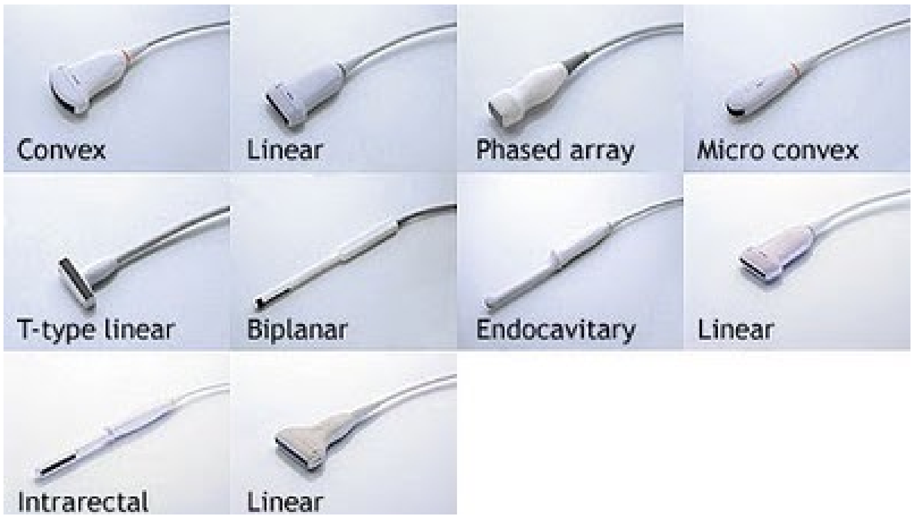
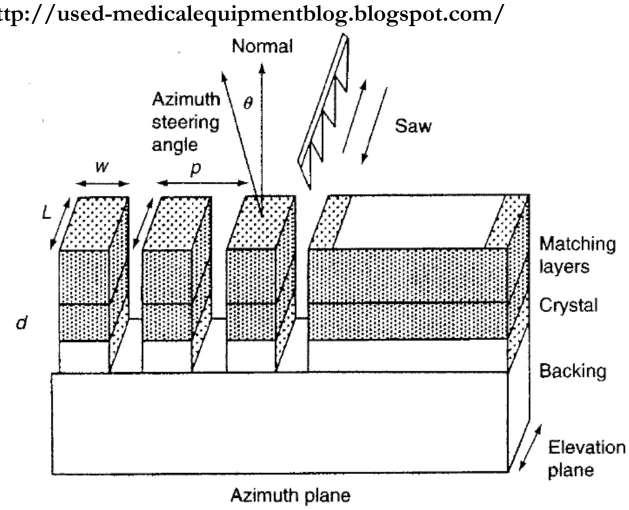
Ultrasound Notes 3: Array Systems

While early US systems used a single focus transducer and mechanical sweeping of the transducer to different angles, all modern US systems are array systems where focusing is not preset and beam steering is done through time delays associated with each element of the array. There are many array types:



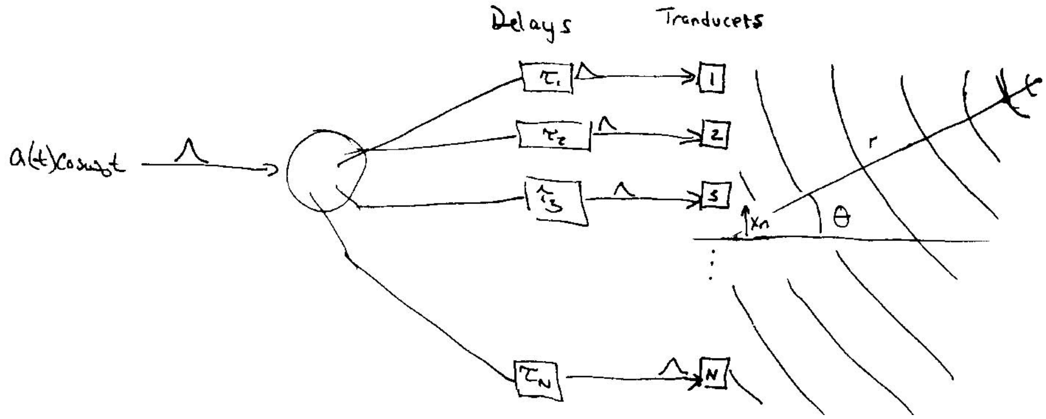
http://used-medicalequipmentblog.blogspot.com/



where d is close to ½ λ along the depth direction in the crystal, pitch (p) is close to ½ λ in the water (note: the speed of sound is significantly higher in the crystal than in the water). L>>d, w>>d, crystal vibrate on thickness mode. Two electrodes on either side of the crystal. "kerf" is p-w, spacing between elements.

Transmit mode: Focusing and Beam Steering

Typical array system shown here in transmit mode:



where x_n is the x coordinate of the n^{th} element (0 is at the center of the array).

For focusing at $(z_0, x_z = 0)$ set the delay to:

