Laboratory exercise Beamforming – Home assignment

Introduction

Aim

After this laboratory exercise the student should understand how beamforming is performed in modern ultrasound scanners.

Equipment

Matlab software on any computer

3 Matlab matrices with pre-beamformed ultrasound data

1 Matlab matrix with post-beamformed ultrasound data

(Matlab files can be found in Live@Lund)

Assignment

Transform pre-beamformed RF-data to a normal B-mode image by implementation of a beamforming-algorithm.

Personal guidance and report

The assignment should be performed in groups of two students. Personal guidance will be given during exercise sessions 6/2 and 13/2. (Registration via registration-note)

On the 20/2 each group should have handed in their solutions (code + resulting images) before lunch (to magnus.cinthio@bme.lth.se). In the afternoon the groups should be prepared to discuss their solutions.

Literary reference

Primarily: Diagnostic Ultrasound – Physics and Equipment, p. 27-43

Instructions

During image construction the lines (columns) are typically constructed one by one. To construct an image of 128 lines (columns) it is often said that the ultrasound scanner emits 128 successive pulses. This is somewhat inaccurate, because each such "pulse" actually consists of typically 64 pulses. For each intended image line the ultrasound scanner uses a specific aperture = a certain number of elements (in the probe) where the middle element is just above the tissue section (line) that will be imaged. The aperture consists of several elements to e.g. focus on a certain depth.

One image line (column) is thus created by the sum of up to 64 pulses which have been emitted from 64 different elements. After the first pulse has been emitted (usually from the outer elements in the aperture) there is a short time delay before the data collection begins. The reason is to ensure that there is enough time for all elements in the aperture to emit pulses and then be ready to receive echoes. Then the data collection begins and the sampling is done simultaneously on all 64 elements. The result is that the information used to create a single line in the ultrasound image consists of a 2D matrix (number of samples x 64 elements). An entire ultrasound image is then a 3D matrix (number of samples x 64 elements x 128 lines).

This recorded raw signal is the direct response of the pressure on the transducer elements. This pressure is not only caused by the ultrasound itself, but also by movements of the transducer and the tissue. These movements are of lower frequency but of higher amplitude. Thus, in order to obtain a good image, the signal usually needs to be high-pass filtered.

It is recommended that you start to experiment with filtering on the post-beamformed data, so that you can get a direct response that the filtering works. After filtering, the result can be viewed in a B-mode image using the Hilbert Transform:

```
Image=abs(hilbert(postbeamformed)) %where the "postbeamformed"-variable is already filtered
figure; imagesc(Image); colormap(gray)
```

It could also be a good idea to look at just one line:

```
figure; plot(postbeamformed(:,1))
```

To examine a "pre-beamformed" B-mode image the 2:nd dimension can be summed using:

```
ThreeToTwo=squeeze(sum(prebeamformed,2));
Image2=abs(hilbert(ThreeToTwo));
figure; imagesc(Image2); colormap(gray)
```

(Note that this is not high-pass filtered by default.)

Hints: Once you solved the beamforming of the 2D matrix (a single line/column) the code could just be repeated 128 times to get the image. Also, when beamforming the 2D matrix, only consider echoes coming straight beneath the center of the aperture (64 elements) – this is the line you want to image.

A Matlab file is loaded by the command "load" and you could use the commands "butter" and "filtfilt" for filtering.

It is a good idea to write the code in a Matlab "m-file" with the aid of Matlab's editor. Help in Matlab can be accessed through F1. The assignment can be solved with simple Matlab commands.

Assignment

Your assignment is to transform pre-beamformed RF-data into a normal B-mode image by implementation of a beamforming-algorithm containing **dynamic focusing**.

Suggested work progress

1. Understand how to apply high-pass filtering (using post-beamformed data).
2. Create the dynamic focus. (The data starts with 3mm dead zone. Don't forget to compensate for this.)
3. High-pass filter and show the result using the Hilbert Transform.
4. (Optional: Add apodisation and dynamic aperture to the beamforming)