MMAM (MTH 420/520) LAB6: Discrete Fourier series, FFT, and music

---

Students in MTH 420 follow DEMOS, follow (a). Turn in (b) and either (c) or (d)

(DEMO 1) *Remember the string in equilibrium described by* $-u_{xx} = f$*, and the model for the vibrating string* $u_{tt} - u_{xx} = 0$ *? Natural frequencies of that string produce sounds which humans can hear. Recall that these vibrations are linear combinations of the functions* $\sin(nx)$ *with amplitude depending on time. Now each of the eigenfunctions* $\sin(nx)$ *of* $-\frac{d^2}{dx^2}$ *is periodic on* $(-\frac{\pi}{n}, \frac{\pi}{n})$ *with period* $T = \frac{\pi}{n}$ *and the frequency* $f = \frac{1}{T}$ *measured in Hz=1/s.*

Now, let us discover the connection between the sines and the sounds. The frequency 220 Hz is an "A" in music, 330 is "E" etc. Doubling the frequency produces a sound at an octave higher.

```
>> %% Plot and play short 'A' note
>> t=linspace(0,2,2000);
>> plot(t,sin(220*2*pi*t));
>> sound(sin(220*2*pi*t));
>> %%% Note: The default sampling rate is $Fs$=8192 Hz (more about this later).
>> %%% Change the default sampling rate and go to next octave
>> sound(sin(2*220*2*pi*t),10000);
>> %%% ... and to the next octave ???
>> sound(sin(?*220*2*pi*t),10000);
```

(DEMO 2) on sound/music in MATLAB (more on FFT later). Or skip to projects.

```
>> %% Now play the 'C' major scale. Does everything sound all right ? If not, correct
>> t=linspace(0,2,10000);
>> for f = [262 294 330 349 392 440 466 524] s=sin(f*2*pi*t);plot(t,s); sound(s); pause; end
```

More ?

```
>> load train
>> sound(y,Fs)
>> plot(y)
>> size(y)
>> plot(abs(fft(y)))
```

How many frequencies (with what amplitudes) do you "see" in the "train" ? You can also

```
>> load chirp; sound(y,Fs)
>> load handle.mat; sound(y,Fs)
%% what about doubling sampling rate ?
>> sound(y,2*Fs);
>> load gong.mat
```

More information at `http://www.mathworks.com/moler/fourier.pdf`*.*

(DEMO 3) To understand more about frequencies, harmonics, and signals, follow:

```
>> Fs = 64;          %% Frequency of sampling
>> T = 1/Fs;         %% Time
>> L = Fs;           %% Length of signal
>> t = (0:L-1)*T;    %% Time vector for plotting
>> %%%%% Construct a pure wave with frequency 50 Hz and 21 Hz
>> x = 0.7*sin(2*pi*50*t) + sin(2*pi*21*t);
>> %%%%% Now plot these superposed waves
>> plot(Fs*t,x)
>> NFFT = 2^nextpow2(L); %% in fft it is important to use powers of 2  (here NFFT = 64 =L)
>> Y = fft(x,NFFT)/L;   %% perform the transform to extract the frequencies
```

```
>> plot(2*abs(Y(1:NFFT/2+1)))  %% Do you see the 0.7 and 1 amplitudes ?
>> bar(2*abs(Y(1:NFFT/2+1)))   %% yet another way
```

Next you can perturb the signal with some noise

```
>> x = 0.7*sin(2*pi*50*t) + sin(2*pi*21*t);
>> %%% repeat steps above starting with plot(Fs*t,x)
```

PROJECT on DFT *Now we explore the use of Discrete Fourier Transform (DFT). Recall the class example*

(1) $$f(t) = \sin(t) + \cos(t),$$

written as a complex Fourier series

(2) $$f(t) = c_0 e^{-it} + c_1 + c_2 e^{it}.$$

Your answer should have been $c_0 = \frac{1}{2} + \frac{i}{2}, c_1 = 0, c_2 = \frac{1}{2} - \frac{i}{2}$. [Here and in what follows we follow notation from the textbook in Section 4.2.] So the function f has been assigned THREE pieces of information, i.e., the coefficients c_0, c_1, c_2 which uniquely identify it. Could we do this more efficiently than by matching at *every* t ?

