EE469B: Assignment 1

Due Wednesday, Sept. 30

Assignments This quarter the assignments will be partly matlab, and partly calculations you will need to work out by hand. If you want, you can typeset your solutions (LaTeX, word, or whatever else you'd like). Convert the output to a pdf file, and email the pdf to me. Or, if you would prefer, turn in handwritten solutions, along with printouts from matlab. Turn these in at class, or leave them in the box outside my door (258 Packard). Either way, you solution is due sometime the night of the due date.

For those of you who would like to use (or learn) LaTeX, I will put the latex source code on the web site, so you can splice your answers directly in. This is a nice way to keep the problems and the solutions together.

One approach for your matlab plots is to generate the figure, and then use the command line

```
>> print -dpdf myplot.pdf
```

This makes sure you don't have a rasterized image. This will be centered on a full page, so use a pdf viewer to trim it to fit. On a mac, Preview does this. First select the plot with the "select" tool, the use the "crop" command, found under the "Tools" menu. You can add annotation in matlab if you'd like, or another program.

The default matlab plot linewidth is 0.5 pt, which almost disappears. You can change this in the plot window by choosing "Axes Properties" under the "Edit " menu. Click on the line you want to change, and you'll get a set of controls to change the width, color, and other properties. A width of two points shows up well.

For the matlab problems, if you are asked to write an mfile, include the code in your solution. This will probably be very short. Include the requested plots, along with the listing of how the plot was generated. Use subplots to save paper. This could be part of a diary file, or simply the command that was invoked to generate the plot.

Make sure that your matlab path doesn't contain the directory `rf_tools`, or you will get unexpected results, and lots of frustration. RF tools uses left handed rotations (as does most of MR), while we will use right handed rotations. Use the matlab `path` command to check.

Introduction This assignment concerns typical Fourier transform designs of excitation pulses. This includes designing windowed sinc pulses, calculating the RF amplitude required, simulating slice profiles, designing a pulse for a specific application, and computing the relative SAR of a pulse sequence. For simulating the slice profile, you will need two m-files that are available on the web site at

`http://ee469b.stanford.edu/mfiles`

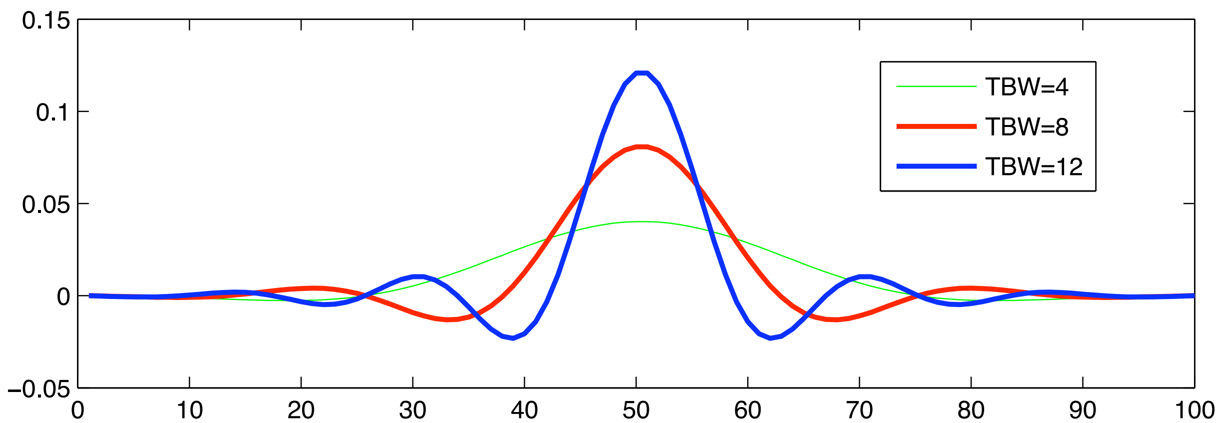
After the homework is due, sample mfiles for the homework problems will also be available here. Make sure your versions work, or download the sample solutions. You will be depending on these for the rest of the class.

1. Design of Windowed Sinc RF Pulses Write an m-file that computes a Hamming windowed sinc pulse, given a time-bandwidth product, and number of samples.

```
>> rf = wsinc(timebandwidth, samples)
```

Write the mfile so that it scales the waveform to sum to one, `sum(rf) = 1`. Plot windowed sincs with TBW of 4, 8, and 12. The TBW=4 pulse is common for 180 degree pulses, the TBW of 8 is typical for excitation pulses, and a TBW of 12 or 16 is typical for slab select pulses.