$$\tau_n' = \tau'(x_n, z_0, x_z = 0) = \frac{x_n^2}{2cz_0}$$

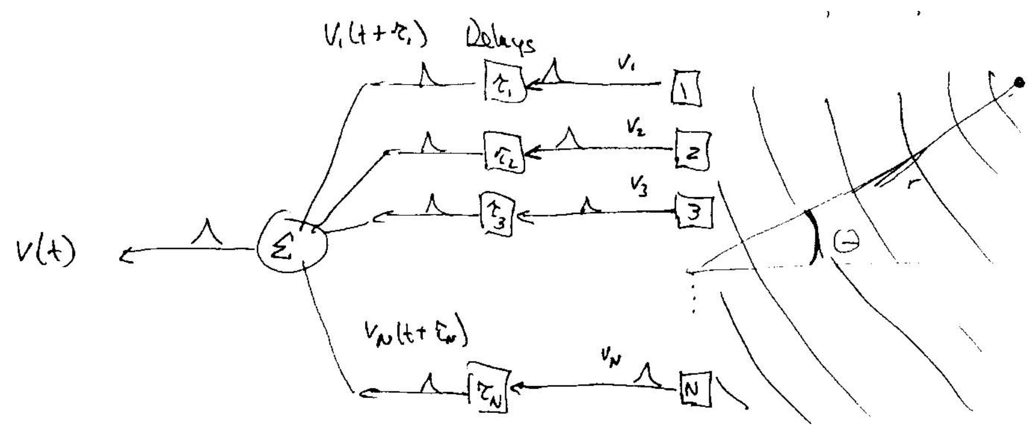
(Actually, this is a negative delay – to focus on-axis in the Fresnel zone, we require that the edge elements (e.g. large $|x_n|$) fire before the center element because they have farther to propagate.)

In polar coordinates, for focusing at (r_0, θ_0):

$$\tau'_n = \tau'(x_n, r_0, \theta_0) = -\frac{x_n \sin\theta_0}{c} + \frac{x_n^2 \cos^2\theta_0}{2cr_0}$$

A focal depth and direction (r_0, θ_0) is selected once when the pulse is transmitted. Once it leaves the transducer, it can no longer be changed.

Receive mode: Delay-Sum Beamforming



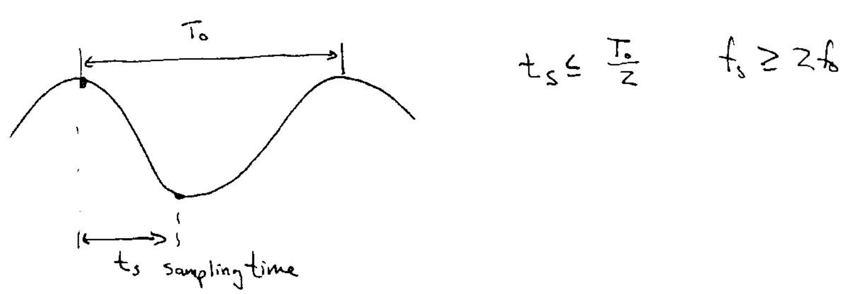
- Signals from each transducer are delayed and summed to produce a signal localized to the focal point, e.g. (r_0,θ_0):

$$v(t)=\sum_{n=1}^{N} v_n(t+\tau'_n)$$

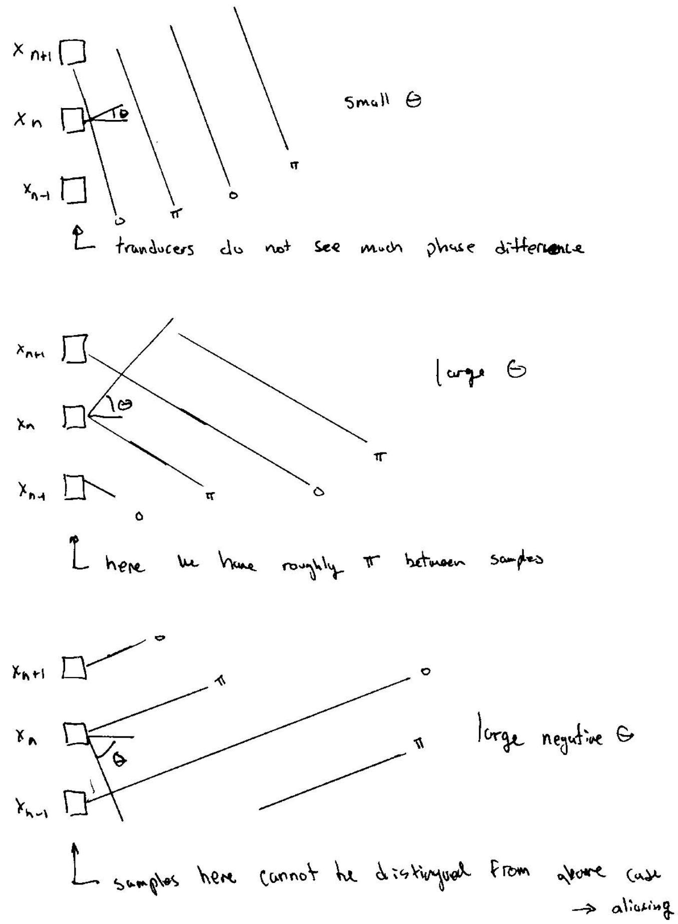
- Signal from other angles are suppressed by destructive interference of the wavefronts when summed together (recall the each pulse is made up of modulated waves at a carrier frequency, f_0.
- Unlike the transmit case, the received data can be combined in many different ways. In other words, we can separately focus for any point in time (depth plane, r). Also we can focus to any angle at by reprocessing the data. Typically, "dynamic" focusing is used to focus for each depth plane – here τ'_n changes for every depth $r = ct/2$.

Sampling in Space

At each point in time, the array is actually sampling the instantaneous pressure wave as a function of space. Aliasing is potentially a problem. A quick review of sampling:



Aliasing can be prevented if less than π has accrued between neighboring samples. For our case of waves impinging upon the detector array, sampling becomes a bigger issues if the source of the waves is coming from a large angle, θ:



The sampling requirement for this case can be written as the maximum difference in propagation delay between a point (r,θ) an two neighboring transducer elements cannot lead to phase accrual of greater than π:

$$\left|\left(\tau(x_n,r,\theta)-\tau(x_{n+1},r,\theta)\right)\cdot\omega_0\right| \le \pi \qquad \forall r,\theta,n$$

assuming the element spacing is d (e.g. $x_{n+1} = x_n + d$), (d = pitch) then:

$$\left|\left(\tau(x_n,r,\theta)-\tau(x_{n+1},r,\theta)\right)\cdot\omega_0\right|$$

$$=\left|\left(r-x_n\sin\theta+\frac{x_n{}^2\cos^2\theta}{2r}-r+(x_n+d)\sin\theta-\frac{(x_n+d)^2\cos^2\theta}{2r}\right)\cdot\frac{\omega_0}{c}\right|$$

$$=\left|\left(d\sin\theta-\frac{2x_nd\cos^2\theta}{2r}-\frac{d^2\cos^2\theta}{2r}\right)\cdot\frac{\omega_0}{c}\right|$$

