Exercise: Ultrasound Doppler signal processing

Aim

To understand how the Doppler shift of ultrasound data is used to estimate the velocity of the object. Keywords: Continuous Doppler (CW), Pulsed Wave Doppler (PW), velocity resolution, range resolution, Nyquist limit, aliasing, clutter filter.

Exercises

Part 1: Theory

a Give an expression of received frequency as a function of transmit frequency (f0), the speed of sound (c), the target velocity (v), and the angle between the ultrasound beam and the movement of the target (ϕ).

b The Doppler shift is the difference from the transmitted frequency. Give an expression for the Doppler shift.

c Assume that we use Continuous Wave (CW) Doppler to measure the blood velocity in a vessel. Assume that: the transmit frequency is 2.5 MHz, the speed of sound c=1540 m/s, and the blood velocity is 1 m/s normal to the ultrasound beam.

  1 What is the received frequency from a sample volume in the vessel? And the Doppler shift?

  2 We steer the beam $\phi = 45°$ to the side. What is the received frequency and Doppler shift now?

d We shall measure the blood velocity in a constricted vessel using Pulsed Wave (PW) Doppler. The vessel lies r=7.7cm under the skin. We have positioned the beam $\phi = 45°$ relative to the bloodflow direction. Because of the constriction (the stenosis) in the blood vessel, the maximum velocity in the systolic part of the heart cycle is $v_{max} = 1.5 m/s$.

  1 What is the maximum pulse repitition frequency (PRF) one can use?

  2 If we use the maximum PRF; what is the maximum frequency we can use to measure the blood velocity at this angle without aliasing problems?

  3 We use the maximum PRF and frequency, as above. We move the sample volume so that the beam is $\phi = 30\,\text{deg}$ relative to the vessel. What is the apparent maximum velocity and Doppler shift?

e Argue whether these claims are true or untrue.

  1 Continuous Wave (CW) Doppler does noe have depth resolution.

  2 The maximum pulse repitition frequency (PRF), and thus the maximum velocity measurable with Pulsed Wave (PW) Doppler is inversely proportional to the distance to the sample volume, if the frequency is kept constant.

f Increasing transmit frequency implies a better velocity resolution in PW Doppler when the PRF is kept constant.

g Color Flow has made CW and PW Doppler superfluous.

h A short pulse length is important to get a good velocity resolution in PW Doppler.

Part 2: Pulsed Wave Doppler

In PW Doppler (see Figure 1) the ultrasound pulses are transmitted at a constant time interval. The received signal is complex demodulated and data samples are gathered in a buffer. A time window of the signal for a given depth is high pass filtered, and the resulting frequency content is estimated in a spectral analyzer and shown on screen. An example is shown in Figure 2. In this way, an image of the distrubution of velocities in the sample volume is shown as a function of time.

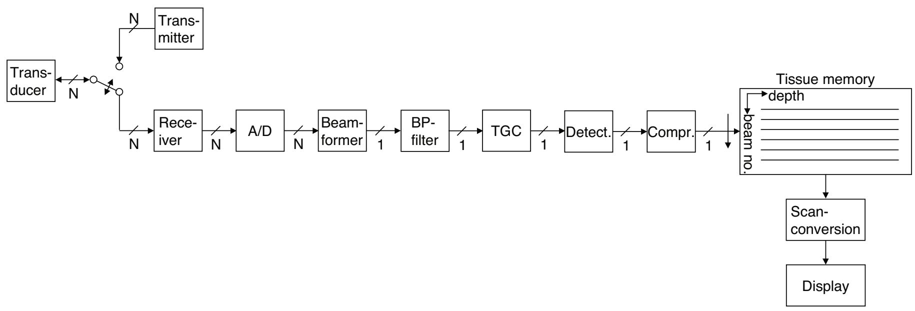

Figure 1: Block diagram of pulsed wave Doppler processing

The spectrum shows the blood flow through the mitral valve, with maximum velocities in magnitudes of about 1 m/s. This requires a PRF of about 5 kHz for frequencies of 2 MHz. We shall use the data sets from exercise 7 to learn about PW Doppler. Here the PRF (i.e. frame rate) is much lower, only 350 Hz, however the velocities are also much smaller. The principles remain the same.

1 Load the middle beam from the file `slowmotion.mat` in the same way as you did in exercise 8. Calculate the expected Doppler shift using the analytical expression for velocity from exercise 8.