Solution:



2. Plot RF Amplitude For convenience, we will assume that the RF waveforms are normalized so that the sum of the RF waveform is the flip angle in radians. The sampled RF waveform can then be thought of as a sequence of small flips. This eliminates the need to explicitly consider the pulse duration in the design and simulation. However, it is sometimes important to compute the RF pulse amplitude in Gauss. In this problem you will write an m-file that takes a normalized RF pulse, and then, given a overall pulse length, scales the waveform to Gauss.

First, generate a 3.2 ms, $TBW = 8$ windowed sinc RF pulse, and scale it to a $\pi/2$ flip angle

```
>> rf = (pi/2)* wsinc(8,256);
```

Then, write an m-file called `rfscaleg` that takes a normalized RF pulse and a pulse duration, and returns a waveform that is scaled to Gauss,

```
>> rfs = rfscaleg(rf, pulseduration);
```

Plot the pulse you generated, scaled to Gauss. Label the axes. What is the peak amplitude?

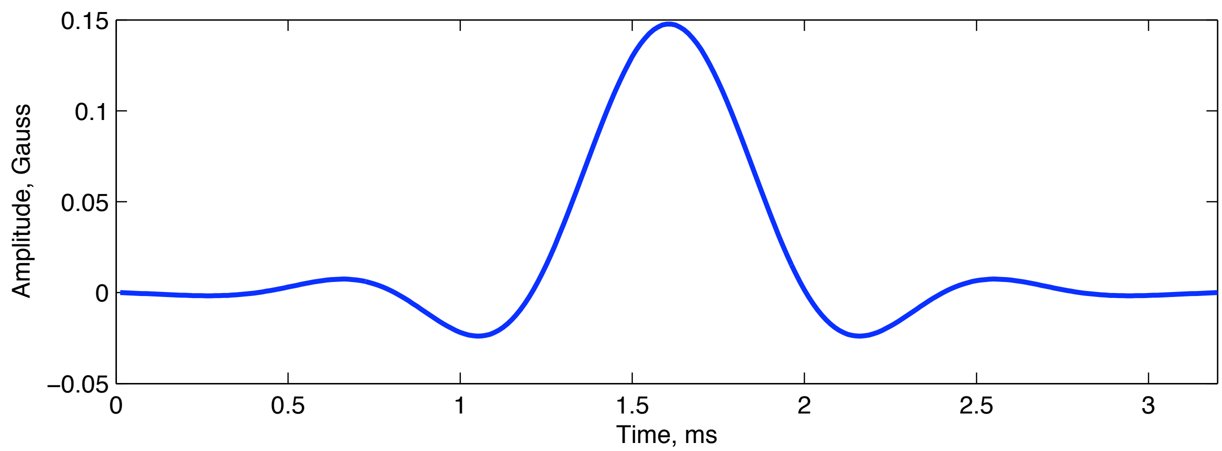
Solution: One possible m-file for `rfscaleg.m` is

```
function rfs = rfscaleg(rf,t);

%  rfs = rfscaleg(rf,t)
%
%    rf   -- rf waveform, scaled so sum(rf) = flip angle
%    t    -- duration of RF pulse in ms
%    rfs -- rf waveform scaled to Gauss
```

```
%

gamma = 2*pi*4.257; % kHz/G
dt = t/length(rf);
rfs = rf/(gamma*dt);
```

This is available on the web site. This takes the rotation produced by each sample, and determines what RF amplitude would be needed to produce that rotation in one sample dwell time. Note that the input RF is in radians, so we need the 2π factor in gamma.

The scaled RF plot looks like



The peak amplitude is a about 0.147 G.

3. Simulated Slice Profiles An RF pulse simulation routine is provided in the m-file `abrm.m` on the web site. This works by multiplying the sequence of rotations produced by the sampled RF waveform, using a spinor representation of the rotations. We'll describe how this works in about two weeks. The transverse magnetization it produces if given by

```
>> mxy = ab2ex(abrm(rf,x))
```

where `ab2ex.m` is another m-file on the web page that converts the spinor solution to the transverse magnetization for an excitation pulse. The vector `x` contains the spatial positions for the samples of the profile that you would like to compute in "resolution units" (this would be one voxel if we were imaging). Note that `mxy` is complex, and is `mx+i*my`.

