## NAME

cuncsd2by1.f −

## SYNOPSIS

### Functions/Subroutines

subroutine **cuncsd2by1** (JOBU1, JOBU2, JOBV1T, M, P, Q, X11, LDX11, X21, LDX21, THETA, U1, LDU1, U2, LDU2, V1T, LDV1T, WORK, LWORK, RWORK, LRWORK, IWORK, INFO)
**CUNCSD2BY1**

## Function/Subroutine Documentation

**subroutine cuncsd2by1 (characterJOBU1, characterJOBU2, characterJOBV1T, integerM, integerP, integerQ, complex, dimension(ldx11,*)X11, integerLDX11, complex, dimension(ldx21,*)X21, integerLDX21, real, dimension(*)THETA, complex, dimension(ldu1,*)U1, integerLDU1, complex, dimension(ldu2,*)U2, integerLDU2, complex, dimension(ldv1t,*)V1T, integerLDV1T, complex, dimension(*)WORK, integerLWORK, real, dimension(*)RWORK, integerLRWORK, integer, dimension(*)IWORK, integerINFO)**
**CUNCSD2BY1**

### Purpose:

CUNCSD2BY1 computes the CS decomposition of an M-by-Q matrix X with orthonormal columns that has been partitioned into a 2-by-1 block structure:

```
                  [ I  0  0 ]
                  [ 0  C  0 ]
    [ X11 ]   [ U1 |    ] [ 0  0  0 ]
X = [-----] = [---------] [----------] V1**T .
    [ X21 ]   [    | U2 ] [ 0  0  0 ]
                  [ 0  S  0 ]
                  [ 0  0  I ]
```

X11 is P-by-Q. The unitary matrices U1, U2, V1, and V2 are P-by-P, (M-P)-by-(M-P), Q-by-Q, and (M-Q)-by-(M-Q), respectively. C and S are R-by-R nonnegative diagonal matrices satisfying Cˆ2 + Sˆ2 = I, in which R = MIN(P,M-P,Q,M-Q)..fi

### Parameters:

*JOBU1*

JOBU1 is CHARACTER
= ’Y’: U1 is computed;
otherwise: U1 is not computed.

*JOBU2*

JOBU2 is CHARACTER
= ’Y’: U2 is computed;
otherwise: U2 is not computed.

*JOBV1T*

JOBV1T is CHARACTER
= ’Y’: V1T is computed;
otherwise: V1T is not computed.

*M*

M is INTEGER
The number of rows and columns in X.

*P*

P is INTEGER
The number of rows in X11 and X12. 0 <= P <= M.

*Q*

Q is INTEGER

The number of columns in X11 and X21. 0 <= Q <= M.

*X11*

X11 is COMPLEX array, dimension (LDX11,Q)
On entry, part of the unitary matrix whose CSD is desired.

*LDX11*

LDX11 is INTEGER
The leading dimension of X11. LDX11 >= MAX(1,P).

*X21*

X21 is COMPLEX array, dimension (LDX21,Q)
On entry, part of the unitary matrix whose CSD is desired.

*LDX21*

LDX21 is INTEGER
The leading dimension of X21. LDX21 >= MAX(1,M-P).

*THETA*

THETA is COMPLEX array, dimension (R), in which R = MIN(P,M-P,Q,M-Q).
C = DIAG( COS(THETA(1)), ... , COS(THETA(R)) ) and
S = DIAG( SIN(THETA(1)), ... , SIN(THETA(R)) ).

*U1*

U1 is COMPLEX array, dimension (P)
If JOBU1 = 'Y', U1 contains the P-by-P unitary matrix U1.

*LDU1*

LDU1 is INTEGER
The leading dimension of U1. If JOBU1 = 'Y', LDU1 >= MAX(1,P).

*U2*

U2 is COMPLEX array, dimension (M-P)
If JOBU2 = 'Y', U2 contains the (M-P)-by-(M-P) unitary matrix U2.

*LDU2*

LDU2 is INTEGER
The leading dimension of U2. If JOBU2 = 'Y', LDU2 >= MAX(1,M-P).

*V1T*

V1T is COMPLEX array, dimension (Q)
If JOBV1T = 'Y', V1T contains the Q-by-Q matrix unitary matrix V1**T.

*LDV1T*

LDV1T is INTEGER
The leading dimension of V1T. If JOBV1T = 'Y', LDV1T >= MAX(1,Q).

*WORK*

WORK is COMPLEX array, dimension (MAX(1,LWORK))
On exit, if INFO = 0, WORK(1) returns the optimal LWORK.
If INFO > 0 on exit, WORK(2:R) contains the values PHI(1), ..., PHI(R-1) that, together with THETA(1), ..., THETA(R), define the matrix in intermediate bidiagonal-block form

remaining after nonconvergence. INFO specifies the number
of nonzero PHI's.

*LWORK*

LWORK is INTEGER
The dimension of the array WORK.

If LWORK = -1, then a workspace query is assumed; the routine
only calculates the optimal size of the WORK array, returns
this value as the first entry of the work array, and no error
message related to LWORK is issued by XERBLA.

*RWORK*

RWORK is REAL array, dimension (MAX(1,LRWORK))
On exit, if INFO = 0, RWORK(1) returns the optimal LRWORK.
If INFO > 0 on exit, RWORK(2:R) contains the values PHI(1),
..., PHI(R-1) that, together with THETA(1), ..., THETA(R),
define the matrix in intermediate bidiagonal-block form
remaining after nonconvergence. INFO specifies the number
of nonzero PHI's.

*LRWORK*

LRWORK is INTEGER
The dimension of the array RWORK.

If LRWORK = -1, then a workspace query is assumed; the routine
only calculates the optimal size of the RWORK array, returns
this value as the first entry of the work array, and no error
message related to LRWORK is issued by XERBLA.

aram[out] IWORK
batim
IWORK is INTEGER array, dimension (M-MIN(P,M-P,Q,M-Q))

*INFO*

INFO is INTEGER
= 0: successful exit.
< 0: if INFO = -i, the i-th argument had an illegal value.
> 0: CBBCSD did not converge. See the description of WORK
above for details.

**References:**

[1] Brian D. Sutton. Computing the complete CS decomposition. Numer. Algorithms, 50(1):33-65, 2009.

**Author:**

Univ. of Tennessee

Univ. of California Berkeley

Univ. of Colorado Denver

NAG Ltd.

**Date:**

July 2012

Definition at line 260 of file cuncsd2by1.f.

**Author**

Generated automatically by Doxygen for LAPACK from the source code.