CS 422: Assignment 4

Due: Friday, March 14, 2008 (P1,P2)) and Wednesday, March 26, 2008 (P3)

Scoring for each problem is on a 0-to-5 scale (5 = complete success, 4 = overlooked a small detail, 3 = good start, 2 = right idea, 1 = germ of the right idea, 0 = missed the point of the problem.) One point will be deducted for insufficiently commented code. Unless otherwise stated, you are expected to utilize fully MATLAB'S vectorizing capability subject to the constraint of being flop-efficient. Test drivers and related material are posted on the course website `http://www.cs.cornell.edu/courses/cs422/2008sp/`. For P1 and P2 submit output and a listing of all scripts/functions that *you* had to write in order to produce the output. You are allowed to discuss *background* issues with other students, but the codes you submit must be your own.

P1. (The Parallel Jacobi Ordering)

The cyclic Jacobi method for computing the Schur decomposition $Q^TAQ = \text{diag}(\lambda_1, \ldots, \lambda_n)$ of a symmetric $A \in \mathbb{R}^{n\times n}$ is structured as follows

$Q \leftarrow I_n$
while $\text{off}(A) > tol \cdot \| A \|_F$
  for $i = 1{:}n-1$
    for $j = i+1{:}n$
      Solve the (i,j) subproblem.
      Update: $A \leftarrow Q_{ij}^T A Q_{ij}$ and $Q \leftarrow QQ_{ij}$.
    end
  end
end

Here,

$$\text{off}(A) = \sqrt{\sum_{i\neq j} a_{ij}^2}\ ,$$

the (i,j) subproblem is the 2-by-2 Schur decomposition problem

$$\begin{bmatrix} c & s \\ -s & c \end{bmatrix}^T \begin{bmatrix} a_{ii} & a_{ij} \\ a_{ji} & a_{jj} \end{bmatrix} \begin{bmatrix} c & s \\ -s & c \end{bmatrix} = \begin{bmatrix} d_{ii} & 0 \\ 0 & d_{jj} \end{bmatrix},$$

and Q_{ij} is the identity except `Qij([i j],[i j]) = [c s ; -s c]`).

Notice that the i and j loops takes us through all $N = n(n-1)/2$ off-diagonal index pairs. Here is a different ordering illustrated for the case $n = 8$:

Rotation Set	Rotations			
1	(1,2)	(3,4)	(5,6)	(7,8)
2	(1,4)	(2,6)	(3,8)	(5,7)
3	(1,6)	(4,8)	(2,7)	(3,5)
4	(1,8)	(6,7)	(4,5)	(2,3)
5	(1,7)	(8,5)	(6,3)	(4,2)
6	(1,5)	(7,3)	(8,2)	(6,4)
7	(1,3)	(5,2)	(7,4)	(8,6)

```
pnew = [1 q(1) p(2:3)]
qnew = [q(2:4) p(4)]
p = pnew
q = qnew
```

This is referred to as the *parallel ordering* because the subproblems and updates associated with any rotation set are decoupled and can be carried out in parallel. To implement Jacobi with the parallel ordering, the i and j loops take the form

```
   for i=1:n-1
%     Perform the updates associated with rotation set i
      for j=1:n/2
%        Solve the jth subproblem of the rotation set and update A and Q
```

Write a function `[Q,D,sweeps] = ParJac(A,tol)` that implements this method. You may assume that n is even. The output matrix `D` should be the final A matrix, presumably within `tol` of being diagonal. The output parameter `sweeps` should be the number of times that the `while` body is executed. Refer to §4.5.8 in the text for details about solving the 2-by-2 subproblem. Test script `P1` on the website.

P2. (Block Jacobi)

In this problem you are to write a function `[Q,D,sweeps] = BlockJac(A,tol,m)` that implements a block version of the cyclic Jacobi procedure discussed in the previous problem. The incoming symmetric matrix is now an n-by-n matrix with m-by-m blocks, e.g.,

$$A = \begin{bmatrix} A_{11} & A_{12} & A_{13} & A_{14} \\ A_{21} & A_{22} & A_{23} & A_{24} \\ A_{31} & A_{32} & A_{33} & A_{34} \\ A_{41} & A_{42} & A_{43} & A_{44} \end{bmatrix}$$

The subproblems are now $2m$-by-$2m$ Schur decomposition problems:

$$Q_{ij}^T \begin{bmatrix} A_{ii} & A_{ij} \\ A_{ji} & A_{jj} \end{bmatrix} Q_{ij} = \begin{bmatrix} D_{ii} & 0 \\ 0 & D_{jj} \end{bmatrix}$$