The width of a resolution unit Δx is

$$\Delta x = \frac{1}{2k_{max}}$$

where $\pm k_{max}$ is the extent of the pulse in excitation k-space, and the area under the slice select lobe is $2k_{max}$. Write an m-file that takes a gradient amplitude in G/cm and a pulse duration in ms, and converts the dimensionless `x` vector to true spatial position measured in cm,

```
>> xt = gt2cm(x,g,t);
```

where `g` is the gradient strength in G/cm, and `t` is the time in ms.

Calculate the slice thickness of the $TBW = 8$ pulse from problem 2, based on the relations presented in class. Assume the slice select gradient is 0.6 G/cm. Simulate the RF pulse using `ab2ex(abrm(rf,x))`, and plot the magnitude of the result

```
>> plot(gt2cm(x,0.6,3.2), abs(ab2ex(abrm(rf,x))));
```

Choose `x` to show the interesting part of the plot, and label the axes. Is the slice the expected width?

Solution: An m-file for `gt2cm.m` is

```
%
%  Takes dimensionless x used by abr, and scales it to cm based
%  on a gradient strength g (G/cm) and pulse duration t (ms)
%
%   xs = gt2cm(x,g,t)
%
%     x  -- normalized frequency x vector used by abr
%     g  -- Gradient strength, G/cm
%     t  -- pulse duration in ms
%
%     xs -- scaled spatial axis, in cm
%

%  written by John Pauly, 1992
%  (c) Board of Trustees, Leland Stanford Junior University

function xs = gt2cm(x,g,t)

xs = x/(4.257*g*t);
```

The vector `x` is given in cycles over the pulse duration. The factor $\frac{\gamma}{2\pi}Gt$ is the gradient area, in cycles/cm. So if we divide `x` by this factor, we get `x` in cm.

The slice width can be found from

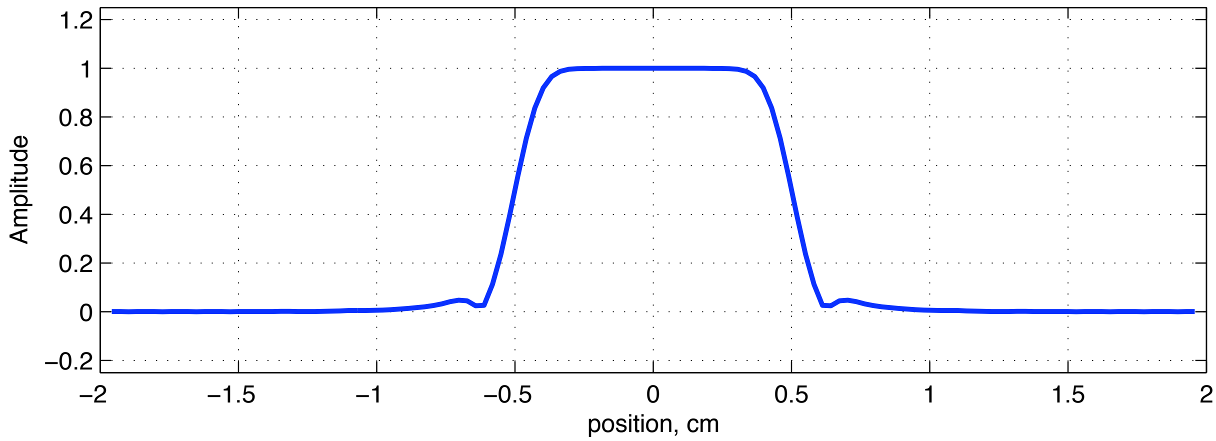
$$T(BW) = 8$$

so

$$BW = 8/(3.2 \text{ ms}) = 2.5 \text{ kHz}$$

The gradient strength is 0.6 G/cm or $\frac{\gamma}{2\pi}0.6$ G/cm $= 2.55$ kHz/cm. The slice thickness is then about

$$\frac{2.5 \text{ kHz}}{2.55 \text{ kHz/cm}} = 0.98 \text{ cm}$$



This shows the half-amplitude width to be about 1 cm, very close to what we calculated.

