

---

```

% Program to create array beam plot

clear all
j=sqrt(-1);

vel=1540; % Speed of sound - all units MKS
num_elems=128; % Number of elements
fc = 5e6; % Center frequency
pitch = vel/(fc); % Array element pitch
z_foc = 50e-3; % Range direction focal distance
theta_steer = 0*pi/180; % Steer angle

fs=fc/64; % Define a sampling frequency (not cri
f=[fs:fs:8*fc]; % Define an adequate frequency range
w=2*pi*f; % Angular frequency radians
ns=length(f); % Number of samples

tdel=1.0e-6; % Use a fixed time offset so that base

bw=30; % Fractional bandwidth as percent
sig=bw*fc/100; % Width of Gaussian
gauss_pulse=exp(-pi*((f-fc)/sig).^2); % