

1 SUMMARY

This subroutine **finds a minimum of a smooth function of a single variable**. Given a function $f(x)$ it finds a value of x which gives a minimum value of $f(x)$.

Safeguarded quadratic interpolation is used with no attempt to avoid local minima which may not be the global minimum.

ATTRIBUTES — **Version:** 1.0.0. **Types:** VD04A, VD04AD **Calls:** A user supplied external function subprogram to calculate $f(x)$. **Original date:** July 1982 **Origin:** M.J.D.Powell. **Licence:** A third-party licence for this package is available without charge.

2 HOW TO USE THE PACKAGE

2.1 The argument list and calling sequence

The single precision version

```
CALL VD04A(FUNC, X, STEP, BOUND, XACC, NFMAX, INFO)
```

The double precision version

```
CALL VD04AD(FUNC, X, STEP, BOUND, XACC, NFMAX, INFO)
```

FUNC is a REAL (DOUBLE PRECISION in the D version) FUNCTION SUBPROGRAM which must be supplied by the user to calculate $f(x)$ given x . **FUNC** must be declared external in the calling program and should be of the form

```
FUNCTION FUNC(X)
```

where X is the independent variable. **FUNC** must not alter X .

X is a REAL (DOUBLE PRECISION in the D version) variable which must be set by the user to an initial guess of the minimum. On return from the subroutine X is set to the value of x which gives the least calculated value of $f(x)$.

STEP is a REAL (DOUBLE PRECISION in the D version) variable which must be set by the user to an order of magnitude estimate of the required change in x . This argument is not altered by the subroutine.

BOUND is a REAL (DOUBLE PRECISION in the D version) variable which must be set by the user to a limit on the amount by which x can be altered from its original value. This argument is not altered by the subroutine.

Restriction: $BOUND > 0$.

XACC is a REAL (DOUBLE PRECISION in the D version) variable which must be set by the user to the required absolute accuracy in the final value of x . On a successful return a minimum of $f(x)$ lies within distance **XACC** of the final value of x . **XACC** may be set to 0 in which case the subroutine will try to achieve the maximum possible accuracy in order to return with **INFO**=5. This argument is not altered by the subroutine. **Restriction:** $XACC \geq 0$.

NFMAX is an INTEGER variable which must be set by the user to the maximum number of calls of **FUNC** which he wishes to allow. This argument is not altered by the subroutine. **Restriction:** $NFMAX > 0$.

INFO is an INTEGER variable which is set by the subroutine to distinguish the following reasons for return:

INFO=1: A minimum has been found to the required accuracy.

INFO=2: The value of **BOUND** is so small that no change has been made to X .

INFO=3: The final value of X is at the bound, and probably the minimum is beyond the bound.

INFO=4: **NFMAX** calls of **FUNC** have been made.

INFO=5: Computer rounding errors prevent further refinement of X .

2.2 User supplied subprogram

The user must supply a REAL (DOUBLE PRECISION in the D version) FUNCTION SUBPROGRAM of the form FUNC(X) to evaluate $f(x)$ given x . This subprogram must be declared EXTERNAL in the calling program and must not alter X .

2.3 Checks on input parameters

Various checks on the suitability of the input parameters are carried out by the subroutine. If BOUND is too small to allow any change in X a return is made with INFO set to 4. If STEP is unsuitably chosen the subroutine computes a value for the initial change in X and continues.

3 GENERAL INFORMATION

Use of common: None.

Workspace: None.

Other routines called directly: A user supplied EXTERNAL function subprogram FUNC(X), (see §2.2).

Input/output: None.

Restrictions:

XACC \geq 0,

NFMAX $>$ 0,

BOUND $>$ 0.

4 METHOD

Safeguarded quadratic interpolation is used with no attempt to avoid local minima which may not be the global minimum. Because positions of minima of smooth functions are poorly defined when gradients are not supplied it is usual for the best accuracy in x to be about the square root of the machine precision.

5 EXAMPLE OF USE

Suppose that we are asked to find a minimum value of the function

$$f(x) = x^4 - x^2 - 1$$

The Fortran code for doing this might be as follows:

```
EXTERNAL FUNC
STEP=0.5
X=0.0
BOUND=50.0
NFMAX=50
XACC=1.E-4
CALL VD04A(FUNC,X,STEP,BOUND,XACC,
*          NFMAX,INFO)
WRITE (6,1) INFO,X
1 FORMAT(/5X,5HINFO=,I2,5X,2HX=,E15.7)
STOP
END
FUNCTION FUNC(X)
FUNC=(X*X-1.0)*X*X-1.0
RETURN
END
```

The above code will find a minimum at $x=\sqrt{2}/2$. However if we now wish to find a minimum for which x is negative we can do this by altering the starting value of x to -10.0 and setting `BOUND` to `10.0` to ensure that the minimum found has a negative x . The subroutine will now find a minimum at $x=-\sqrt{2}/2$.