1 PURPOSE

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

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

where n ≥ 1 and m ≥ 1, in the case where the derivatives of the functions are not available. This means that the subroutine will find a local minimum of the function

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

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 nonlinear equations.

2 ARGUMENT LIST

CALL VG12A(FUNC,N,M,X,DX,DXMAX,STEPL,EPS,MAXFUN,W,IW)

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

```
SUBROUTINE FUNCT(N,M,X,B)
DIMENSION X(1), B(1)
```

It must calculate the function values at the point $\underline{x}$ = (X(1),X(2), ..., X(N)) and store these in the following way

$$\mathrm{B(J)} = f_J(\underline{x}), \qquad \mathrm{J} = 1,...,\mathrm{M}.$$

NOTE: The name of this user subroutine, which can be any name of the user’s own choice, must appear in an EXTERNAL statement in the calling program.

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 for VG12AD) array which must be set by the user to an approximation to the solution, X(i) = x_j, i = 1,...,n. On exit X will contain the best solution found by the subroutine.

DX is a REAL (DOUBLE PRECISION for VG12AD) 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 $\{\underline{x} \mid \|\underline{x}-\underline{x}^{(0)}\| < \mathrm{DX}\}$ the functions f_j can be approximated reasonably well 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.

DXMAX is a REAL (DOUBLE PRECISION for VG12AD) variable which must be set by the user to a generous upper bound of the step length. DXMAX may be set to an estimate of the distance between the starting vector and the solution vector. Its value must be positive.

STEPL is a REAL (DOUBLE PRECISION for VG12AD) variable which must be set by the user. It is used by the subroutine to calculate an initial approximation to the derivative matrix, which is found by the differences

$$\frac{\partial f_J}{\partial x_i} \approx \{f_j(x_1^{(0)}...,x_i^{(0)}+STEPL,...,x_n^{(0)})-f_j(\underline{x}^{(0)})\}/\mathrm{STEPL}$$

where $\underline{x}^{(0)}$ is the starting vector. A value near .001 $\|\underline{x}^{(0)}\|$ will often be adequate.

EPS is a REAL (DOUBLE PRECISION for VG12AD) variable which must be set by the user to indicate the accuracy of the solution. The iteration is stopped when $\underline{x}$ = (X(1),...,X(N)) is a nearby stationary point, more precisely when

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

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 FUNC. If the number of iterations required exceeds MAXFUN, EPS will be set to 0, W(M+1) to (MAXFUN+1), and the routine will return. The value of MAXFUN must be non-negative.

W is a REAL (DOUBLE PRECISION for VG12AD) array which is used for working space. Its length must be at least (mn + 2n^2 + 3m + 13n + 16). On exit it will contain the function values at the solution, i.e.

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

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

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

3 GENERAL INFORMATION

Use of COMMON:	none
Workspace:	provided by the user, see argument W and IW
Other subroutines:	calls MA25A/AD
Input/Output:	none
Restrictions:	N≥1, M≥1, DX>0, DMAX>0, STEPL>0
Date of routine:	June 1974.

4 METHOD

The method is similar to that described by K. Madsen (1973). The initial derivative matrix is found by a finite difference approximation, and then the derivative matrices are approximated by using the Broyden (1965) updating formula

$$B_{k+1} = B_k + \frac{[\gamma_k - B_k\, h_k]\, h_k^T}{\|h_k\|_2^2}$$

where

$$\gamma_k^T = (f_1(x_{k+1}) - f_1(x_k) - f_m(x_{k+1}) - f_m(x_k))$$

and

$$h_k = x_{k+1} - x_k$$

Linear independence of the directions h_k is maintained by the method given by Power (1970). The method will be documented in a forthcoming report.

REFERENCES

Broyden, C.G. (1965). A class of methods for solving nonlinear simultaneous equations. Math. Comp. (19), 577-593.

Madsen, K. (1973). An algorithm for minimax solution for over determined systems of non-linear equations. A.E.R.E. Report T.P. 559.

Powell, M.J.D. (1970). A Fortran subroutine for unconstrained minimization, requiring first derivatives of the objective function. A.E.R.E. Report R.6469.