leigqNEWTON_init_vec — initial eigenvector guess

Construct a normalized initial right-vector guess `v0` for a candidate left eigenvalue.

`leigqNEWTON_init_vec` is a small helper used by the refinement routines. It builds an initial eigenvector guess for the (left) eigenvalue candidate `lambda`.

Supported Name-Value options (see `help leigqNEWTON_init_vec`): 'Side' : 'left' (default) or 'right' (internal; kept for completeness) 'Pivot' : [] (default, choose automatically) or an index 1..n 'Gauge' : true (default) apply a gauge fixing for numerical stability 'Normalize' : true (default) normalize the returned quaternion vector

Setup (requirements + path)

```
hasQuat = true;
try
    quaternion(0,0,0,0);
catch
    hasQuat = false;
end
if ~hasQuat
    disp('This toolbox requires MATLAB''s built-in quaternion class
(quaternion(w,x,y,z)).');
    disp('Examples in this page are skipped.');
    return;
end

% Ensure toolbox root is on the path (add only the root, no genpath).
if exist('leigqNEWTON_init_vec','file') ~= 2
    thisFile = mfilename('fullpath');
    if ~isempty(thisFile)
        rootGuess = fileparts(fileparts(fileparts(thisFile))); % .../docs/source ->
toolbox root
        if exist(fullfile(rootGuess,'leigqNEWTON_init_vec.m'),'file')
            addpath(rootGuess);
            rehash toolboxcache
        end
    end
end
if exist('leigqNEWTON_init_vec','file') ~= 2
    error('leigqNEWTON_init_vec not found on the MATLAB path. Add the toolbox root
folder.');
end
```

Example: initialize v0 for a computed eigenvalue candidate

A tiny 2×2 quaternion matrix (deliberately with non-real entries).

```
q0 = quaternion(0,0,0,0);
q1 = quaternion(1,0,0,0);
qi = quaternion(0,1,0,0);
```

```
qj = quaternion(0,0,1,0);

A  = [ q0, qi;
       qj, q1 ];

% Get a candidate eigenvalue (fast profile is fine here).
lambda = leigqNEWTON(A,'SolveProfile','fast','Seed',1);
lam1   = lambda(1);

% Default initialization
[v0,info0] = leigqNEWTON_init_vec(A, lam1);

% Pair-residual certificate (smaller is better).
r0 = leigqNEWTON_cert_resPair(A, lam1, v0);
disp('Default v0 residual certificate:');
```

```
Default v0 residual certificate:
```

```
disp(r0);
```

```
   2.6804e-11
```

Pivot / gauge options (diagnostic)

You can force a pivot entry and/or disable gauge fixing.

```
[v1,info1] = leigqNEWTON_init_vec(A, lam1, 'Pivot', 1, 'Gauge', false);
r1 = leigqNEWTON_cert_resPair(A, lam1, v1);

[v2,info2] = leigqNEWTON_init_vec(A, lam1, 'Pivot', 2, 'Gauge', true);
r2 = leigqNEWTON_cert_resPair(A, lam1, v2);

disp('Residuals for different initialization choices:');
```

```
Residuals for different initialization choices:
```

```
disp(table([r0;r1;r2], 'VariableNames', {'resPair'}, ...
    'RowNames', {'default','Pivot=1,Gauge=off','Pivot=2,Gauge=on'}));
```

	resPair
default	2.6804e-11
Pivot=1,Gauge=off	2.6804e-11
Pivot=2,Gauge=on	2.6804e-11

See also

leigqNEWTON_refine_polish, leigqNEWTON_refine_lambda, leigqNEWTON_cert_resPair