NAG Toolbox for MATLAB

f01rk

1 Purpose

f01rk returns the first ℓ rows of the n by n unitary matrix P^{H}, where P is given as the product of Householder transformation matrices.

This function is intended for use following f01rj.

2 Syntax

```
[a, ifail] = f01rk(wheret, m, nrowp, a, theta, 'n', n)
```

3 Description

P is assumed to be given by

$$P = P_m P_{m-1} \cdots P_1,$$

where

$$P_k = I - \gamma_k u_k u_k^H,$$

$$u_k = \begin{pmatrix} w_k \\ \zeta_k \\ 0 \\ z_k \end{pmatrix}$$

γ_k is a scalar for which $\mathrm{Re}(\gamma_k) = 1.0$, ζ_k is a real scalar, w_k is a $(k-1)$ element vector and z_k is an $(n-m)$ element vector. w_k must be supplied in the kth row of **a** in elements $\mathbf{a}(k,1),\ldots,\mathbf{a}(k,k-1)$. z_k must be supplied in the kth row of **a** in elements $\mathbf{a}(k,m+1),\ldots,\mathbf{a}(k,n)$ and θ_k, given by

$$\theta_k = (\zeta_k, \mathrm{Im}(\gamma_k)),$$

must be supplied either in $\mathbf{a}(k,k)$ or in $\mathbf{theta}(k)$, depending upon the parameter **wheret**.

4 References

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

Wilkinson J H 1965 *The Algebraic Eigenvalue Problem* Oxford University Press, Oxford

5 Parameters

5.1 Compulsory Input Parameters

1: **wheret – string**

Indicates where the elements of θ are to be found.

wheret = 'I' (In **a**)

The elements of θ are in **a**.

wheret = 'S' (Separate)

The elements of θ are separate from **a**, in **theta**.

Constraint: **wheret** = 'I' or 'S'.

2: **m – int32 scalar**

m, the number of rows of the matrix A.

Constraint: $\mathbf{m} \geq 0$.

3: **nrowp – int32 scalar**

ℓ, the required number of rows of P.

If $\mathbf{nrowp} = 0$, an immediate return is effected.

Constraint: $0 \leq \mathbf{nrowp} \leq \mathbf{n}$.

4: **a(lda,∗) – complex array**

The first dimension of the array **a** must be at least $\max(1, \mathbf{m}, \mathbf{nrowp})$

The second dimension of the array must be at least $\max(1, \mathbf{n})$

The leading m by m strictly lower triangular part of the array **a**, and the m by $(n - m)$ rectangular part of **a** with top left-hand corner at element $\mathbf{a}(1, \mathbf{m} + 1)$ must contain details of the matrix P. In addition, if $\mathbf{wheret} = \text{'I'}$, the diagonal elements of **a** must contain the elements of θ.

5: **theta(∗) – complex array**

Note: the dimension of the array **theta** must be at least $\max(1, \mathbf{m})$ if $\mathbf{wheret} = \text{'S'}$, and at least 1 otherwise.

If $\mathbf{wheret} = \text{'S'}$, the array **theta** must contain the elements of θ. If $\mathbf{theta}(k) = 0.0$, P_k is assumed to be I, if $\mathbf{theta}(k) = \alpha$ and $\text{Re}(\alpha) < 0.0$, P_k is assumed to be of the form

$$P_k = \begin{pmatrix} I & 0 & 0 \\ 0 & \alpha & 0 \\ 0 & 0 & I \end{pmatrix},$$

otherwise $\mathbf{theta}(k)$ is assumed to contain θ_k given by

$$\theta_k = (\zeta_k, \text{Im}(\gamma_k)).$$

If $\mathbf{wheret} = \text{'I'}$ or 'i', the array **theta** is not referenced.

5.2 Optional Input Parameters

1: **n – int32 scalar**

Default: The second dimension of the array **a**.

n, the number of columns of the matrix A.

Constraint: $\mathbf{n} \geq \mathbf{m}$.

5.3 Input Parameters Omitted from the MATLAB Interface

lda, work

5.4 Output Parameters

1: **a(lda,∗) – complex array**

The first dimension of the array **a** must be at least $\max(1, \mathbf{m}, \mathbf{nrowp})$

The second dimension of the array must be at least $\max(1, \mathbf{n})$

The first **nrowp** rows of the array **a** are overwritten by the first **nrowp** rows of the n by n unitary matrix P^{H}.

2: **ifail – int32 scalar**

0 unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail $= -1$

On entry, **wheret** $\neq$ 'I' or 'S',
or **m** < 0,
or **n** $<$ **m**,
or **nrowp** < 0 or **nrowp** $>$ **n**,
or LD $< \max(1, \mathbf{m}, \mathbf{nrowp})$.

7 Accuracy

The computed matrix P satisfies the relation

$$P = Q + E,$$

where Q is an exactly unitary matrix and

$$\|E\| \leq c\epsilon,$$

where ϵ the ***machine precision*** (see x02aj), c is a modest function of n, and $\|.\|$ denotes the spectral (two) norm. See also Section 7 of the document for f01rj.

8 Further Comments

The approximate number of floating-point operations is given by

$\frac{8}{3}n[(3n-m)(2\ell-m)-m(\ell-m)]$, if $\ell \geq m$, and

$\frac{8}{3}\ell^2(3n-\ell)$, if $\ell < m$.

9 Example

```
wheret = 'Separate';
m = int32(3);
nrowp = int32(5);
a  =  [complex(0.7878385971583354,  +0),  complex(-0.2549249642552304,  -
0.4005963724010764), ...
          complex(-0.2773500981126146,  -0.2773500981126146),  complex(-
0.2849798528036647, ...
                    +0.5586154898871638),     complex(0.1153902061831629,
+0.7031378339767638);
           complex(0.0395714585753771,  +0.5221710199103989),  complex(-
2.112235418114766, ...
                +0),    complex(-1.109400392450458,   -0.5547001962252291),
complex(0.1283479119207073, ...
                +0.2317354439625548),    complex(0.07898771606177842,    -
0.03608286554775068);
                    complex(-0.2265135394293346,    +0.2265135394293346),
complex(0.04530270788586691, ...
     +0.3171189552010684), complex(-3.605551275463989, +0), complex(0, -
0), complex(0, +0.5436324946304029);
       complex(0, +0), complex(0, +0), complex(0, +0), complex(0, +0),
complex(0, +0);
       complex(0, +0), complex(0, +0), complex(0, +0), complex(0, +0),
complex(0, +0)];
theta = [complex(1.038730559968832, -0.100646644310659);
```

```
     complex(1.180971629471905, +0.3808634421302063);
     complex(1.224430551972102, -0)];
[aOut, ifail] = f01rk(wheret, m, nrowp, a, theta)
```

```
aOut =
  Columns 1 through 4
   -0.1970 - 0.1970i    0.0394 - 0.2757i    0.3151 + 0.1576i    0.1970 +
0.5909i
    0.1639 + 0.4916i   -0.2950 + 0.4261i    0.4516 + 0.3205i   -0.0473 +
0.3314i
   0.2774 + 0.2774i  -0.0555 + 0.3883i  -0.4992                 0
    0.3637 - 0.3213i   -0.4752 - 0.0982i   -0.2762 + 0.3049i    0.5121 +
0.0475i
    0.0123 - 0.5142i   -0.4187 + 0.2987i   -0.0339 - 0.3867i   -0.3613 +
0.3239i
  Column 5
  -0.1182 + 0.5646i
   0.0328 - 0.2076i
        0 + 0.6656i
  -0.2287 - 0.2072i
   0.2901 - 0.0254i
ifail =
           0
```