Exercise 9 - Ultrasound signals from moving targets

Aim

To understand how object movement affects the backscattered ultrasound signal. Keywords: Timeshift, phaseshift, Nyquist limit, aliasing, angle dependence.

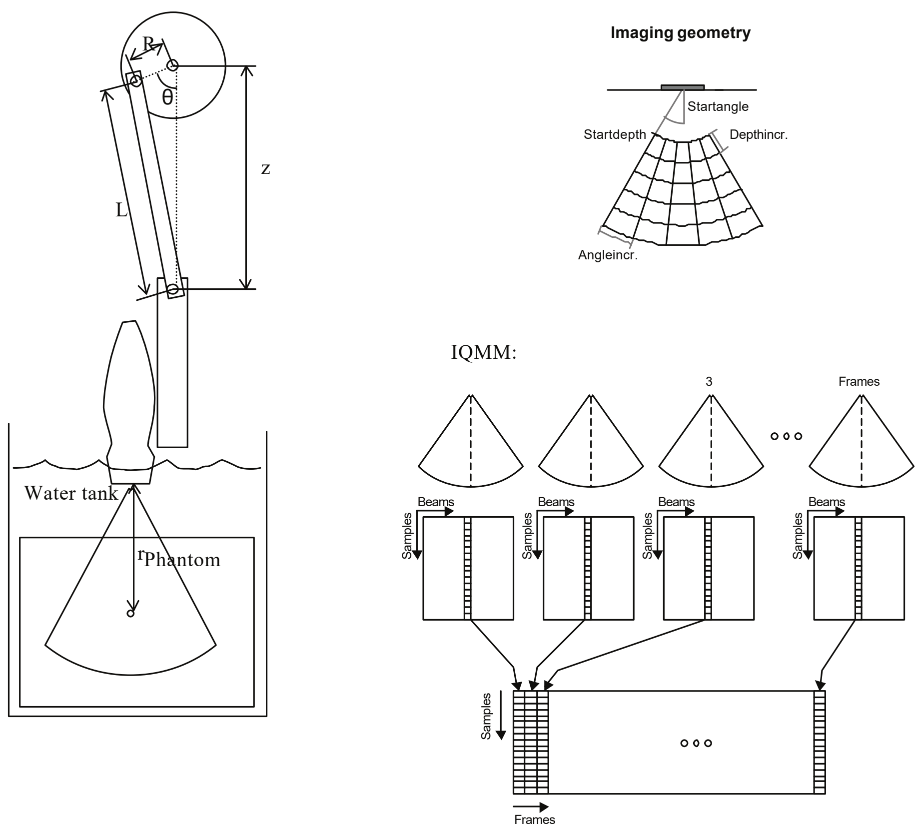

Figure 1: Phantom setup.

Background

IQ-data is aquired with the probe mounted on a motorized piston, as in Figure 1. The scatterers are stationary, but appear as moving in the image, because the probe is moving. The IQ-data is aquired at a frame rate of about 350 frames per second. To achieve this high frame rate, the number of transmitted pulses must be reduced, and therefore each image has a limited amount of beams.

Assuming that $R \ll L$, it can be shown that the distance between a fixed point and the probe varies approximately as:

$$r(t) \approx -R\cos(2\pi(t-t_0)/T) + L \tag{1}$$

The velocity of the piston is found by differentiation of z:

$$r'(t) = 2\pi R/T \sin(2\pi(t-t_0)/T) \tag{2}$$

At t_0 the probe is closest to the phantom, giving the minimum value of r.

Exercises

Part 1: M-mode

- Load IQ data and corresponding settings from the file `slowmotion.mat`, and make a new matrix containing the beam in the middle of the sector from every frame. The data in the 3D IQ matrix is organized as *samples x beams x frames*.
- Make an M-mode image of this matrix with distance from the probe as y-axis and time along the x-axis. Use logarithmic compression and adjust the gain and dynamic range.
- Estimate the rotation period T, the time t_0, and the exenter distance R from the image. Insert the values into the analytic expression, and plot $r(t)$ on top of the image. $L = 10cm$. How is the calculated movement compared to the observed movement?

Tips:

- The Matlab function `squeeze` removes singleton dimensions.
- To make the time axis, you might want to use the frame rate, which is given in the parameter `s.Framerate_fps`. Use `s.iq.DepthIncrementIQ_m` to make the distance axis. The number of beams, samples and frames can be read from the parameters named `s.iq.SamplesIQ`, `s.iq.BeamsIQ` and `s.iq.FramesIQ`.
- The Matlab function `ginput` may be useful for manual measurements in figures.

Part 2: Timeshift in the RF-data

- Insert the values of T, t_0 and R into the analytical expression $r'(t)$, and plot the resulting function versus time. Select center beams $x(t)$ and $y(t)$, the same spatial beam from frames 42 and 43 of IQ-data from the M-mode matrix . Use $r'(t)$ to estimate the velocity during these frames. Also calculate the radial displacement between the consecutive frames and the corresponding time shift in the RF-signal.
- Convert the two beams of IQ-data to RF-data with a sampling rate of 200 MHz, use the helper function `iq2rf`. Plot both beams in the same figure using different color/linetype. Zoom in on a small segment of the signal near the highest amplitudes, and manually read the approximate timeshift between the pulses in nanoseconds (ns).

- A timeshift equal to the pulse period $T_p = 1/f_0$ corresponds to a phaseshift of 2π radians. Calculate the phaseshift in the RF-signal for this image both analytically and for the estimated timeshift.

Tip: The sampling frequency in axial direction is `s.iq.frsIQ_Hz`, the demodulation frequency is `s.iq.fDemodIQ_Hz`.

Part 3. The autocorrelation method.

The time shift observed in the RF data in part 2 should correspond to a phase shift in the IQ data. This motivates the use of the autocorrelation estimator:

$$\Delta\theta = \arg\{< x_{iq}(r)^* y_{iq}(r) >\} \tag{3}$$

Here r is an index along the beam direction, $\arg\{\dots\}$ means phase angle (`angle` i Matlab) and $*$ means complex conjugate (`conj`). $< \cdots >$ denotes the expectation value.

- Use the same two beams as in part 2. Use the autocorrelation method to estimate the phaseshift between the two pulses, and plot the phaseshift for all axial samples. Average (`mean`) 20 axial samples near the highest amplitudes to find the expectation value. Compare with the calculated and estimated phaseshift in part 2.
- Give an expression for the relation between the velocity v, frame rate FR, wavelength λ and the phaseshift in the IQ-signal $\Delta\theta$.
- Now estimate the phaseshift for all frames in the image sequence. Convert phase shift to velocity, and plot it together with the analytically calculated velocity.

Tip: Matlab executes FOR-loops slowly, so it is better to vectorize the operations: Make two matrices, where the first one contains beam 1..N-1, and the second contains beam 2..N, where N is the number of beams in the M-mode matrix. Complex conjugate the first matrix and multiply element by element with the other. Average along each column, then find the phase angle.

Part 4. Aliasing

Run the same code on the file `fastmotion.mat`. Generate the M-mode image for the center beam as for the first file. Estimate the phaseshift in the IQ-data for the whole sequence, convert to velocity, and plot the result. Comment on what happens for high velocities.

The phenomenon is called aliasing, and the limit for when this happens is called the Nyquist limit. Derive the Nyquist limit for the velocity expressed with frequency f, speed of sound c and frame rate FR.

Assuming a speed of sound of 1540 m/s, what is the Nyquist limit for the data in this exercise? How much should we increase the frame rate to be able to measure velocities of up to 20 cm/s using the same frequency? What if the frequency is reduced to 1.67 MHz?