



1 SUMMARY

To find a minimax solution to a system of nonlinear algebraic equations

$$f_j(\underline{x}) \equiv f_j(x_1, \dots, x_n) = 0, \quad j = 1, 2, \dots, m,$$

where $n \geq 1$ and $m \geq 1$, in the case where the derivatives of the functions are supplied by the user. This means that the subroutine will find a local minimum of the function

$$F(\underline{x}) \equiv \max_j |f_j(\underline{x})| .$$

An important application of the subroutine is nonlinear minimax data-fitting, and it is also an efficient method for finding a zero of a set of non-linear equations.

ATTRIBUTES — **Version:** 1.0.0. **Types:** VG11A; VG11AD. **Calls:** calls MA25A/AD. **Original date:** February 1993. **Origin:** K. Madsen, Harwell, modified by M.J. Hopper, Rutherford Appleton Laboratory. **Remark:** This is a rewritten version of VG01A/AD.

2 HOW TO USE THE PACKAGE

2.1 The argument list and calling sequence

The single precision version

```
CALL VG11A(FDF, N, M, X, DX, DPS, MAXFUN, W, IW)
```

The double precision version

```
CALL VG11AD(FDF, N, M, X, DX, DPS, MAXFUN, W, IW)
```

FDF is the name of a subroutine written by the user. It must have the form

```
SUBROUTINE FDF(N, M, X, A, B)
  DIMENSION X(1), A(M,1), B(1)
```

It must calculate the values of the functions and their derivatives at the point $\underline{x} = (X(1), X(2), \dots, X(N))$, and store these in the following way

$$\begin{aligned} B(J) &= f_j(\underline{x}), & J &= 1, \dots, M \\ A(J,I) &= \partial f_j / \partial x_i(\underline{x}), & J &= 1, \dots, M; \quad I = 1, \dots, N \end{aligned}$$

NOTE: The name of this user subroutine, which can be any name of the user's choice, must appear in an EXTERNAL statement in the calling program. The arrays A, B and X must be declared with the precision appropriate to the version (VG11A or VG11AD) being used.

N is an INTEGER variable and must be set to n, the number of unknowns. Its value must be positive, and it is not changed by the subroutine.

M is an INTEGER variable and must be set to m, the number of functions. Its value must be positive, and it is not changed by the subroutine.

X is a REAL (DOUBLE PRECISION in the D version) array which must be set by the user to an approximation to the solution, $X(i) = x_i$, $i=1, \dots, n$. On exit X will contain the best solution found by the subroutine.

DX is a REAL (DOUBLE PRECISION in the D version) variable which controls the step length of the iterative method used. It must be set by the user to an initial value corresponding to the starting vector $\underline{x}^{(0)}$. DX should be chosen

so that in the region $\{\underline{x} \mid \|\underline{x} - \underline{x}^{(0)}\| < DX\}$ the functions f_j can be approximated reasonably good by linear functions. If the functions are nearly linear DX could be set to an approximate value of the distance between the starting vector and the solution, but if more curvature is present this value may be too large. However it is normally not severe to choose a bad initial value of DX , since DX is adjusted by the routine during the iteration. The value of DX must be positive.

EPS is a REAL (DOUBLE PRECISION in the D version) variable which must be set by the user to indicate the accuracy of the solution. The iteration is stopped when $\underline{x} = (X(1), \dots, X(N))$ is a nearby stationary point. More precisely when

$$\max_j |f_j(\underline{x})| - \min_{\|h\| \leq \|x\|} \{ \max_j |f_j(\underline{x}) + (\nabla f_j(\underline{x}), h)| \} < EPS|.$$

If EPS has been chosen too small the routine will stop when no better estimate to the solution can be obtained because of rounding errors. Therefore $EPS = 0$ will be an acceptable value. If the inequality is satisfied EPS will be unchanged by the routine, in other cases EPS will be set to 0.

MAXFUN is an INTEGER variable. It must be set by the user and its value gives an upper bound for the number of calls of FDF. If the number of iterations required exceeds **MAXFUN**, $W(M+1)$ to $(MAXFUN+1)$, and the routine will return. The value of **MAXFUN** must be positive.

W is a REAL (DOUBLE PRECISION in the D version) array which is used for working space. Its length must be at least the larger of the integers $(2mn+2m+2n)$ and $(mn+n^2+2m+13n+16)$. On exit it will contain the function values at the solution, i.e.

$$W(i) = f_i(X), \quad i=1, \dots, m$$

Further $W(M+1)$ will give the number of iterations used.

IW is an INTEGER array of length at least $m + 3n + 4$ which is used by the routine as workspace.

3 GENERAL INFORMATION

Use of common: None

Workspace: Provided by the user, see arguments **W** and **IW**

Other routines called directly: calls MA25A/AD

Input/output: None

Restrictions: $N \geq 1, M \geq 1, DX > 0$

4 METHOD

The method is described in Harwell Report T.P.559 by K. Madsen.