The worst case is for plane waves, that is $r\to\infty$:

$$\left|\left(\tau(x_n,r,\theta)-\tau(x_{n+1},r,\theta)\right)\cdot\omega_0\right|=\left|d\sin\theta\cdot\frac{\omega_0}{c}\right|$$

$$=\left|d\sin\theta\cdot\frac{2\pi}{\lambda}\right|\le\pi$$

and thus:

$$\left|d\sin\theta\right|\le\frac{\lambda}{2}$$

and if we want to unambiguously distinguish between arrival directions (points of reflectivity) over a full π of angle, then:

$$\text{pitch }\ d\le\frac{\lambda}{2}$$

In this case, the number of transducer elements, N, should be:

$$N=\frac{2a}{d}\ge\frac{4a}{\lambda}=2\cdot(\text{Numerical Aperture})$$

The "Numerical Aperture" for an array is the size of the transducer in terms of the number of wavelengths (e.g. N.A. = $2a/\lambda$).

Comments:

- We've just talked about the array sampling a pressure wave propagating towards the transducer. It is important to realize that sampling also occurs during transmit as well – in order to unambiguously transmit a wave in a particular direction (over π angles) , we also need to satisfy the above transducer spacing.
- Below under discussion of the point spread function, we will see the effects of insufficient sampling by the array.

Angular Sampling of the Object

For this analysis, we will consider the far-field (or focal plane) solution:

$$p(r,\theta) = \frac{\cos\theta \cdot e^{ikr}}{r} \Im\{s(x)\}\Big|_{u=\frac{\sin\theta}{\lambda}}$$

$$= K'' S\left(\frac{\sin\theta}{\lambda}\right)$$

As described above, we must transmit our beam at a particular angle (θ_i) and thus, this axis is discretized as well:

$$S(\cdot) \text{ is sampled at locations } \frac{\sin\theta_i}{\lambda}$$

This is a reverse sampling argument – we are sampling the FT of $s(x)$ and thus the samples of $S(u)$ must be positioned at least as close as half the maximum extent of $s(x)$. Thus, if $s(x)$ goes from [$-a$, a], then:

$$\Delta\left[\frac{\sin\theta}{\lambda}\right] \leq \frac{1}{2a}$$

$$\Delta \sin\theta \leq \frac{\lambda}{2a} = \frac{1}{\text{Numerical Aperture}}$$

This is the solution for the receive-only or transmit-only case.

Recall that the actual beam function is:

$$B(r,\theta) = h_T(\omega,r,\theta) h_R(\omega,r,\theta)$$

$$= [p(r,\theta)]^2$$

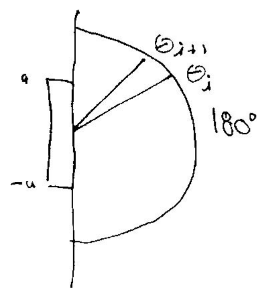
$$= [K'']^2 \left[S\left(\frac{\sin\theta}{\lambda}\right)\right]^2$$

and this is the function that is used to sample the object in the angular direction. Since we must now sample the square of the FT of the space limited aperture function, we must know the maximum extent of its IFT, which is $s(x)* s(x)$. If if $s(x)$ goes from [$-a$, a], then

$s(x)* s(x)$ is bounded by a function that extends of [-2a, 2a]. For example, if $s(x) = \text{rect}(x/2a)$, then $s(x)* s(x) = \text{triangle}(x/2a)$, which goes from -2$a$ to 2a. Thus:

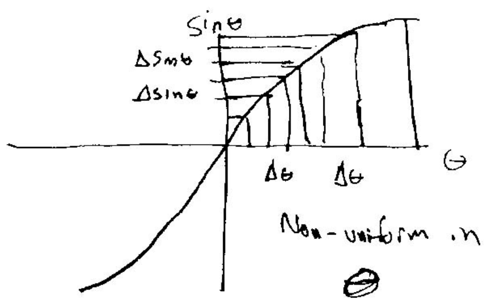
$$\Delta \sin\theta \le \frac{\lambda}{4a} = \frac{1}{2 \cdot \text{N.A.}}$$

- How many beams to sample π?



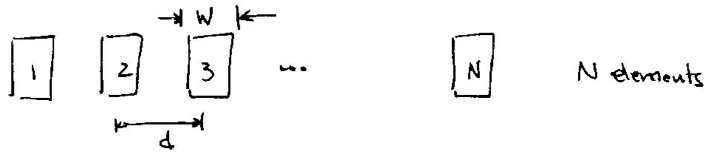
$$\frac{\max(\sin\theta) - \min(\sin\theta)}{\Delta(\sin\theta)} = \frac{1-(-1)}{\lambda/4a} = \frac{8a}{\lambda} = 4 \cdot \text{N.A.}$$

- Is sampling uniform in θ? No, sampling is uniform in sinθ.



- Shouldn't we just sample more finely than we need to? No.
  - It takes extra time (you can't transmit the next pulse at a new angle until the last one is received).
  - It adds no new information.
- What happens if we don't sample finely enough? We might miss an object feature – it would be better to increase the beam width (lower lateral resolution, perhaps be decreasing the N.A.) to sample fewer beams.