2 The block diagram for the PW Doppler system contains a spectral analyzer. In this exercise we will use a simple form of spectral analyzer, the Fast Fourier Transform (FFT). Obtain a sliding window of the M-mode data with length 16 and use a Hamming window to weight the data. Use FFT in slow-time to calculate the frequency spectrum. Use zero-padding to 64 samples in the FFT (Nfft=64), and average in depth (fast-time) over about 20 samples at the middle of the beam. Remember that the time resolution in slow-time is 1/framerate

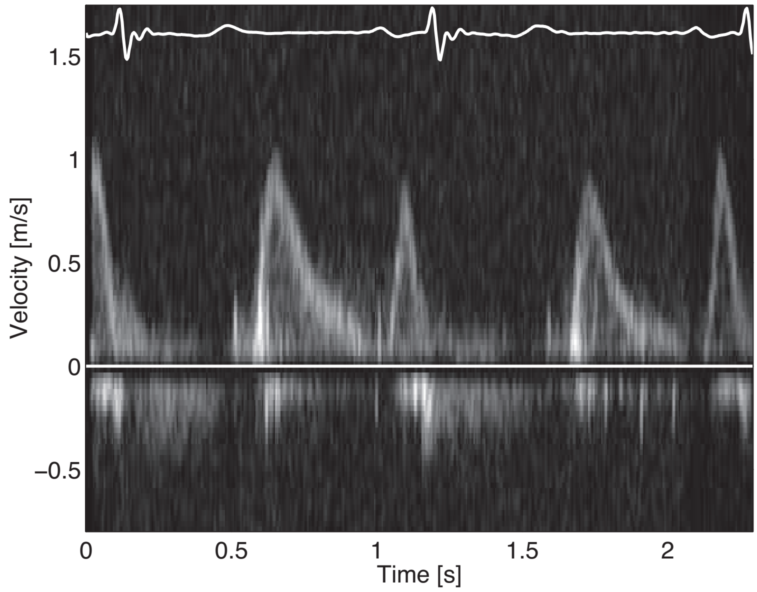

Figure 2: Pulsed wave Doppler spectrum from a patient with a leaking mitral valve.

to make the time and frequency axis. Make an image of the frequency spectrum as a function of time, that is with frequency (=Doppler shift) as y-axis and time (frameno, slow-time) as x-axis. Use a logarithmic compression with about 30-40 dB dynamic range. Plot the analytically calculated Doppler shift in the same figure. Comment.

```
% Tip: This is an exerpt of the code, you must fill in the rest
%------------------------------------------------------------
Nfft=64; %Zeropadding to length 64
crop=16;
P=zeros(Nfft, frames-crop+1);
for n=1:frames-crop+1,
iq=iqmm(depthindex,n+[0:crop-1])';
iq=iq.*(hamming(crop)*ones(1,length(depthindex)));
P(:,n)=mean(abs(fftshift(fft(iq,Nfft))),2);
end;
%Frequency axis
frequencyaxis=(([0:Nfft-1]/Nfft)-0.5)*framerate;
%Greyscale image of frequency specter in dB
colormap(gray(64));
P=imagelog(P,gain,dyn);
image(timeaxis,frequencyaxis,P);
```

A Hamming window will make the main lobe wider in frequency, but reduce the side lobes. This effect can be seen by commenting out the Hamming-weighting in the code above.

Part 3: Doppler shift and aliasing

Load the middle beam from the file `fastmotion.mat`. Repeat the processing as in the previous part, and make an image. Plot the analytically calculated Doppler shift in the same figure. Comment on what happens.

Make another image by stacking the Doppler spectrum three times (P=[P;P;P];). Extend the Doppler axis appropriately:

```
frequencyaxis=([0:Nfft-1]/Nfft)-0.5;
frequencyaxis=[frequencyaxis-1, frequencyaxis, frequencyaxis+1]*framerate;
```

Plot the calculated Doppler shift over the image. Indicate the Nyquist limit in the image (positive and negative).

Part 4: Doppler sound (You will need a sound card)

The largest velocities in blood flow imaging are up to 5 m/s, for example with an insufficiency (leakage) in the mitral valve. With a transmit frequency in the range 2-5 MHz, and a speed of sound at 1540 m/s, this gives a maximum Doppler shift of about 10 kHz. This is within the range of hearable frequencies, and on todays ultrasound machines it is common to modulate sound from measured Doppler spectra as sound. The movement in the file `slowmotion.mat` gives a maximum Doppler shift of about 200 Hz. In this part of the exercise we will generate sound from this dataset.

Load the middle beam in `slowmotion.mat`. Extract the IQ-signal for a single sample in the middle of the beam:

```
x=iqmm(round(end/2),:);
```

The sequence contains about two oscillations of the movement. Extract more signals to lengthen the sound signal:

```
x=[x,x,x,x,x];
```

Use the matlab-command SOUNDSC and play the real part of the signal:

```
soundsc(real(x),framerate);
```

Tip 1 The performance of soundsc can be poor on some laptop computers (you get no sound). Try scaling the signal manually, so it is in the range of [-1 1]. Try using `sound(real(x_scaled), framerate, 8 OR 16)`.

Tip 2 If you still can not get any sound. Try to resample the data to 8192Hz, using the `resample()` command. Then use `sound(real(x_scaled_resampled),8192 (or leave empty as this is the default))`

Make a plot of real(x) as a function of time. What does it look like?

