NAG Fortran Library Routine Document

F08AFF (SORGQR/DORGQR)

Note: before using this routine, please read the Users' Note for your implementation to check the interpretation of ***bold italicised*** terms and other implementation-dependent details.

1 Purpose

F08AFF (SORGQR/DORGQR) generates all or part of the real orthogonal matrix Q from a QR factorization computed by F08AEF (SGEQRF/DGEQRF) or F08BEF (SGEQPF/DGEQPF).

2 Specification

```
SUBROUTINE F08AFF(M, N, K, A, LDA, TAU, WORK, LWORK, INFO)
ENTRY      sorgqr (M, N, K, A, LDA, TAU, WORK, LWORK, INFO)
INTEGER           M, N, K, LDA, LWORK, INFO
real              A(LDA,*), TAU(*), WORK(*)
```

The ENTRY statement enables the routine to be called by its LAPACK name.

3 Description

This routine is intended to be used after a call to F08AEF (SGEQRF/DGEQRF) or F08BEF (SGEQPF/DGEQPF), which perform a QR factorization of a real matrix A. The orthogonal matrix Q is represented as a product of elementary reflectors.

This routine may be used to generate Q explicitly as a square matrix, or to form only its leading columns.

Usually Q is determined from the QR factorization of an m by p matrix A with $m \geq p$. The whole of Q may be computed by:

```
CALL SORGQR (M,M,P,A,LDA,TAU,WORK,LWORK,INFO)
```

(note that the array A must have at least m columns) or its leading p columns by:

```
CALL SORGQR (M,P,P,A,LDA,TAU,WORK,LWORK,INFO)
```

The columns of Q returned by the last call form an orthonormal basis for the space spanned by the columns of A; thus F08AEF (SGEQRF/DGEQRF) followed by F08AFF (SORGQR/DORGQR) can be used to orthogonalise the columns of A.

The information returned by the QR factorization routines also yields the QR factorization of the leading k columns of A, where $k < p$. The orthogonal matrix arising from this factorization can be computed by:

```
CALL SORGQR (M,M,K,A,LDA,TAU,WORK,LWORK,INFO)
```

or its leading k columns by:

```
CALL SORGQR (M,K,K,A,LDA,TAU,WORK,LWORK,INFO)
```

4 References

Golub G H and van Loan C F (1996) *Matrix Computations* (3rd Edition) Johns Hopkins University Press, Baltimore

5 Parameters

1: M – INTEGER *Input*

On entry: m, the order of the orthogonal matrix Q.

Constraint: $\mathrm{M} \geq 0$.

2: N – INTEGER *Input*

On entry: n, the number of columns of matrix Q that are required.

Constraint: $\mathrm{M} \geq \mathrm{N} \geq 0$.

3: K – INTEGER *Input*

On entry: k, the number of elementary reflectors whose product defines the matrix Q.

Constraint: $\mathrm{N} \geq \mathrm{K} \geq 0$.

4: A(LDA,*) – ***real*** array *Input/Output*

Note: the second dimension of the array A must be at least $\max(1, \mathrm{N})$.

On entry: details of the vectors which define the elementary reflectors, as returned by F08AEF (SGEQRF/DGEQRF) or F08BEF (SGEQPF/DGEQPF).

On exit: the m by n matrix Q.

5: LDA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F08AFF (SORGQR/DORGQR) is called.

Constraint: $\mathrm{LDA} \geq \max(1, \mathrm{M})$.

6: TAU($*$) – ***real*** array *Input*

Note: the dimension of the array TAU must be at least $\max(1, \mathrm{K})$.

On entry: further details of the elementary reflectors, as returned by F08AEF (SGEQRF/DGEQRF) or F08BEF (SGEQPF/DGEQPF).

7: WORK($*$) – ***real*** array *Workspace*

Note: the dimension of the array WORK must be at least $\max(1, \mathrm{LWORK})$.

On exit: if INFO = 0, WORK(1) contains the minimum value of LWORK required for optimum performance.

8: LWORK – INTEGER *Input*

On entry: the dimension of the array WORK as declared in the (sub)program from which F08AFF (SORGQR/DORGQR) is called, unless $\mathrm{LWORK} = -1$, in which case a workspace query is assumed and the routine only calculates the optimal dimension of WORK (using the formula given below).

