

Will update into an html in final version!

Rebeamforming algorithm:

Inputs - 2d RF array (length and width of array also needed), number of elements, channelspacing, sound speed

Outputs - 2d rebeamformed RF array

Effectively applies another beamforming procedure on the RF array (turning it into the 2d rebeamformed RF array) using the input parameters.

Rebeamforming works by calculating the time delay function (using speed of sound and channel spacing) across elements and summing together RF elements corresponding to this delay. The resulting RF array is then normalized (averaged out) and returned as a corrected RF array for PA imaging.

Integration into Ulterius Platform:

After analyzing the code from spring break, I have identified the main function needed to allow for reprocessing of RF beamforming data.

The function we must modify is the onNewData function, which takes in US data from the US machine. The interface allows users to specify the data input (B mode, M mode, RF mode, etc.)

To achieve basic functionality, we first intend to tweak the RF mode to have additional PA rebeamforming.

This will be achieved by effectively applying the beamforming function we developed into the data received from the Ulterius onNewData function.

Currently, we have successfully built the interface and are looking at the RF data format provided. Once we understand exactly how it is acquired into the system, we can retweak our beamforming algorithm as needed to properly process this data.