**MAPL 600 / CMSC 760 Fall 2007**
Take-Home Exam 2
Show all work.
All work must be your own (i.e., no group efforts are allowed).
If you use a reference, cite it, or you will lose credit!
**Work problems totaling 50 points.**
(I'll stop grading after that, so don't hand in extra parts.)
Due: Friday Oct 19, 8am. (See late penalty policy on information sheet.)

1a. (10) Let $\mathbf{T}_m$ be a symmetric tridiagonal matrix of size $m \times m$ and let $\mathbf{T}_{m-1}$ be the $(m-1) \times (m-1)$ matrix formed from its first $m-1$ rows and columns. Denote the eigenvalues of $\mathbf{T}_{m-1}$ by $\tau_1 \geq \ldots \geq \tau_{m-1}$, and denote the eigenvalues of $\mathbf{T}_m$ by $\lambda_1 \geq \ldots \geq \lambda_m$. Prove that for $j = 1, \ldots, m-1$,

$$\lambda_j \geq \tau_j \geq \lambda_{j+1}.$$

Note: The result is true for general symmetric matrices, but we only need it for the tridiagonal case. Hint: You may use without proof the Courant-Fischer Minimax Theorem which says that if $\mathbf{A}$ is symmetric then

$$\lambda_j(A) = \min\ \max\ \frac{\mathbf{y}^T \mathbf{A} \mathbf{y}}{\mathbf{y}^T \mathbf{y}}.$$

The min is taken over all choices of $j-1$ linearly independent vectors, and the max is taken over all nonzero vectors $\mathbf{y}$ that are orthogonal to these vectors. You may also use the Maximin Theorem

$$\lambda_j(A) = \max\ \min\ \frac{\mathbf{y}^T \mathbf{A} \mathbf{y}}{\mathbf{y}^T \mathbf{y}}.$$

The max is taken over all choices of $n-j$ linearly independent vectors, and the min is taken over all nonzero vectors $\mathbf{y}$ that are orthogonal to these vectors.

1b. (10) This leads to an algorithm for approximating some of the eigenvalues of a large sparse symmetric matrix $\mathbf{A}$: run Arnoldi and use the eigenvalues of $\mathbf{T}_m$ as approximations to some eigenvalues of $\mathbf{A}$. Prove (using the Arnoldi relation and about 2 lines of writing) that $\mathbf{T}_n$ is similar to the $n \times n$ matrix $\mathbf{A}$, and show how to use the eigenvectors of $\mathbf{T}_m$ to form approximations to some eigenvectors of $\mathbf{A}$.

1c. (10) Implement your algorithm to approximate some of the eigenvalues and eigenvectors of a symmetric positive definite $\mathbf{A}$, by modifying `cg.m` (available on the website) or by writing your own program. Use Matlab's `eig` to find the eigenvalues and eigenvectors of $\mathbf{T}_m$ (although faster algorithms exist, which take advantage of the tridiagonal structure of $\mathbf{T}_m$). Try it on the matrix `a = gallery('wathen',20,20)` and compare your computed eigenvalues at $m =$

100 iterations (using $\mathbf{b}$ = the vector of all ones) with the true values, computed by `eig`.

Discuss the results.

2a. (10) Suppose we apply the (symmetric) Lanczos (tridiagonalization) algorithm (p.186) to the matrix

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{C} \\ \mathbf{C}^T & \mathbf{0} \end{bmatrix}$$

where $\mathbf{C}$ is $m \times n$, $m \geq n$. Show that we obtain vectors $\mathbf{z}_1, \ldots, \mathbf{z}_k$ and $\mathbf{w}_1, \ldots, \mathbf{w}_k$ satisfying

$$\begin{aligned} \mathbf{C}\mathbf{Z}_k &= \mathbf{W}_{k+1}\bar{\mathbf{T}}_{k+1}, \\ \mathbf{C}^T\mathbf{W}_k &= \mathbf{Z}_{k+1}\bar{\mathbf{T}}_{k+1} \end{aligned}$$

where $\bar{\mathbf{T}}_{k+1}$ is $(k+1) \times k$ and tridiagonal. (Hint: Write out what $\mathbf{AV} = \mathbf{VT}$ means for this particular matrix.) Now show that the eigenvalues of $\mathbf{T}_{m+n}$ are equal to the singular values of $\mathbf{C}$, the negatives of the singular values of $\mathbf{C}$, and (possibly) zeros.

Hint: Every matrix has a singular value decomposition (SVD)

$$\mathbf{C} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^*$$

where

- $\mathbf{U}$ has dimension $m \times m$ and $\mathbf{U}^T\mathbf{U} = \mathbf{I}$,
- $\mathbf{\Sigma}$ has dimension $m \times n$, the only nonzeros are on the main diagonal, and these *singular values* are nonnegative real numbers $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n \geq 0$,
- $\mathbf{V}$ has dimension $n \times n$ and $\mathbf{V}^T\mathbf{V} = \mathbf{I}$.
- The columns of $\mathbf{U}$ are the eigenvectors of $\mathbf{C}\mathbf{C}^T$.
- The columns of $\mathbf{V}$ are the eigenvectors of $\mathbf{C}^T\mathbf{C}$.
- The eigenvalues of $\mathbf{C}^T\mathbf{C}$ (and the nonzero eigenvalues of $\mathbf{C}\mathbf{C}^T$) are $\sigma_1^2, \ldots, \sigma_n^2$.

2b. (10) Suppose we compute, column by column, the relations $\mathbf{CZ} = \mathbf{WB}$ and $\mathbf{C}^T\mathbf{W} = \mathbf{Z}\mathbf{B}^T$, where $\mathbf{W}^T\mathbf{W} = \mathbf{I}$, $\mathbf{Z}^T\mathbf{Z} = \mathbf{I}$, and $\mathbf{B}$ is zero except in its main diagonal and superdiagonal. (You need not show that this $\mathbf{W}$ and $\mathbf{Z}$ are the same as those from 2a.) Show that the singular values of $\mathbf{C}$ are equal to the singular values of $\mathbf{B}_n$. Write down the recurrences for an algorithm for approximating the singular values and singular vectors of $\mathbf{C}$ by computing the first $k$ columns of the relations and applying an SVD algorithm to $\mathbf{B}_k$.

2c. (10) Implement your algorithm from 2b in Matlab and try it on the matrix from `load('west0479.mat')`. Compare the singular values computed for $k = 100$ to the true values computed using `svd(full(west0479))`. Discuss the results.