Title

gschurd() — Generalized Schur decomposition

Syntax

void gschurd(*A*, *B*, *T*, *R*, *U*, *V*, *w*, *b*)

void _gschurd(*A*, *B*, *U*, *V*, *w*, *b*)

void gschurdgroupby(*A*, *B*, *f*, *T*, *R*, *U*, *V*, *w*, *b*, *m*)

void _gschurdgroupby(*A*, *B*, *f*, *U*, *V*, *w*, *b*, *m*)

Description

gschurd(*A*, *B*, *T*, *R*, *U*, *V*, *w*, *b*) computes the generalized Schur decomposition of two square, numeric matrices, *A* and *B*, and the generalized eigenvalues. The decomposition is returned in the Schur-form matrix, *T*; the upper-triangular matrix, *R*; and the orthogonal (unitary) matrices, *U* and *V*. The generalized eigenvalues are returned in the complex vectors *w* and *b*.

gschurdgroupby(*A*, *B*, *f*, *T*, *R*, *U*, *V*, *w*, *b*, *m*) computes the generalized Schur decomposition of two square, numeric matrices, *A* and *B*, and the generalized eigenvalues, and groups the results according to whether a condition on each generalized eigenvalue is satisfied. *f* is a pointer to the function that implements the condition on each generalized eigenvalue, as discussed below. The number of generalized eigenvalues for which the condition is true is returned in *m*.

_gschurd() mirrors gschurd(), the difference being that it returns *T* in *A* and *R* in *B*.

_gschurdgroupby() mirrors gschurdgroupby(), the difference being that it returns *T* in *A* and *R* in *B*.

_gschurd_la() and _gschurdgroupby_la() are the interfaces into the LAPACK routines used to implement the above functions; see [M-1] **LAPACK**. Their direct use is not recommended.

Remarks and examples

Remarks are presented under the following headings:

Generalized Schur decomposition

The generalized Schur decomposition of a pair of square, numeric matrices, **A** and **B**, can be written as

$$\mathbf{U}' \times \mathbf{A} \times \mathbf{V} = \mathbf{T}$$

$$\mathbf{U}' \times \mathbf{B} \times \mathbf{V} = \mathbf{R}$$

where **T** is in Schur form, **R** is upper triangular, and **U** and **V** are orthogonal if **A** and **B** are real and are unitary if **A** or **B** is complex. The complex vectors **w** and **b** contain the generalized eigenvalues.

If **A** and **B** are real, **T** is in real Schur form and **R** is a real upper-triangular matrix. If **A** or **B** is complex, **T** is in complex Schur form and **R** is a complex upper-triangular matrix.

In the example below, we define A and B, obtain the generalized Schur decomposition, and list T and R.

```
:A = (6, 2, 8, -1\-3, -4, -6, 4\0, 8, 4, 1\-8, -7, -3, 5)
:B = (8, 0, -8, -1\-6, -2, -6, -1\-7, -6, 2, -6\1, -7, 9, 2)
:gschurd(A, B, T=., R=., U=., V=., w=., b=.)
: T
                  1              2              3              4
    +-------------------------------------------------------------+
  1 |   12.99313938    1.746927947    3.931212285   -10.91622337  |
  2 |             0     .014016016    6.153566902    1.908835695  |
  3 |             0   -4.362999645    1.849905717   -2.998194791  |
  4 |             0              0              0   -5.527285433  |
    +-------------------------------------------------------------+
: R
                  1              2              3              4
    +-------------------------------------------------------------+
  1 |   4.406836593    6.869534063   -1.840892081    1.740906311  |
  2 |             0    13.88730687              0   -.6995556735  |
  3 |             0              0    9.409495218   -4.659386723  |
  4 |             0              0              0    9.453808732  |
    +-------------------------------------------------------------+
: w
                 1                         2                         3              4
    +-----------------------------------------------------------------------------+
  1 |   12.9931394   .409611804+1.83488354i   .024799819-.111092453i   -5.52728543  |
    +-----------------------------------------------------------------------------+
: b
                  1              2              3              4
    +----------------------------------------------------------+
  1 |   4.406836593    4.145676341    .2509986829    9.453808732  |
    +----------------------------------------------------------+
```

Generalized eigenvalues can be obtained by typing

```
: w:/b
                 1                         2                         3              4
    +-----------------------------------------------------------------------------+
  1 |   2.94840508   .098804579+.442601735i   .098804579-.442601735i   -.584662287  |
    +-----------------------------------------------------------------------------+
```

Grouping the results

gschurdgroupby() reorders the generalized Schur decomposition so that a selected group of generalized eigenvalues appears in the leading block of the pair w and b. It also reorders the generalized Schur form T, R, and orthogonal (unitary) matrices, U and V, correspondingly.

