Author(s) : J.J. Moré, M.Y. Cosnard
Submitter : K.S. Kölbig
Language : Fortran

Library: MATHLIB
Submitted: 01.06.1989
Revised: 01.12.1994

Numerical Solution of Systems of Nonlinear Equations

Subroutine subprograms RSNLEQ and DSNLEQ compute a vector x_i, $(i = 1, 2, \ldots, n)$, which approximates an exact solution x_i^* of the system of n nonlinear equations with n unknowns

$$F_i(x_1, \ldots, x_n) = 0, \quad (i = 1, 2, \ldots, n).$$

These subroutines incorporate two convergence test, making use of arguments FTOL and XTOL respectively. If x_i, $(i = 1, 2, \ldots, n)$, denotes the result of the current iteration, and x_i' the result of the previous iteration, the calculation is terminated if either of the following tests is successful:

$$\begin{aligned} &\text{Test 1 :} \quad \max |F_i(x_1, \ldots, x_n)| \le \mathtt{FTOL}, \\ &\text{Test 2 :} \quad \max |x_i - x_i'| \le \mathtt{XTOL} * \max |x_i|, \end{aligned}$$

where the maxima are taken over $1 \le i \le n$.

On computers other than CDC and Cray, only the double-precision version DSNLEQ is available. On CDC and Cray computers, only the single-precision version RSNLEQ is available.

Structure:

SUBROUTINE subprograms
User Entry Names : RSNLEQ, DSNLEQ
Obsolete User Entry Names : SNLEQ $\equiv$ RSNLEQ
Files Referenced : Unit 6
External References: User-supplied SUBROUTINE subprogram

Usage:

For t = R (type REAL), t = D (type DOUBLE PRECISION),

```
CALL tSNLEQ(N,X,F,FTOL,XTOL,MAXF,IPRT,INFO,SUB,W)
```

N (INTEGER) Number n of equations and variables.

X (type according to t) One-dimensional array of length $\ge$ N. On entry, X(i), (i = 1, ..., N), must contain an estimate to a solution x_i^* of the system of equations. On exit, X(i) contains the final estimate to x_i^*.

F (type according to t) One-dimensional array of length $\ge$ N. On exit, F(i), (i = 1, ..., N), contains the final value of the residual $F_\mathtt{i}$(X(1), ..., X(N)).

FTOL (type according to t) Accuracy parameter for Test 1.

XTOL (type according to t) Accuracy parameter for Test 2.

MAXF (INTEGER) Maximum permitted number of iterations, where each iteration involves N calls to the user-supplied subroutine SUB. The recommended value for MAXF is 50*(N+3).

IPRT (INTEGER) If IPRT = 0 no intermediate results are printed.
If IPRT = 1 the values of i and X(i), (i = 1, 2, ..., n), are printed after each iteration.

INFO (INTEGER) On exit, the value of INFO shows the reason why execution was terminated as follows:

0 Unacceptable input arguments (N < 1 or FTOL $\le$ 0 or XTOL $\le$ 0).

	1	Test 1 is successful.
	2	Test 2 is successful.
	3	Test 1 and Test 2 are both successful.
	4	Number of iterations is $\geq$ MAXF.
	5	Approximate (finite difference) Jacobian matrix is singular
	6	Iterations are not making good progress.
	7	Iterations are diverging.
	8	Iterations are converging, but either (i) XTOL is too small, or (ii) convergence is very slow because the Jacobian is nearly singular near x_i^* or because the variables x_i are badly scaled.
SUB	Name of a user-supplied SUBROUTINE subprogram, declared EXTERNAL in the calling program.	
W	(type according to t) Array containing at least N*(N+3) elements required as working-space.	

The user-supplied SUBROUTINE subprogram SUB should be of the form

```
SUBROUTINE SUB(N,X,F,K)
DIMENSION X(*),F(*)
...
```

Statements which set F(K) equal to the value of $F_{\mathrm{K}}(\mathrm{X}(1), ..., \mathrm{X}(\mathrm{N}))$ without changing any other element of array F.

```
...
RETURN
END
```

where X and F are of type t.
Subroutine SUB should not change the value of the argument K unless the user wants to terminate the execution of tSNLEQ, in which case K should be set equal to a negative integer, whose value will be copied into argument INFO of tSNLEQ before exit.

Method:

A modification of Brent's method as described in Ref. 1.

Error handling:

See description of argument INFO.

Notes:

1. Whenever possible the equations $F_i = 0$ should be numbered in order of increasing nonlinearity.
2. These subroutines do not use any techniques which attempt to obtain global convergence. Convergence may therefore fail to occur if the initial estimate is too far from an exact solution.

Source:

This subroutine has been adapted from the Fortran program published in Ref. 2.

References:

1. J.J. Moré and M.Y. Cosnard, Numerical solution of nonlinear equations, ACM Trans. Math. Software **5** (1979) 64–85.
2. J.J. Moré and M.Y. Cosnard, Algorithm 554 BRENTM, A FORTRAN subroutine for the numerical solution of systems of nonlinear equations, Collected Algorithms from CACM (1980).

•