Try this. Instead of matching $f(t)$ at every $t \in (0, T)$, we will only use a certain number of points $t_0, t_1, \ldots$ at which the function is sampled. This is called DFT, the Discrete Fourier Transform.

DFT is realized in MATLAB via the famous FFT (Fast Fourier Transform) but we do not introduce details of this powerful algorithm. FFT was originally developed to solve PDEs with constant coefficients, and now is widely used in numerical harmonic analysis and inverse problems.

We proceed with DFT for (2) . Since we seek three coefficients, we will look for a match at 3 points t_0, t_1, t_2. (it In general, for other frequencies, when matching with the series $c_0 e^{-ikt} + c_1 e^{-i(k-1)t} + \ldots c_{k-1} e^{-it} + c_k + c_{k+1} e^{it} + \ldots c_{2k+1} e^{ikt}$, we will need $n = 2k+1 = 3$ coefficients, thus n sampling points. For (2) we simply had $k = 1$ and $n = 3$).

Which points $t_0, t_1, \ldots$ should we use ? Since $f(t)$ is periodic with period $T = 2\pi$, it makes no sense to look outside $[0, 2\pi)$! The sampling points will be chosen to be equidistant in $[0, 2\pi]$. We choose for them to be $\phi = T/n = T/3$ apart. With the notation from the book, we define $w = e^{i\phi} = e^{iT/n}$, and $t_j = j\phi, \quad j = 0, 1, 2, \ldots n-1$. Now we sample (2) at each t_j and we obtain n equations

$$f(t_j) = c_0 e^{-it_j} + c_1 + c_2 e^{it_j}.$$

To simplify, we multiply each such equation corresponding to t_j by the highest term $e^{it_j} = w^j$. Now we summarize and write the system of equations

(3) $$c_0 + c_1 e^{it_j} + c_2 e^{2it_j} + \ldots = f(t_j) e^{it_j}, j = 0, 1, \ldots$$

(a) Write (3) as a linear system to be solved for $c = [c_0, c_1, c_2]^T$ in the form $Fc = g$. Identify the matrix F and compare with that in textbook on p.293. Also define

(4) $$g_j := f(t_j) e^{it_j}.$$

Solve for c_0, c_1, c_2 by hand or in MATLAB and check with what you got by hand in class.

```
>> f = @(t)(sin(t)+cos(t));
>> n = 3;tn=(0:n-1)*(2*(pi)/n);tn = tn';
>> fn = f(tn);
>> gn = exp(i*tn).*fn
>> %% Your calculation of F goes here ...
>> F =
```

```
>> %% Now get your coefficients c
>> c = F \ gn
>> %% Did you get the values you wanted ? Compare with LAB5.
```

(b) Now we use `fft`. It realizes DFT that is, it computes c, given f. (The process is much simpler than your work in (a)).

```
>> %% continue with the previous variables ...
>> fn = ...
>> gn = ...
>> cfft = ifft(gn)
>> %%% Compare with vector c you obtained in (a). Same or shifted? (shift cannot be helped)
>> ...
>> %% Plot the amplitudes. What do you see ?
>> plot(abs(cfft))
```

(c) Now, new examples. In (1) and (2) both components of f had the same amplitudes and frequencies, and you only needed $k = 1$. Repeat the steps (a-b) for

$$f(t) = 10\cos(t) - 1 + 1/3\sin(2t). \tag{5}$$

(You must find the period T first. Next choose k and an appropriate number of sampling points $n = 2k + 1$ depending on k that you selected. Write the system of equations, find the matrix F and solve for c using the linear system and fft.)

(d) Instead (c), consider

$$f(t) = \sin(t) + \sin(3t/2) + \sin(5t/4). \tag{6}$$

(These are frequencies corresponding to the major triad in C major scale. You can hear this function!) Repeat (a)-(d) for the function (6).

(Extra credit) Make (6) sound. Describe how you did it. Send me your code so I can enjoy it too.

(Extra Extra Extra credit, or do this instead of (c) or (d): Code your favorite SHORT song such as "Lightly row". Make it sound and send me your code.)