We must pass gschurdgroupby() a pointer to a function that implements our criterion. The function must accept two arguments, a complex scalar and a real scalar, so that it can receive a generalized eigenvalue, and it must return the real value 0 to indicate rejection and a nonzero real value to indicate selection.

In the following example, we use gschurdgroupby() to put the finite, real, generalized eigenvalues first. One of the arguments to schurdgroupby() is a pointer to the function onlyreal() which accepts two arguments, a complex scalar and a real scalar that define a generalized eigenvalue. onlyreal() returns 1 if the generalized eigenvalue is finite and real; it returns zero otherwise.

```
: real scalar onlyreal(complex scalar w, real scalar b)
> {
>         if(b==0) return(0)
>         if(Im(w/b)==0) return(1)
>         return(0)
> }
: gschurdgroupby(A, B, &onlyreal(), T=., R=., U=., V=., w=., b=., m=.)
```

We obtain

```
: T
                 1             2             3             4
    +---------------------------------------------------------+
  1 |  12.99313938    8.19798168   6.285710813   5.563547054  |
  2 |            0  -5.952366071  -1.473533834   2.750066482  |
  3 |            0             0  -.2015830885   3.882051743  |
  4 |            0             0   6.337230739   1.752690714  |
    +---------------------------------------------------------+
: R
                 1             2             3             4
    +---------------------------------------------------------+
  1 |  4.406836593   2.267479575  -6.745927817   1.720793701  |
  2 |            0   10.18086202  -2.253089622    5.74882307  |
  3 |            0             0   -12.5704981             0  |
  4 |            0             0             0   9.652818299  |
    +---------------------------------------------------------+
: w
               1             2                       3                       4
    +---------------------------------------------------------------------------+
  1 |  12.9931394   -5.95236607   .36499234+1.63500766i   .36499234-1.63500766i  |
    +---------------------------------------------------------------------------+
: b
                 1             2             3             4
    +-----------------------------------------------------+
  1 |  4.406836593   10.18086202   3.694083258   3.694083258  |
    +-----------------------------------------------------+
: w:/b
               1             2                         3                         4
    +-------------------------------------------------------------------------------+
  1 |  2.94840508   -.584662287   .098804579+.442601735i   .098804579-.442601735i  |
    +-------------------------------------------------------------------------------+
```

m contains the number of real, generalized eigenvalues

```
: m
  2
```

Conformability

`gschurd(`A, B, T, R, U, V, w, b`):`

input:

A:	$n \times n$
B:	$n \times n$

output:

T:	$n \times n$
R:	$n \times n$
U:	$n \times n$
V:	$n \times n$
w:	$1 \times n$
b:	$1 \times n$

`_gschurd(`A, B, U, V, w, b`):`

input:

A:	$n \times n$
B:	$n \times n$

output:

A:	$n \times n$
B:	$n \times n$
U:	$n \times n$
V:	$n \times n$
w:	$1 \times n$
b:	$1 \times n$

`gschurdgroupby(`A, B, f, T, R, U, V, w, b, m`):`

input:

A:	$n \times n$
B:	$n \times n$
f:	1×1

output:

T:	$n \times n$
R:	$n \times n$
U:	$n \times n$
V:	$n \times n$
w:	$1 \times n$
b:	$1 \times n$
m:	1×1

`_gschurdgroupby(`A`, `B`, `f`, `U`, `V`, `w`, `b`, `m`):`

input:

A:	$n \times n$
B:	$n \times n$
f:	1×1

output:

A:	$n \times n$
B:	$n \times n$
U:	$n \times n$
V:	$n \times n$
w:	$1 \times n$
b:	$1 \times n$
m:	1×1

Diagnostics

`_gschurd()` and `_gschurdgroupby()` abort with error if A or B is a view.

`gschurd()`, `_gschurd()`, `gschurdgroupby()`, and `_gschurdgroupby()` return missing results if A or B contains missing values.

Also see

[M-1] **LAPACK** — The LAPACK linear-algebra routines

[M-5] **ghessenbergd( )** — Generalized Hessenberg decomposition

[M-5] **geigensystem( )** — Generalized eigenvectors and eigenvalues

[M-4] **matrix** — Matrix functions