CAAM 499 · Physics of Strings VIGRE Seminar
Computing Eigenvalues with Spectral Methods

This worksheet describes several computational exercises that will provide practice with the numerical computation of eigenvalues for our damped wave problems.

We begin with the simple second order eigenvalue problem

$$u''(x) = \lambda u(x) \qquad (1)$$

for $x \in [0, 1]$ with $u(0) = u(1) = 0$. This problem emerged in our seminar on 4 October when Sean solved the wave equation using separation of variables.

We seek values of λ ('eigenvalues') and nonzero functions $u(x)$ ('eigenfunctions' or 'modes') that satisfy equation (1).

You can verify that nontrivial solutions exist for eigenvalues

$$\lambda_n = -n^2\pi^2, \qquad \lambda = 0, 1, \ldots \qquad (2)$$

and corresponding eigenfunctions

$$u_n(x) = \sin(n\pi x), \qquad \lambda = 0, 1, \ldots. \qquad (3)$$

1. Download the code `cheb01.m` (a modification of Trefethen's `cheb.m` code from *Spectral Methods in MATLAB*) from the class website

   `http://www.caam.rice.edu/~embree/vigre/notes.html`

   Create a matrix approximation to the second derivative operator in the following way:

```
n = 16;               % N = discretization size
[D,x] = cheb01(n);    % D = first derivative matrix, x = grid points
D2 = D*D;             % make a second derivative matrix
D2 = D2(2:n,2:n);     % trim matrix to encode zero boundary conditions
```

   (See *Spectral Methods in MATLAB* for an explanation of this code.)

   Now use MATLAB's `eig` command to compute the eigenvalues of `D2`. Some of these will be accurate, others will be wildly wrong. We wish to sort out which ones we can trust.

(a) Which eigenvalues of D2 match the exact eigenvalues given in (2) to high accuracy?

(b) What happens as you increase the number of grid points, n?

(c) The command [U,L] = eig(D2) will compute a matrix V whose jth column (accessed through U(:,j)) is the eigenvector of D2 corresponding to the eigenvalue stored in L(j,j). This eigenvector should approximate the eigenfunction $u_n(x)$ if the corresponding eigenvalue approximates λ_n accurately.

Plot a few of the accurate eigenvectors, e.g., using the code plot(x, U(:,j), 'k.-') for several values of j.

(d) Compare the accuracy of the eigenvalues computed from the spectral method to those obtained with a second order finite difference discretization of the second derivative. Recall that this matrix can be generated in MATLAB via

```
n = 16;                % N = discretization size
h = 1/n;               % h = space between grid points
x = 0:h:1;             % x = uniform grid
D2 = (-2*eye(n-1)+diag(ones(n-2,1),1)+diag(ones(n-2,1),-1))/(h^2);
```

In particular, how does the accuracy of your computed eigenvalues computed from this matrix compare to your observations in part (b)?

Now we turn our attention to the wave equation

$$u_{tt}(x,t) = u_{xx}(x,t) - 2au(x,t), \tag{4}$$

for $x \in [0,1]$ with $u(0,t) = u(1,t) = 0$ and $u(x,0) = u_0(x)$. If we let $v = u_t$, then $v_t = u_{tt}$, and so we can write (4) as

$$\begin{bmatrix} u_t \\ v_t \end{bmatrix} = \begin{bmatrix} 0 & I \\ \mathrm{d}^2/\mathrm{d}x^2 & -2a(x) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}.$$

We shall write the matrix on the right hand side as

$$A = \begin{bmatrix} 0 & I \\ \mathrm{d}^2/\mathrm{d}x^2 & -2a(x) \end{bmatrix}. \tag{5}$$

The eigenvalues of this matrix will be those values of μ for which we can find some functions u and v ($[u\ v]^T \neq [0\ 0]^T$) such that $u(0) = u(1) = 0$ and

$$\begin{bmatrix} 0 & I \\ \mathrm{d}^2/\mathrm{d}x^2 & -2a(x) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \mu \begin{bmatrix} u \\ v \end{bmatrix}.$$

This is equivalent to the pair of equations

$$\begin{aligned} v &= \mu u \\ u'' &= \mu v + 2a(x)v, \end{aligned}$$

where we use primes to denote differentiation with respect to x (i.e., $u'' = u_{xx}$). Substituting the first of these equations into the second yields

$$u'' = \mu^2 u + 2\mu a(x)u.$$

If $a(x) = a$ is constant for all $x \in [0,1]$, then this equation becomes

$$u'' = (\mu^2 + 2a)u,$$

which is the problem (1) in disguise. If $a = 0$, then from (1) and (2) we find

$$\mu_{\pm n} = \pm\sqrt{\lambda_n} = \pm n\pi\mathrm{i}. \tag{6}$$

2. Download the code `makeAconst.m` from the class website. This code, printed below, computes a matrix approximation to the damped wave operator (5) when $a(x)$ is constant.

```
 function [A,x] = makeAconst(N,a);
% Spectral approximation of the wave operator with constant damping a(x) = c.
% N: indicates grid size (grid uses N-1 interior points)
% a: the constant damping parameter
% A: block matrix [0 I; d^2/dx^2 -2a]; dim(A) = 2*N (default N=32)
% x: the interior grid points

[D,x] = cheb01(N);
D2 = D^2;
D2 = D2(2:N,2:N); x = x(2:end-1);
ax = a*ones(size(x));
A = [zeros(N-1) eye(N-1); D2 -2*diag(ax)];
```

   (a) Compute the eigenvalues of `A` with no damping, $a(x) = 0$. How do they compare to the exact values (6)?

   (b) Generalize (6) for the case when $a(x) = a$ is a positive constant.

   (c) Verify your answer to (b) computationally using `makeAconst`.

   (d) Make a movie in MATLAB illustrating the evolution of the eigenvalues as a increases from zero to 2π. You can adapt the following code.

```
N = 32;
figure(1), clf
set(gcf,'doublebuffer','on')      % avoid flashing screen
a = linspace(0,2*pi,50);
for j=1:length(a)
    [A,x] = makeAconst(N,a);
    clf, plot(eig(A), 'k.')
    axis([-5 1 -50 50])           % use same axis for all plots
    pause(0.25)                   % wait .25 seconds btwn frames
end
```

3. Modify `makeAconst.m` to use the discontinuous damping function

$$a(x) = \begin{cases} 0 & x \in [0, 1/3) \\ c & x \in [1/3, 2/3] \\ 0 & x \in (2/3, 1]. \end{cases}$$

This corresponds to a string that is only damped in the middle third of its domain. It sparks an interesting theoretical question: if we only damp on part of the string, will all initial plucks decay at an exponential rate? Unlike our previous examples, we cannot easily analyze this damping function to get an exact answer. Instead, we will use our numerical approach to get some insight into the behavior of these eigenvalues. By previously computing the answers to problems whose answers we could check, you should have built up some faith in the accuracy of these numerical methods.

(a) Experiment with different values of $c \in [0, 2\pi]$. How do the eigenvalues qualitatively differ from the case of constant damping studied above?

(b) We do not have an exact formula for the true eigenvalues, but you can assess the accuracy of the approximate eigenvalues the extent they change as the number of grid points, `n`, increases. How does the accuracy compare to the results you obtained for constant a?

(c) Make a movie as you did for question 2(d) that illustrates the evolution of these eigenvalues as c grows.