Point Spread Function for Discrete Transducers



Consider that the transducer width is $D = 2a = Nd$, and we can model the source function as:

$$s(x) = \left[\frac{1}{d}\operatorname{comb}\left(\frac{x}{d}\right)\operatorname{rect}\left(\frac{x}{2a}\right)\right] * \operatorname{rect}\left(\frac{x}{w}\right)$$

and the FT is:

$$\begin{aligned} S(u) &= w\operatorname{sinc}(wu)\cdot[\operatorname{comb}(du) * 2a\operatorname{sinc}(2au)] \\ &= w\operatorname{sinc}(wu)\cdot\sum_{n=-\infty}^{\infty}\operatorname{sinc}\left(2a\left(u-\frac{n}{d}\right)\right) \end{aligned}$$

Substituting $u = \sin\theta/\lambda$, in the far field (or focal plane) for an on-axis beam, we get:

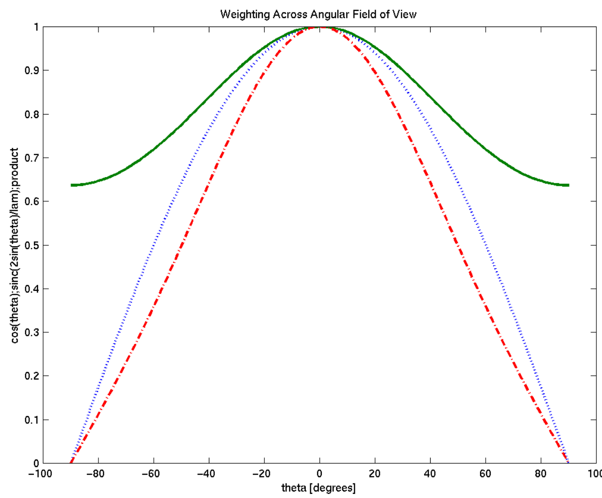
$$p(r,\theta) = K\cos\theta\operatorname{sinc}\left(w\frac{\sin\theta}{\lambda}\right)\cdot\sum_{n=-\infty}^{\infty}\operatorname{sinc}\left(2a\left(\frac{\sin\theta}{\lambda}-\frac{n}{d}\right)\right)$$

- the first sinc function provides additional weighting as a function of θ. This could reduce the angular "field of view" by restricting over what range of angles we can effectively transmit to/receive from.

Let's look at these effects for a Nyquist sampled array ($d=\lambda/2$). First we look at the effect of the terms weighting the angular field of view:

$$\cos\theta\operatorname{sinc}\left(w\frac{\sin\theta}{\lambda}\right)$$

We know that $w < d$, so $w < \lambda/2$. The worst case (narrowest) is for $w = \lambda/2$. Here are plots of these functions (and the product of these functions – the lowest line on the plots):



Thus the weighting, while present, is not significantly different than the original $\cos\theta$ weighting. These weighting functions are independent of the numerical aperture.

Sampling Effects

For the Nyquist sampled array ($d=\lambda/2$), we get:

$$p(r,\theta) = K\cos\theta\,\mathrm{sinc}\!\left(w\frac{\sin\theta}{\lambda}\right)\cdot\sum_{n=-\infty}^{\infty}\mathrm{sinc}\big((\mathrm{N.A.})\cdot(\sin\theta-2n)\big)$$

The argument of the sinc is zero only for n=0 – the main lobe (desired beam) and this occurs at θ=0. For $d=\lambda$, we get:

$$p(r,\theta) = K\cos\theta\,\mathrm{sinc}\!\left(w\frac{\sin\theta}{\lambda}\right)\cdot\sum_{n=-\infty}^{\infty}\mathrm{sinc}\big((\mathrm{N.A.})\cdot(\sin\theta-n)\big)$$

The argument of the sinc is zero for.

n	θ	Type
0	0	Main Lobe
1	$\pi/2$	Grating
-1	$-\pi/2$	Grating

For $d=2\lambda$, we get:

$$p(r,\theta) = K\cos\theta\,\mathrm{sinc}\!\left(w\frac{\sin\theta}{\lambda}\right)\cdot\sum_{n=-\infty}^{\infty}\mathrm{sinc}\left((\mathrm{N.A.})\cdot\left(\sin\theta-\frac{n}{2}\right)\right)$$

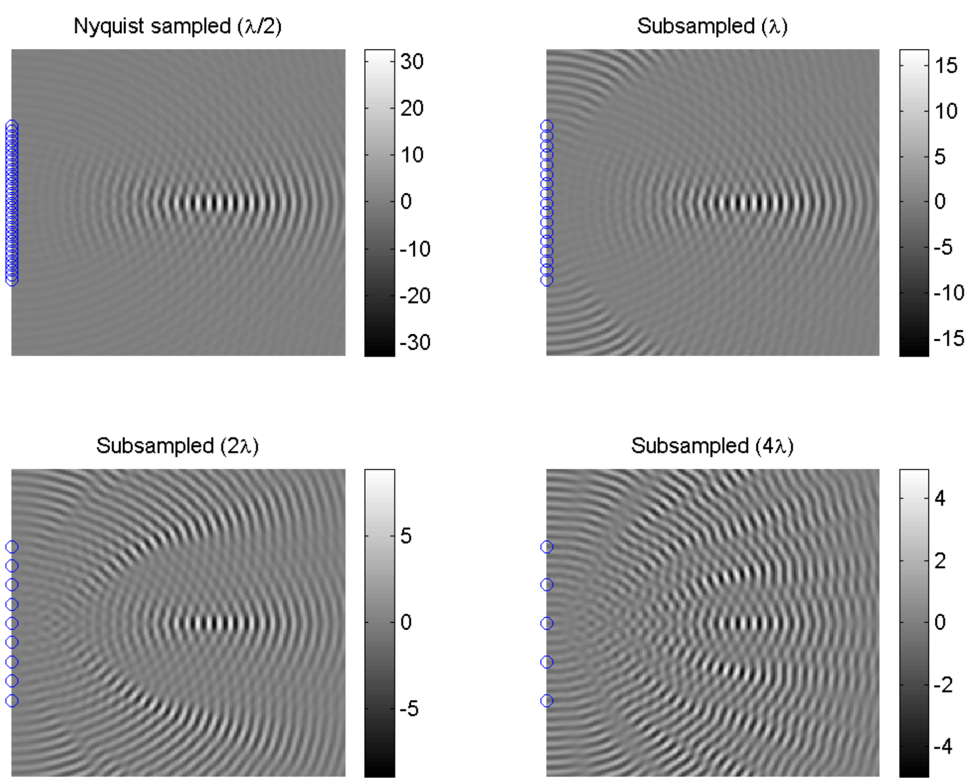
The argument of the sinc is zero for.

