), and should be used whenever derivatives can be evaluated readily. It should however not be used either if storage space is at a premium (use VA08) or if the function is a sum of squares (use VA07). The subroutine complements VA06: the latter requires four times the storage, and some comparisons (R. Fletcher, AERE Report R-7125 (1972)) indicate that VA06 is marginally slower and more affected by round off error. As VA06 is more difficult to use, it is suggested that VA09 should be used in the first instance on any problem. If VA09 fails then VA06 should be tried as it is guaranteed to converge if the effect of rounding errors can be neglected.

**ATTRIBUTES** — **Version:** 1.0.0. **Types:** VA09A, VA09AD **Calls:** MC11, FD05. **Original date:** April 1972. **Origin:** R. Fletcher, Harwell.

## 2 HOW TO USE THE PACKAGE

### 2.1 The argument list and calling sequence

*The single precision version*

```
CALLVA09A(FUNCT, N, X, F, G, H, W, DFN, EPS, MODE, MAXFN, IRPRINT, IEXIT)
```

*The double precision version*

```
CALLVA90AD(FUNCT, N, X, F, G, H, W, DFN, EPS, MODE, MAXFN, IPRINT, IEXIT)
```

FUNCT is the name of the user's subroutine. See Section 3.

N is an INTEGER to be set to the number of variables ( $N \geq 2$ ).

X is a REAL (DOUBLE PRECISION in the D version) array of N elements in which the current estimate of the solution is stored. An initial approximation must be set in X on entry to VA09 and the best estimate obtained will be returned on exit.

F is 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 is a REAL (DOUBLE PRECISION in the D version) array of N elements in which the gradient vector corresponding to X above will be returned. This array need not to be set on entry.

H is a REAL (DOUBLE PRECISION in the D version) array of  $N(N+1)/2$  elements in which an estimate of the hessian matrix  $\partial^2 F/(\partial x_i \partial x_j)$  is stored. The matrix is represented in the product form  $\mathbf{LDL}^T$  where  $\mathbf{L}$  is a lower triangular matrix with unit diagonals and  $\mathbf{D}$  is a diagonal matrix. The lower triangle of  $\mathbf{L}$  is stored by columns in H excepting that the unit diagonal elements are replaced by the corresponding elements of D. The setting of H on entry is controlled by the parameter MODE (q.v.).

W is a REAL (DOUBLE PRECISION in the D version) array of  $3N$  elements used as working space.

DFN is a REAL (DOUBLE PRECISION in the D version) number which must be set so as to give VA09 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 on the sign of DFN which should be set in one of the following ways:

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

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

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

**EPS** is a REAL (DOUBLE PRECISION in the D version) array of  $N$  elements to be set on entry to the accuracy required in each element of  $X$ .

**MODE** is an INTEGER which controls the setting of the initial estimate of the hessian matrix in the parameter  $H$ . The following settings of **MODE** are permitted:

if  $MODE=1$ , an estimate corresponding to a unit matrix is set in  $H$  by VA09A.

if  $MODE=2$ , VA09A assumes that the hessian matrix itself has been set in  $H$  by columns of its lower triangle, and the conversion to  $LDL^T$  form is carried out by VA09A. The hessian matrix must be positive definite.

if  $MODE=3$ , VA09A assumes that the hessian matrix has been set in  $H$  in product form. This is convenient when using the  $H$  matrix from one problem as an initial estimate for another, in which case the contents of  $H$  are passed on unchanged.

**MAXFN** is an INTEGER set to the maximum number of calls of **FUNCT** permitted.

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

Function value  $X(1), X(2), \dots, X(N)$ ; 8 to a line (5 in VA09AD)

$G(1), G(2), \dots, G(N)$ ; 8 to a line (5 in VA09AD)

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  $IPRINT=0$ .

**IEXIT** is an INTEGER giving the reason for exit from VA09A. This will be set by VA09A as follows:

if  $IEXIT=0$  ( $MODE=2$  only), the estimate of the hessian matrix is not positive definite.

if  $IEXIT=1$ , a normal exit has been made in which  $|DX(I)| < EPS(I)$  for all  $I=1, 2, \dots, N$ , where  $DX(I)$  is the change in  $X$  on an iteration.

if  $IEXIT=2$ ,  $\mathbf{G}^T \mathbf{DX} \geq 0$ . This is not possible without rounding error. The probable cause is that  $EPS$  is set too small for the computer word length.

if  $IEXIT=3$ , **FUNCT** has been called **MAXFN** times.

### 3 USER SUBROUTINE

The user must provide a subroutine headed:

*The single precision version*

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

*The double precision version*

```
SUBROUTINE XXX(N,X,F,G)
DOUBLE PRECISION X(1),G(1)
```

where **XXX** is an identifier chosen by the user. This subroutine should use the variables  $x$  supplied in  $X(1), X(2), \dots, X(N)$  to evaluate the function and gradient vector and place them in  $F$  and  $G(1), G(2), \dots, G(N)$  respectively. **XXX** must be passed to VA09A as VA09A's first argument, see Section 2, and appear in an **EXTERNAL** statement in the program that calls VA09A.

### 4 GENERAL

**Use of common:** : None.

**Workspace:** :  $N(N+1)/2$  words +  $4N$  words provided by the user in H and W.

**Input/output:** : Controlled by the user through IPRINT. All output is on stream 6.

**Restrictions:** : None.

## 5 METHOD

The method used is a quasi-Newton method described by Fletcher (*Computer Journal*, **13**, p.317 (1970)), and is a modification of earlier methods of this type, such as that implemented by VA01A. The method is superior to that of VA01A on three counts:

1. It uses a formula to update the hessian approximation  $\mathbf{H}$  which has proved to be more efficient and reliable.
2. It uses a crude line search which has been shown to be more efficient than an accurate line search.
3. It represents  $\mathbf{H}$  by the product  $\mathbf{LDL}^T$ , which enables the positive definiteness of  $\mathbf{H}$  to be guaranteed, even in the presence of round-off error.