

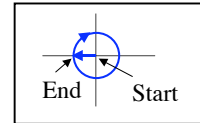
HOMEWORK #2
Due by 4 p.m., Wednesday 10/12/05

Readings:

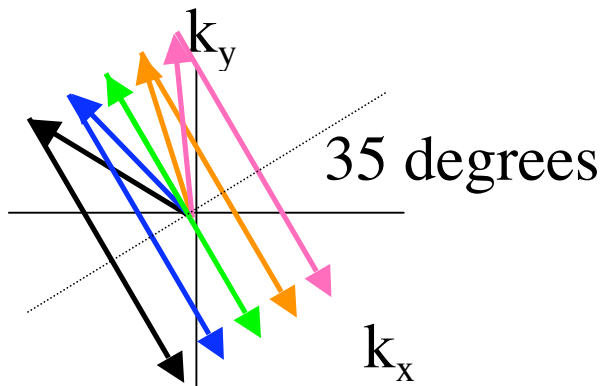
1. Review last week's reading as necessary. Read Chapter 2, sections 2.1-2.6.

Problems:

1. Draw a pulse sequence that achieves the k-space trajectory shown and give analytical expressions for the gradient waveforms. Note this starts at the origin and circles clockwise once with radius K . Assume that the time to make the circle is T .



2. Draw a pulse sequence that will achieve the k-space trajectory shown below. Describe a situation where you might use this pulse sequence. **Hint:** this is just the spin-warp sequence in class, rotated by 35 degrees in k-space.



3. Problem 12.8. For this problem, assume that T_1 , T_2 , and the local field offset $\Delta B(x,y,z)$ are unknown. You may use more than one repetition of the pulse sequence to determine T_1 .
4. Problem 13.13. For this problem, in addition to drawing the gradients, also draw ADC waveform that indicates when the ADC is collecting data.

MATLAB Exercise: This week's exercise uses the same image set as was used in Homework 1. The goal is to develop familiarity with relationship between 2D image space and 2D Fourier space, and to see for yourself how some of the images shown in lecture were derived.

Steps:

1. First download the file BE280Ahw1im.mat from the course website.
2. Load the image into MATLAB with the command: `load BENG280Ahw1im.`

3. Compute the 2D Fourier transform of the image with the command $Mf = \text{fft2}(Mimage)$; where the 2D transform will now be stored in the variable Mf . Remember to add the semi-colon at the end of the command, otherwise MATLAB will display all the numbers in the matrix! The command fft2 puts the zero-frequency value of the transform at the first indices of the matrix. For display it's convenient to put the zero-frequency value in the center of the matrix. To do this, type $Mf = \text{fftshift}(Mf)$;

4. **Resolution.** What happens when we zero out the outer regions of the Fourier transform?

(a) *Resolution reduction in the x-direction.*

```
>> res_span = 129+(-16:16);
>> Mf2 = zeros(256,256);
>> Mf2(:,res_span) = Mf(:,res_span);
>> Mf2 = fftshift(Mf2);
>> M_resx = ifft2(Mf2);
>> imagesc(abs(M_resx)); % This will show reduction of resolution in the x-direction.
```

(b) Demonstrate resolution reduction in the y-direction. **Hand in code and image.**

(c) Demonstrate resolution reduction in the x and y directions. **Hand in code and image**

5. **Missing data in k-space.** We can also zero out the inner regions of the Fourier Transform.

```
>> Mfzero = Mf;
>> Mfzero(ky,kx) = 0;
>> iMFzero= ifft2(fftshift(Mfzero)); % look at resulting image.
```

Zero out the following:

- (a) $kx = 129$; $ky = 129$
- (b) $kx = 1:256$; $ky = 129+(-16:16)$;
- (c) $kx = 129+(-16:16)$; $ky = 1:256$;
- (d) $kx = 129+(-16:16)$; $ky = 129 + (-16:16)$;
- (e) $kx = 1:2:256$; $ky = 1:256$;

For each set of parameters, plot out the Fourier transform and the resulting image. Give a qualitative explanation of why the image looks the way it does.

6. **Spikes in the data.** You can put a spike at location (kx,ky) in Fourier space with the following commands

```
>> Mfspike = Mf;
>> spike = 100e6;
>> Mfspike(ky,kx) = Mf(ky,kx) + spike; % add spike
>> iMFspike = ifft2(fftshift(Mfspike)); % look at resulting image.
```

Put spikes at:

- (a) $kx = 129$; $ky = 129$ – note this point corresponds to the center of k-space (i.e. $kx = 0$, $ky = 0$ in terms of the coordinates we used in class).
- (b) $kx = 161$; $ky = 129$
- (c) $kx = 161$; $ky = 161$

For each spike location, plot out the Fourier transform and the resulting image. Give a qualitative explanation for why the image looks the way it does. For example, what accounts for the direction of the artifacts?