n	θ	Type
0	0	Main Lobe
1	$\pi/6$	Grating
-1	$-\pi/6$	Grating
2	$\pi/2$	Grating
-2	$-\pi/2$	Grating

Comments:

- Observe that he main lobe doesn't change in each case.
- The grating lobes are aliases resulting from subsampling of the spatial waves by the discrete transducer.
- In the sum of sinc functions, the n=0 term is the main beam
- The $n \neq 0$ terms are "grating" lobes – additional responses at angular locations other than the desired angle (in this case $\theta_0 = 0$).

Main and grating lobes in the following simulation of discrete transmitters:



Grating Lobes with Beam Steering

We can derive the above expressions for the case where the beam is steered to another location, θ_0:

$$s(x) = \left[\frac{1}{d}\mathrm{comb}\left(\frac{x}{d}\right)\mathrm{rect}\left(\frac{x}{2a}\right)e^{-i2\pi x \sin\theta_0/\lambda}\right] * \mathrm{rect}\left(\frac{x}{w}\right)$$

which has as its transform:

$$S(u) = \frac{Dw}{d}\mathrm{sinc}(wu)\cdot\sum_{n=-\infty}^{\infty}\mathrm{sinc}\left(2a\left(u - \frac{sin\,\theta_0}{\lambda} - \frac{n}{d}\right)\right)$$

and thus:

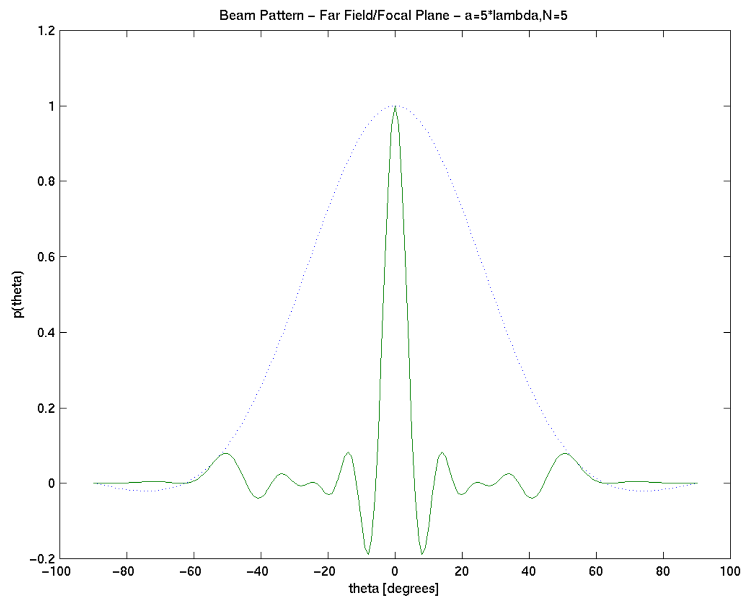
$$p(r,\theta) = K\cos\theta\ \mathrm{sinc}(w\frac{\sin\theta}{\lambda})\cdot\sum_{n=-\infty}^{\infty}\mathrm{sinc}\left(\frac{2a}{\lambda}\left(\sin\theta - \sin\theta_0 - \frac{n}{d}\right)\right)$$

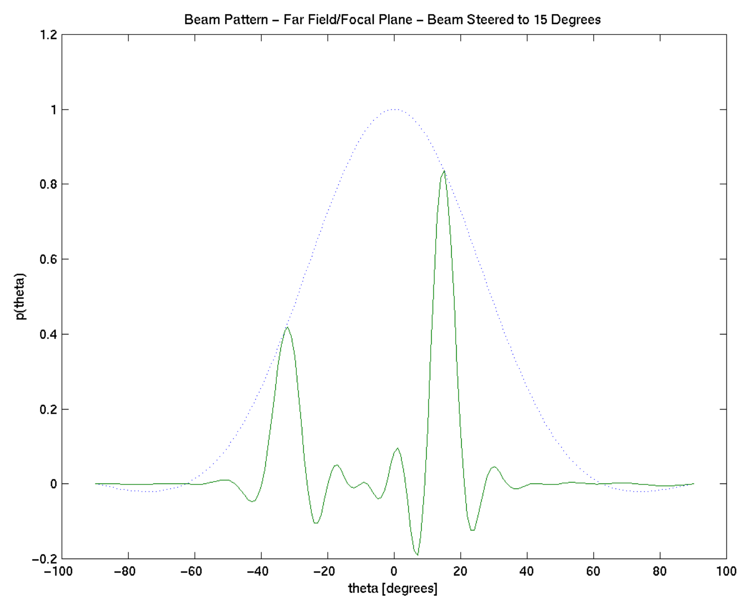
Finally, for a Nyquist sampled array ($d=\lambda/2$):

$$p(r,\theta) = K\cos\theta\,\mathrm{sinc}(w\frac{\sin\theta}{\lambda})\cdot\sum_{n=-\infty}^{\infty}\mathrm{sinc}\big((\mathrm{N.A.})\cdot(\sin\theta - \sin\theta_0 - 2n)\big)$$

The size of the grating lobes, however, is dramatically affected by the numerical aperture. First, let's determine if the main lobe of any of the grating lobes can appear in the field of view. For $-\pi < \theta, \theta_0 < \pi$, there are <u>no</u> cases for which $(\sin\theta - \sin\theta_0) \geq 2$. Thus, the argument of the sinc can never be zero and the main lobe of any of the grating lobes will not appear in the field of view. Side lobes, however, can appear. The value of the argument of the sinc is weighted by the numerical aperture and thus, we are pushed farther out. In general, these sidelobes produce additional ripple in the background.