4. Design a Slab Select Pulse You are designing a 3D gradient echo pulse sequence, and you need a slab select pulse in the z dimension. You have 6 ms for the pulse, and want it to be as sharp as possible. You also have a peak RF amplitude constraint of 0.17 G.

1. What is the highest time-bandwidth you can allow, given that the maximum flip angle will be 90 degrees?

   Solution: Trying a few values we get

```
>> max(rfscaleg((pi/2)*wsinc(16,256),6))
ans =
    0.1572
>> max(rfscaleg((pi/2)*wsinc(17,256),6))
ans =
    0.1667
>> max(rfscaleg((pi/2)*wsinc(18,256),6))
ans =
    0.1762
```

   So $TBW = 17$ is OK, and $TBW = 18$ is not. We could go for more resolution if you want, and come put with a $TBW = 17.35$ that is pretty close to 0.17 G, but 16 or 17 is fine.

2. Assume we want the minimum slab thickness to be 8 cm. What is the gradient amplitude that this requires?

   Solution: Assuming $TBW = 17$, and a 6 ms pulse, the frequency bandwidth is

$$BW = \frac{17}{6 \text{ ms}} = 2.83 \text{ kHz}$$

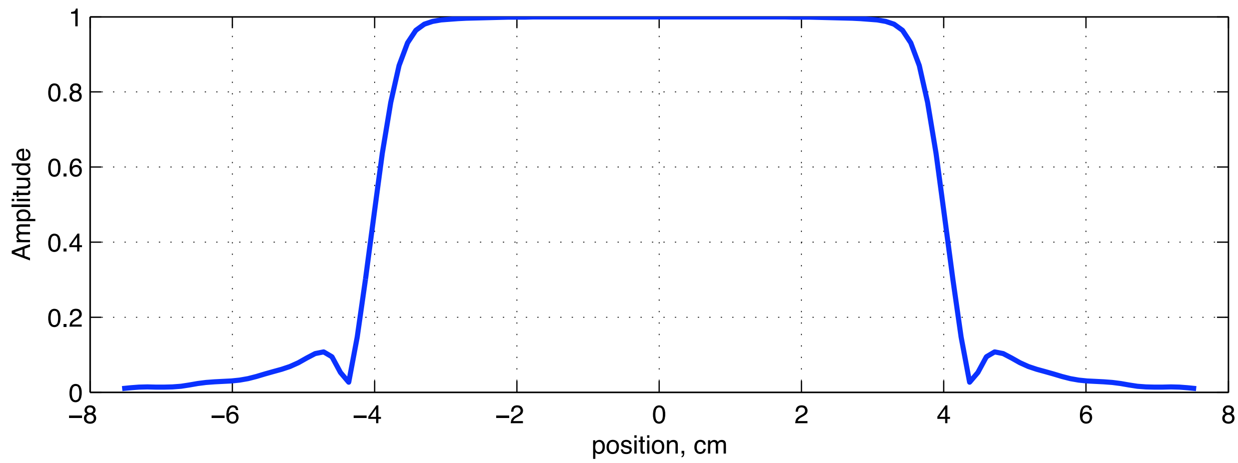
   We want this to be 8 cm, so $\frac{\gamma}{2\pi}G(8 \text{ cm}) = 2.83$ kHz, and