Use a sliding FFT on the real value of iqmm and make an image (Tip: Use the code from part 2, and set `fft(real(iq),Nfft)` instead of `fft(iq,Nfft)`). How does the center frequency change in the signal? What is the difference compared to the spectrum of the complex signal?

Challenge: Stereo sound In the same way as the FFT of a real signal does not discriminate between positive and negative frequencies, the ear is not capable to hear the difference between positive and negative frequencies. In the Doppler instruments, this is solved by sending positive and negative frequency shifts to separate channels in a stereo sound system.

If you have a stereo soundcard on your computer, you can make a matrix with two columns in Matlab to obtain a stereo sound signal. Process the data so that you obtain positive and negative frequency shifts as separate signals(Hint: Hilbert-transform).

Part 5: Clutter and clutter filter

In practical situations using Doppler, for example when measuring blood velocities through a heart valve, the received signal from blood will contain echoes from surrounding tissue. These echoes can be up to 80-100 dB stronger then the blood signal. Such strong, slow-moving signal components are called clutter. Another source of clutter are reflections from the body-wall.

In the file `slowmotion.mat` the whole phantom is moving (actually the probe is moving), and except a couple of strong reflections in the near-field that are constant over time, the data do not contain any clutter. The file `slowmotion_clutter.mat` is just like slowmotion.mat except that a slow-moving artifact has been added. This clutter component gives much stronger echoes than the time varying component of the phantom. We see this if we make a M-mode image of the data, see figure 3. The dark curve shows the movement of the phantom without clutter. It is almost impossible to see the movement beneath the clutter. This is the same case as in blood flow imaging. The strong echoes from tissue hide the Doppler signal from the blood.

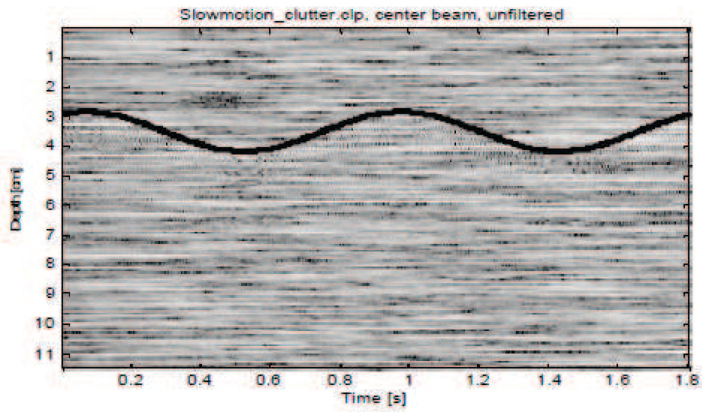

Figure 3: Mmode of center beam from slowmotion.mat, obstructed by clutter. The curve is the actual phantom movement

1 Make an image of the Doppler spectrum for the file `slowmotion_clutter.mat` as in part 2. How does it look? What is the difference between this spectrum, and that of the file `slowmotion.mat`?

2 To remove the clutter, a high pass filter is used in the block diagram for the PW Doppler system. Use a highpass filter to filter the data in "'slow time"' and make a new Doppler-spectrum. How does the spectrum look now? Vary the length of the filter from N=2 til N=10. What is the optimal length to remove all the clutter?

   Tip: A simple and intuitive way to higpass filter the signal, is to first make a lowpass filter (that is to take the average of N samples) and then subtract this from the data:

```
% A simple lowpass filter:
```

```
b=ones(1,N); %=boxcar(N). May also use hamming(N), hanning(N), ....
b=b/sum(b); %Normalization of filter coefficients
iq\_lp=filter(b,1,iqmm,[],2); %Filter along rows
iq\_hp=iqmm-iq\_lp; %Subtract low pass component:
```

3 Make a M-mode image of the highpass filtered signal.

4 Make sound of the unfiltered and filtered signal. Comment on the difference.

Part 6. Blood flow measurement using Doppler

The main application of Doppler in medical imaging is to measure blood flow. The data we have used so far are not realistic for blood flow measurements. The file `Dopplerdata.mat` contains a dataset with realistic Doppler data. Use the command `load Dopplerdata.mat` to load the file into Matlab. The variables in the file are:

```
iq:  IQ-demodulated data, 103 range samples x 2032 beams
prf:  pulse repetition frequency 2.5 kHz
f0:  pulse frequency 2.5 MHz
fs:  sampling frequency along the axis , 10 MHz
```

Use the same code as in part 5 to generate the Doppler spectrum. Use a depth segment of about 10 samples (ex. depthindex=[70:80]) and Nfft=256. Make a logarithmic greyscale image. Vary the length of the segment between 8, 16, 32 og 64. Find a combination of segment length, high-pass filter settings and logarithmic compression that gives the best Doppler spectrum. The dataset contains a lot of aliasing. Stack 3 spectra on top of each other to ease interpretation. What is the maximum velocity in the data. What is the Nyquist limit? What PRF is needed to avoid the aliasing?