Here are two cases of arrays for which Nyquist sampling has not been achieved.





Comments:

- Observe that the grating lobes are worse for off-axis beams.
- For Nyquist sampled arrays, the effects of grating lobes is minimal.
- The grating lobes affect both the transmit and receive patterns.

Processing for Delay-Sum Beamforming

- The US pulse is fired once → therefore, we must choose a particular steering and depth of focus and set up the delays at the time of transmission. This is easily done - for focusing at (r_0, θ_0):

$$\tau'_n = \tau'(x_n, r_0, \theta_0) = -\frac{x_n \sin\theta_0}{c} + \frac{x_n^{\ 2}\cos^2\theta_0}{2cr_0}$$

- The receive signal at time t corresponds to depth $r = tc/2$.
    - At each point t we need a new focal depth, r.
    - Usually, the receive angle is the same as the transmit angle, θ_0.
    - We need a different set of delay for each depth (time sample).
    - Samples associated with each delay are selected and summed together as in:

    $$v(t) = \sum_{n=1}^{N} v_n(t + \tau'_n)$$

In practice, the samples are just stored in memory and the right sample is selected.
First, we look at focusing only:

$$\tau_n' = \frac{x_n^{\ 2}}{2cr} = \frac{x_n^{\ 2}}{c^2 t} \approx m\Delta t$$

thus the delay relative to central element ($x_n = 0$) is:

$$m(t) = \mathrm{int}\left[\frac{x_n^{\ 2}}{c^2 t \Delta t}\right]$$

Thus, in order to focus a different depths the receive signals for each transducer elements need to be stored in memory until all depths needing that sample have been used

- How small does Δt need to be? The maximum phase error introduced by the coarseness of the delays (nearest neighbor) is:

$$\Delta\phi = \frac{2\pi f_0 \Delta t}{2}$$

- Typically, a phase error less than 10 degrees ($\frac{2\pi}{36}$ radians) is necessary for good quality beam forming:

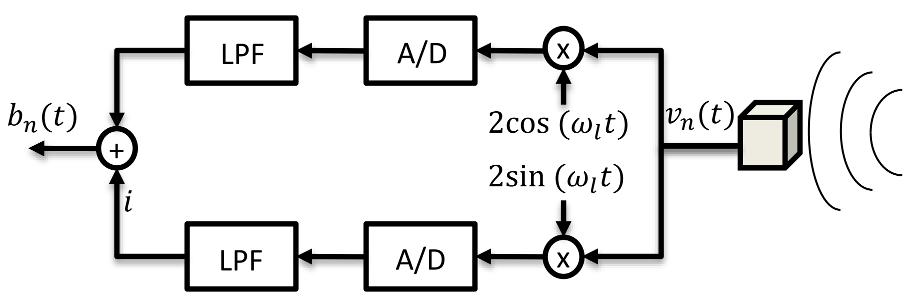
$$\Delta t = \frac{2\pi}{36}\frac{1}{\pi f_0} = \frac{1}{18 f_0}$$

For example, if f_0 = 5 MHz, then $\Delta t \le$ 12 ns (sampling rate = 90 MHz).

- This is a (sort-of) high sampling rate – fast samplers can be expensive and power hungry.
- In addition, sampling synchrony/accuracy end-to-end across the array's many channels is difficult.
- Reduced sampling, also reduced computation

Practical Beam Forming System

While the above could be built with today's technology, there is an eloquent solution to this hardware challenge that can lead to cost savings, greater accuracy and importantly, allows access to the complex signal (as opposed to the real-valued shifted cosines). This solution involves the "quadrature" or complex demodulation to an intermediate frequency (ω_l), which is then sampled and followed by a software phase correction term.



Now, first consider a reflection at some location (r_0,θ), the propagation time will be $\tau_n = \tau(x_n, r, \theta)$ and the analytic signal at transducer n will be $a(t-\tau_n)e^{-i\omega_0(t-\tau_n)}$.

Recall the actual received waveform is a real valued function:

$$v_n(t) = \Re\left\{a(t-\tau_n)e^{-i\omega_0(t-\tau_n)}\right\} = a(t-\tau_n)\cos\omega_0(t-\tau_n)$$

(with some scaling factors dropped). First, let's look at demodulation part of this receiver. There are two channels – the "I" or in-phase channel (upper path in figure) and "Q" or quadrature (lower) channel:

$$I = \mathrm{LPF}\{a(t-\tau_n)\cos\omega_0(t-\tau_n)2\cos\omega_l t\}$$

$$= \mathrm{LPF}\{a(t-\tau_n)[\cos((\omega_0-\omega_l)t-\omega_0\tau_n)+\cos(\omega_0+\omega_l)t]\}$$

$$= a(t-\tau_n)\cos(\Delta\omega t-\omega_0\tau_n)$$

where $\Delta\omega = \omega_0 - \omega_l << \omega_0$ (the difference between the transmit frequency and the local oscillator in the receiver). For the Q channel:

$$= \mathrm{LPF}\{a(t-\tau_n)\cos\omega_0(t-\tau_n)2\sin\omega_l t\}$$
$$= \mathrm{LPF}\{a(t-\tau_n)[\sin((\omega_0-\omega_l)t-\omega_0\tau_n)+\sin(\omega_0+\omega_l)t]\}$$
$$= a(t-\tau_n)\sin(\Delta\omega t-\omega_0\tau_n)$$

Now, we define the baseband signal for element n, $b_n(t)$:

$$b_n(t) = I \mp iQ = a(t-\tau_n)e^{\mp i\Delta\omega t}e^{\pm i\omega_0\tau_n}$$

We include $\mp$ to note that there are two ways to construct the baseband signal – this is equivalent to choosing either the positive or negative sideband during the demodulation process (recall cosine modulation results in both positive and negative sidebands).