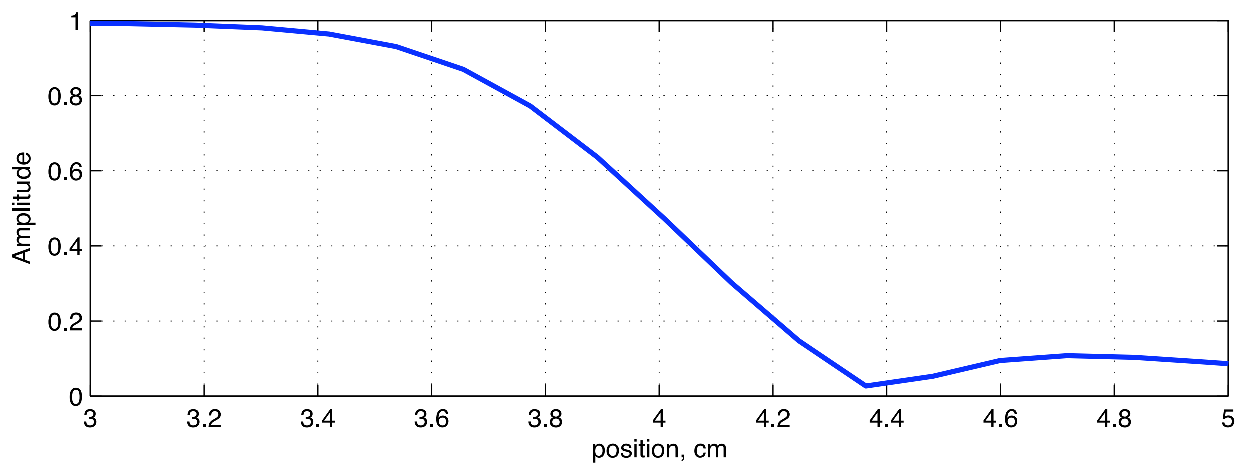
$$G = \frac{2.83 \text{ kHz}}{(4.257 \text{ kHz/G})(8 \text{ cm})} = 0.083 \text{ G/cm}$$

Your answer will differ, depending on the TBW product you used. The important point to note is that this is very low, only about 350 Hz/cm. This means that patient susceptibility shifts are going to distort the slab.

3. Simulate the slice profile. How wide is the transition band compared to the passband? Assume that the passband edge is 95% of the middle of the passband, and the stopband edge is 5% of the passband.

   Solution:





The slab thickness is about 8 cm which is what we designed for, and the transition width (neglecting the sidelobe) is about 0.7 cm. We'd expect the transition width to be about 8 cm/ (17/2) = 0.94 cm. This is off somewhat, but is due to how exactly the transition width is defined. It is on the right order, though.

5. Compute the Relative SAR of a Pulse Sequence A software model gives the SAR limit measured in terms of 1 ms rectangular 180 degree pulses ("hard" 180's). This depends on the patient weight, body part, and RF coil. The limit might be 100 hard 180s per second, for example. The power of a particular RF pulse is

$$P = \int B_1^2(t)\, dt.$$

The relative SAR of a particular RF pulse is it's power divided by the power in a 1 ms hard 180. The relative SAR of a pulse sequence is computed as the sum of the relative SAR's of each of the pulses, measured in equivalent hard 180's.

Write an mfile `rsar.m` which takes a normalized RF pulse and pulse duration, and returns the relative SAR, measured in equivalent hard 180's

```
>> eq180 = rsar(rf,t)
```

Assume that you are developing an SSFP sequence. The excitation pulse is a 1 ms TBW=2 windowed sinc. Assume you need a TR of 2.25 ms to do fat suppression with SSFP. What is the maximum flip angle you can allow, given that the SAR limit is 100 equivalent 180's per second?

Solution: The samples of the normalized RF pulse are

$$\gamma B_1(t_i)\Delta t$$

which is the flip angle in radians that each sample produces. What we want to compute is

$$\sum_i (\gamma B_1(t_i))^2 \Delta t$$

and compare this to the value of a hard 180

$$(\gamma B_1)^2 \Delta t = \pi^2 (1 \text{ ms}) = \pi^2$$

An m-file computes this ratio is given below:

```
function npi = rsar(rf,t)

% npi = rsar(rf,t)
%
%   rf -- rf waveform, scaled so that sum(rf) = flip angle
%   t  -- duration in ms
%
%   npi -- number of equivalent hard, 1 ms, pi pulses
%

dt = t/length(rf);

%convert radians to radians/ms
rf = rf/dt;

% compute power (radians^2)(ms)
p = sum(rf.*rf)*dt;

% normalize by value for a hard pi pulse
npi = p/(pi*pi);
```

To answer the flip angle question, first define a TBW=2 RF pulse

```
>> rf = wsinc(2,256);
```

This gives an RF pulse that sums to 1 radian. The relative sar of a single 1ms pulse is

```
>> rsar(rf,1)
ans =
    0.1754
```

With a TR of 2.25 ms, we are applying 444 pulses/second. We want the total relative sar to be 100. So we want to find a constant `theta` such that

```
>> 444*rsar(theta*rf,1)
```

gives an answer of 100. The value `theta` is then the maximum flip angle in radians. Experimenting, we can rapidly close in on an answer of

$$\theta = 1.135 \text{ radian} = 65^\circ.$$