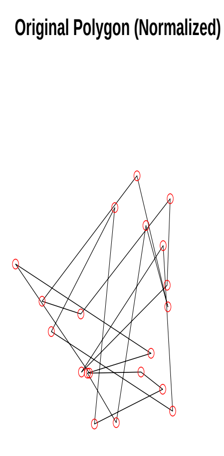
and the orthogonal update transformations have the form

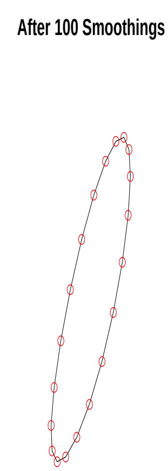
$$Q_{24} = \begin{bmatrix} I_m & 0 & 0 & 0 \\ 0 & X & 0 & X \\ 0 & 0 & I_m & 0 \\ 0 & X & 0 & X \end{bmatrix} \qquad \text{(The case } (i,j) = (2,4)\text{).}$$

You may use `schur` to solve the subproblems. Test script `P2` on the website.

P3. (Eigenanalysis–10pts)

Play with the function `ShowP3` available on the course website. This function starts with a random polygon P_0 whose centroid is at the origin. It then generates a sequence of polygons P_1, $P_2, \ldots$ where P_{k+1} is obtained by connecting P_k's edge midpoints. (The polygons are normalized in size for display purposes.) Apparently, the smoothing process "untangles" P_0 in that the P_k seem to converge to nice convex polygons with vertices on an ellipse.





Your job is to explain this phenomena by performing an eigenanalysis of the iteration. A 2+ page write up is expected. Your explanation can include any mix of theorems, experiments, and figures that you like. You will be graded on how convincing you are. "Proof by MATLAB" is OK as long as the experiments are well thought out and scientifically presented. Be concise: score is not proportional to length. Include listings of all scripts and functions that you developed in conjunction with your write-up. Treat this assignment as an opportunity to improve your technical writing. Hints and relevant study questions follow.

- The shift matrix $S_n = [I_n(:,n)\ \ I_n(:,1{:}n-1)]$ is critical, e.g.,

$$S_4 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

  The eigenvectors and eigenvalues of S_n are completely known. If

$$\omega_j = \cos\left(\frac{2\pi j}{n}\right) + i \sin\left(\frac{2\pi j}{n}\right)$$

  then for $j = 0{:}n-1$ we have

$$S_n \begin{bmatrix} 1 \\ \omega_j \\ \omega_j^2 \\ \vdots \\ \omega_j^{n-1} \end{bmatrix} = \omega_j \begin{bmatrix} 1 \\ \omega_j \\ \omega_j^2 \\ \vdots \\ \omega_j^{n-1} \end{bmatrix}$$

  This is confirmed by noting that $\omega_j^n = 1$. Note that 1 is an eigenvalue and the vector of all ones is its eigenvector. Also, ω_j and ω_{n-j} are complex conjugates.

- In the polygon iteration, x_{new} is a multiple of $(I + S_n)x_{current}$. Likewise, y_{new} is a multiple of $(I + S_n)y_{current}$. Thus, we have a pair of power methods going on that involve the matrix $M = I_n + S_n$.

- What is the dominant eigenvalue of M? Why don't the x's and y's converge to the dominant eigenvector? Hint: The polygons have their centroid at the origin.

- Explain why the x and y vectors are increasingly close to the span of these two vectors:

$$c = \begin{bmatrix} 1 \\ \cos(\theta) \\ \cos(2\theta) \\ \vdots \\ \cos((n-1)\theta) \end{bmatrix} \qquad s = \begin{bmatrix} 0 \\ \sin(\theta) \\ \sin(2\theta) \\ \vdots \\ \sin((n-1)\theta) \end{bmatrix} \qquad \theta = \frac{2\pi}{n}$$

  These are the real and imaginary parts of the eigenvectors associated with the eigenvalues ω_1 and ω_{n-1}. Describe the rate of convergence.

- What can you say about the matrix vector products Mc and Ms? Why are these products relevant to the analysis of the limiting polygons? Indeed, does the sequence of polygons $\{P_k\}$ converge to a particular polygon? Does a subsequence converge? Are the vertices of the limiting polygon on an ellipse? Can you compute the tilt angle of the ellipse from the initial distribution of the points?