To simplify the subsequent analysis, we will let $\omega_l = \omega_0$ or $\Delta\omega = 0$, or:

$$b_n(t) = a(t-\tau_n)e^{\pm i\omega_0\tau_n}$$

- Note that the $e^{\pm i\omega_0\tau_n}$ is a constant phase term not a time varying term. One interesting analogy is that this brings us back to the steady-state approximate that we used in the original derivation of the Frensel zones. Specifically, time delays are now converted into phase factors (recall, we first put phase functions to focus and later derived that this is equivalent to time delays).
- Now, we need to sample $b_n(t)$, but this has phase variations that occur much more slowly. The sampling must occur only fast enough to capture:
    - Amplitude changes in $a(t)$ and not in the carrier ($\omega_0 t$).
- Sampling at $f_s = (2 \text{ to } 4)f_0$ is adequate for good beam forming. Thus we get a 10-fold reduction in the sampling rate and memory requirements.

When constructing the final baseband signal, we need to do 2 things:

1. Shift the envelop functions by discrete sample values - $[\tau_n']$ so that they align. Since $a(t)$ is smoothing varying and we are not relying on it to do constructive or destructive interference, this shift correction can be fairly coarse.
2. Apply a phase term $= -\omega_l\tau_n' = -\omega_0\tau_n'$ (essentially the quadratic and linear phases for focusing and steering). This phase factor insures that the wavefronts will add constructively for the focal point and destructively for wavefronts coming from other directions. Observe that this is a multiplicative factor and doesn't need to discretized in the same way the shifts do. Thus, high accuracy is possible.

The final baseband signal is:

$$\begin{aligned} b(t) &= \frac{1}{N}\sum_{n=1}^{N} b_n\left(t + [\tau_n']\right) e^{\mp i\omega_0\tau_n'} \\ &= \frac{1}{N}\sum_{n=1}^{N} a\left(t - \tau_n + [\tau_n']\right) e^{\pm i\omega_0\tau_n} e^{\mp i\omega_0\tau_n'} \\ &= \frac{1}{N}\sum_{n=1}^{N} a\left(t - \tau_n + [\tau_n']\right) e^{\pm i\omega_0(\tau_n - \tau_n')} \end{aligned}$$

I've included the + and – terms here because you may not always know how your demodulation is done (e.g. in the US project). Now, keeping only the positive term, we make an approximation that $a(t)$ is varies little over Δt and thus:

$$b(t) = \frac{1}{N}\sum_{n=1}^{N} a(t - \tau_n + \tau_n') e^{i\omega_0(\tau_n - \tau_n')}$$

$$\approx \frac{1}{N}\sum_{n=1}^{N} a\left(t - \frac{r_0}{c}\right) e^{i\omega_0\left(\frac{r_0}{c}\right)} = a\left(t - \frac{r_0}{c}\right) e^{i\omega_0\left(\frac{r_0}{c}\right)}$$

Recall that:

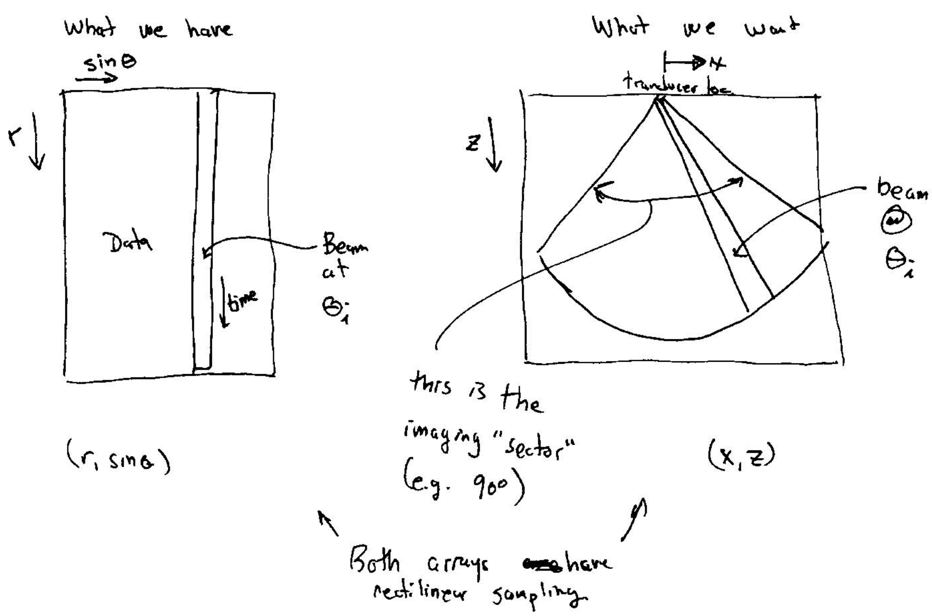
$$\tau_n = \frac{r_0}{c} - \frac{x_n \sin\theta_0}{c} + \frac{x_n^2 \cos^2\theta_0}{2cr_0}$$

$$\tau'_n = -\frac{x_n \sin\theta_0}{c} + \frac{x_n^2 \cos^2\theta_0}{2cr_0}$$, and $[\tau'_n]$ is discretized to $m\Delta t$