Suggested value: for optimum performance LWORK should be at least $\mathrm{N} \times nb$, where nb is the ***blocksize***.

Constraint: $\mathrm{LWORK} \geq \max(1, \mathrm{N})$ or $\mathrm{LWORK} = -1$.

9: INFO – INTEGER *Output*

On exit: INFO = 0 unless the routine detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the routine:

INFO < 0

If INFO $= -i$, the ith parameter had an illegal value. An explanatory message is output, and execution of the program is terminated.

7 Accuracy

The computed matrix Q differs from an exactly orthogonal matrix by a matrix E such that

$$\|E\|_2 = O(\epsilon),$$

where ϵ is the ***machine precision***.

8 Further Comments

The total number of floating-point operations is approximately $4mnk - 2(m+n)k^2 + \frac{4}{3}k^3$; when $n = k$, the number is approximately $\frac{2}{3}n^2(3m - n)$.

The complex analogue of this routine is F08ATF (CUNGQR/ZUNGQR).

9 Example

To form the leading 4 columns of the orthogonal matrix Q from the QR factorization of the matrix A, where

$$A = \begin{pmatrix} -0.57 & -1.28 & -0.39 & 0.25 \\ -1.93 & 1.08 & -0.31 & -2.14 \\ 2.30 & 0.24 & 0.40 & -0.35 \\ -1.93 & 0.64 & -0.66 & 0.08 \\ 0.15 & 0.30 & 0.15 & -2.13 \\ -0.02 & 1.03 & -1.43 & 0.50 \end{pmatrix}.$$

The columns of Q form an orthonormal basis for the space spanned by the columns of A.

9.1 Program Text

Note: the listing of the example program presented below uses ***bold italicised*** terms to denote precision-dependent details. Please read the Users' Note for your implementation to check the interpretation of these terms. As explained in the Essential Introduction to this manual, the results produced may not be identical for all implementations.

```
*     F08AFF Example Program Text
*     Mark 16 Release. NAG Copyright 1992.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          MMAX, NMAX, LDA, LWORK
      PARAMETER        (MMAX=8,NMAX=8,LDA=MMAX,LWORK=64*NMAX)
*     .. Local Scalars ..
      INTEGER          I, IFAIL, INFO, J, M, N
      CHARACTER*30     TITLE
*     .. Local Arrays ..
      real             A(LDA,NMAX), TAU(NMAX), WORK(LWORK)
*     .. External Subroutines ..
      EXTERNAL         sgeqrf, sorgqr, X04CAF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F08AFF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) M, N
      IF (M.LE.MMAX .AND. N.LE.NMAX .AND. M.GE.N) THEN
*
```

```
*        Read A from data file
*
         READ (NIN,*) ((A(I,J),J=1,N),I=1,M)
*
*        Compute the QR factorization of A
*
         CALL sgeqrf(M,N,A,LDA,TAU,WORK,LWORK,INFO)
*
*        Form the leading N columns of Q explicitly
*
         CALL sorgqr(M,N,N,A,LDA,TAU,WORK,LWORK,INFO)
*
*        Print the leading N columns of Q only
*
         WRITE (NOUT,*)
         WRITE (TITLE,99999) N
         IFAIL = 0
*
         CALL X04CAF('General',' ',M,N,A,LDA,TITLE,IFAIL)
*
      END IF
      STOP
*
99999 FORMAT ('The leading ',I2,' columns of Q')
      END
```

9.2 Program Data

```
F08AFF Example Program Data
  6  4                            :Values of M and N
 -0.57  -1.28  -0.39   0.25
 -1.93   1.08  -0.31  -2.14
  2.30   0.24   0.40  -0.35
 -1.93   0.64  -0.66   0.08
  0.15   0.30   0.15  -2.13
 -0.02   1.03  -1.43   0.50   :End of matrix A
```

9.3 Program Results

```
 F08AFF Example Program Results

 The leading  4 columns of Q
          1       2       3       4
 1  -0.1576  0.6744 -0.4571  0.4489
 2  -0.5335 -0.3861  0.2583  0.3898
 3   0.6358 -0.2928  0.0165  0.1930
 4  -0.5335 -0.1692 -0.0834 -0.2350
 5   0.0415 -0.1593  0.1475  0.7436
 6  -0.0055 -0.5064 -0.8339  0.0335
```