qmrdivideNEWTON — right matrix division B/A for quaternion matrices

Solve X*A = B for X using a real embedding. Convert to Live Script: File → Save As… → Live Script (*.mlx)

Setup

```
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

if exist('qmrdivideNEWTON','file') ~= 2
    thisFile = mfilename('fullpath');
    if ~isempty(thisFile)
        rootGuess = fileparts(fileparts(fileparts(thisFile))); % .../docs/source ->
toolbox root
        if exist(fullfile(rootGuess,'qmrdivideNEWTON.m'),'file')
            addpath(rootGuess);
            rehash toolboxcache
        end
    end
end
if exist('qmrdivideNEWTON','file') ~= 2
    error('qmrdivideNEWTON not found on the MATLAB path. Add the toolbox root
folder.');
end
```

Syntax

- X = qmrdivideNEWTON(B,A)

Example

```
rng(1);
n = 3;
A = quaternion(randn(n),randn(n),randn(n),randn(n));
% Make A better conditioned for the demo:
A = A + quaternion(5,0,0,0)*eye(n);

B = quaternion(randn(n),randn(n),randn(n),randn(n));
```

```
X = qmrdivideNEWTON(B,A);

disp('A:'); disp(A);
```

```
A:
        4.351 +  0.58644i +   1.1752j +   1.4022k      -1.1096 -   1.5094i +  0.60366j +   1.2708k      -0.55868 +  (
        1.1812 -  0.85189i +   2.0292j -   1.3677k       4.1544 +  0.87587i +   1.7813j + 0.066009k       0.17838 -
       -0.75845 +  0.80032i -  0.27516j -  0.29253k      -0.57266 -  0.24279i +   1.7737j +  0.45129k        4.8031 -
```

```
disp('B:'); disp(B);
```

```
B:
       -0.49079 -  0.34563i -   2.0187j +  0.28297k      -0.63579 +  0.92622i -     1.27j +  0.42286k      -0.15508 - 0.
        1.7972 -   1.1714i +  0.19974j + 0.063561k       0.60335 -   1.4817i -  0.48522j +   1.2995k       0.61212 -
        0.5907 -  0.68559i +  0.42586j +  0.43343k      -0.53525 -  0.55806i +  0.59431j -   1.0498k       -1.0443 +
```

```
disp('X = B/A (computed):'); disp(X);
```

```
X = B/A (computed):
       -0.30797 -  0.19433i -  0.17634j - 0.087305k      -0.42424 +  0.18695i -    0.3133j +  0.14351k      0.034737 -  (
        0.12961 -  0.28642i + 0.080214j +  0.20383k       0.15925 -  0.17614i -  0.49225j +  0.38786k        0.4229 -  (
        0.28911 -  0.10296i - 0.035079j + 0.067281k      -0.017417 -  0.17696i +  0.18939j -  0.16635k       -0.19188 - 0.
```

```
% Check residual: ||X*A - B|| (componentwise max-norm)
R = qmtimesNEWTON(X,A) - B;
[w,x,y,z] = parts(R);
maxAbsRes = max(abs([w(:);x(:);y(:);z(:)]));
fprintf('Residual max-abs component: %.3e\n', maxAbsRes);
```

```
Residual max-abs component: 6.661e-16
```

See also

qmldivideNEWTON, qmtimesNEWTON