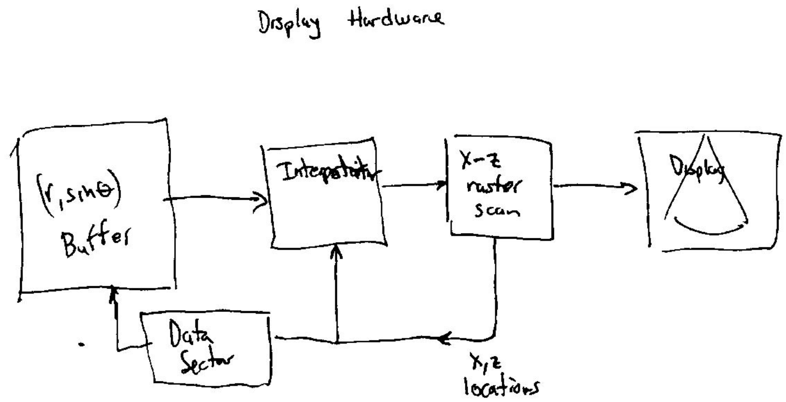
- Observe the additional phase term $e^{-i\omega_l \tau_n'}$ is necessary to bring align the phases before summing.
- The phase correction term is calculated (floating point) is not discretized and is applied to the digitally sampled data and thus can be performed with very high accuracy.
- The final received signal is the proportional to $b(t)$ and the reflectivity at a point at depth, r_0.
- Commonly, data is presented as $|R(tc/2,\theta)\mathrm{b}(t)|$ or $\log|R(tc/2,\theta)\mathrm{b}(t)|$.
- Let's look again at errors – the error in phase due to the coarseness of samples is $\Delta\phi = \Delta\omega\Delta t$ and $\Delta\omega$ can be made arbitrarily small (even 0).
- The other source of error is scalar from poor sampling of $a(t)$, but with a sample rate of $f_s = (2\text{-}4)f_0$ and the pulse length $T=3/f_0$, we insure that 6-12 samples are made over the pulse shape. (The delays align the pulses prior to summation.)

Scan Conversion

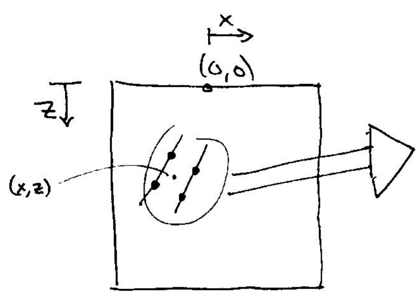
The final process in a US system is scan conversion to present the data as an image in a rectilinear coordinate space to the user. This involves conversion from our beam data, which exists in a domain that is equally spaced in $\sin\theta$ and in r, to physical space, which is a domain that is equally spaced in x and z:

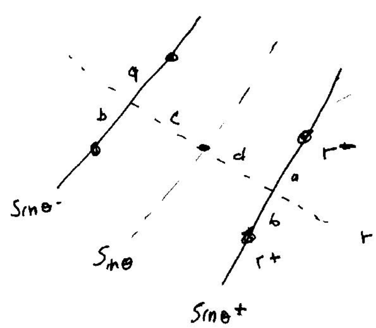


This process is often driven by needs of display memory, which scans its domain (x, z) and requests interpolates from the acquired data in ($\sin\theta$, r) space. In this case, the output coordinates drives the interpolation process – in other words, it is "output-driven." There are approaches to interpolation that are "input driven," but we will not describe those here.



One simple and common approach to interpolation is bilinear interpolation:





Here, we find the 4 nearest neighbors and perform linear interpolation in two directions. First, let's take our output coordinates and give its equivalent in the input space:

$$r = \sqrt{x^2 + z^2} \quad \text{and} \quad \sin\theta = \frac{x}{r}$$

The of four nearest neighbors are determined from:

$$r^+ = \Delta r \cdot \text{ceil}\left(\frac{r}{\Delta r}\right); r^- = \Delta r \cdot \text{floor}\left(\frac{r}{\Delta r}\right);$$

$$\sin\theta^+ = \Delta\sin\theta \cdot \text{ceil}\left(\frac{\sin\theta}{\Delta\sin\theta}\right); \sin\theta^- = \Delta\sin\theta \cdot \text{floor}\left(\frac{\sin\theta}{\Delta\sin\theta}\right)$$

and the data are:

$$P_1 = data(r^-, \sin\theta^-); P_2 = data(r^-, \sin\theta^+); P_3 = data(r^+, \sin\theta^-); P_4 = data(r^+, \sin\theta^+)$$

and distances of several segments are:

$$a = r - r^-; b = r^+ - r; c = \sin\theta - \sin\theta^-; d = \sin\theta^+ - \sin\theta$$

and finally:

$$P(x,z) = \frac{bdP_1 + bcP_2 + adP_3 + acP_4}{\Delta r \Delta \sin\theta}$$

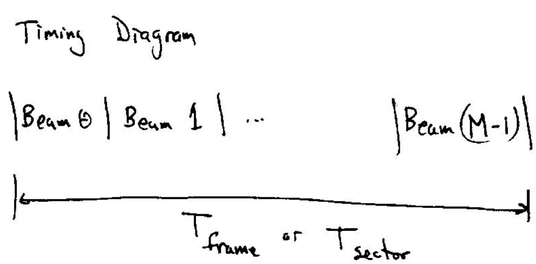
There are, of course, many other (and better) methods for interpolation in a 2D space.

Temporal Resolution

The temporal resolution of an US system is determined by the time required to image an entire sector. Typically only one pulse is traversing the body at any given time. Thus the minimum time between pulses is given by:

$$T = \frac{2r_{\max}}{c}$$

For example, if $r_{\max}$ = 20 cm, then T = 267 μs.



For M beams in a sector, the time for one frame will be $T_{\text{frame}} = TM$. For $r_{\max}$ = 20 cm and M = 128 then $T_{\text{frame}} \approx$ 33 ms. Thus the frame rate will be about 30 Hz.

Overall System Architecture