Test of the linear systems from the convection-diffusion equations

This package provides implementations of comparing BICGSTAB, GMRES, DQGMRES, FOM, DIOM, SCG and SWI for solving the linear systems from the convection-diffusion equations:

$$Ax = b,$$

where $A \in \mathbb{R}^{n \times n}$ is unsymmetric positive definite.

Contents

BICGSTAB

Using Matlabs function "bicgstab" directly.

DQGMRES

DQGMRES (Direct Quasi-GMRES): Algorithms 6.6 and 6.13 in Yousef Saad's "Iterative Methods for Sparse Linear System (2nd Edition)"

```
function [x, k, res, resvec] = dqgmres(A, b, m, tol, x0, maxit)
```

• [x] = dqgmres(A, b) attempts to find a solution x to the system of linear equations $Ax = b$. The n-by-n coefficient matrix A must be positive definite but need not be symmetric. The right hand side column vector b must have length n.
• [x] = dqgmres(A, b, m) specifies the number of the sliding window. If m is [ ] then dqgmres uses the default, n.
• [x] = dqgmres(A, b, m, tol) specifies the tolerance of the method. If tol is [ ] then dqgmres uses the default, 1e-6.
• [x] = dqgmres(A, b, m, tol, x0) specifies the initial guess. If x0 is [ ] then dqgmres uses the default, an all zero vector.
• [x] = dqgmres(A, b, m, tol, x0, maxit) specifies the maximum number of iterations. If maxit is [ ] then dqgmres uses the default, 10000.
• [x, k] = dqgmres(A, b, ...) returns the iteration number at which x was computed: $1 \leq k \leq$ maxit.
• [x, k, res] = dqgmres(A, b, ...) also returns the last relative residual norm $\|b - Ax\|/\|b\|$.

• [x, k, res, resvec] = dqgmres(A, b, ...) also returns a vector of estimates of the residual norms at each iteration, including $\|b - Ax\|$.

GMRES: set $m = n$ in the function "dqgmres".

DIOM(m)

DIOM (Direct Incomplete Orthogonalization Method): Algorithms 6.6 and 6.8 in Yousef Saad's "Iterative Methods for Sparse Linear System (2nd Edition)"

```
function [x, k, res, resvec] = diom(A, b, mk, tol, x0, maxit)
```

• [x] = diom(A, b) attempts to find a solution x to the system of linear equations $Ax = b$. The n-by-n coefficient matrix A must be positive definite but need not be symmetric. The right hand side column vector b must have length n.
• [x] = diom(A, b, m) specifies the number of the sliding window. If m is [ ] then diom uses the default, n.
• [x] = diom(A, b, m, tol) specifies the tolerance of the method. If tol is [ ] then diom uses the default, 1e-6.
• [x] = diom(A, b, m, tol, x0) specifies the initial guess. If x0 is [ ] then diom uses the default, an all zero vector.
• [x] = diom(A, b, m, tol, x0, maxit) specifies the maximum number of iterations. If maxit is [ ] then diom uses the default, 10000.
• [x, k] = diom(A, b, ...) returns the iteration number at which x was computed: $1 \leq k \leq$ maxit.
• [x, k, res] = diom(A, b, ...) also returns the last relative residual norm $\|b - Ax\|/\|b\|$.
• [x, k, res, resvec] = diom(A, b, ...) also returns a vector of estimates of the residual norms at each iteration, including $\|b - Ax\|$.

FOM: set $m = n$ in the function "diom".

SWI(m)

SWI: Sliding window implementation with pre-allocated memory
SWIWP: Sliding window implementation without pre-allocated memory

```
[x, k, res, resvec] = swi(A, b, m, tol, x0, maxit)
[x, k, res, resvec] = swiwp(A, b, m, tol, x0, maxit)
```

• [x] = swi/swiwp(A, b) attempts to find a solution x to the system of linear equations $Ax = b$. The n-by-n coefficient matrix A must be positive definite but need not be symmetric. The right hand side column vector b must have length n.
• [x] = swi/swiwp(A, b, m) specifies the number of the sliding window. If m is [ ] then swi uses the default, n.

• [x] = swi/swiwp(A, b, m, tol) specifies the tolerance of the method. If tol is [ ] then swi uses the default, 1e-6.
• [x] = swi/swiwp(A, b, m, tol, x0) specifies the initial guess. If x0 is [ ] then swi uses the default, an all zero vector.
• [x] = swi/swiwp(A, b, m, tol, x0, maxit) specifies the maximum number of iterations. If maxit is [ ] then swi uses the default, 10000.
• [x, k] = swi/swiwp(A, b, ...) returns the iteration number at which x was computed: $1 \leq k \leq$ maxit
• [x, k, res] = swi/swiwp(A, b, ...) also returns the last relative residual norm $\|b - Ax\|/\|b\|$.
• [x, k, res, resvec] = swi/swiwp(A, b, ...) also returns a vector of estimates of the residual norms at each iteration, including $\|b - Ax\|$.

SCG: set $m = n$ in the function "swiwp".

Test

1. Download the files: diom.m, dqgmres.m, swi.m, swiwp.m, test_condiff.m, A.mat and b.mat;

2. Run the function "test_condiff" directly, i.e.,

```
>> test_condiff
```

```
BICGSTAB      129   0.0048   5.3573e-08
GMRES          43   0.0080   4.1714e-07
FOM            43   0.0066   4.4702e-07
DIOM(2)      5751   0.2269          NaN
DIOM(5)        49   0.0033   8.2513e-07
DIOM(10)       60   0.0050   2.3306e-07
SCG            43   0.0039   4.4702e-07
SWI(2)         71   0.0045   8.2779e-07
SWI(5)         52   0.0027   1.5787e-07
SWI(10)        63   0.0042   3.5019e-07
```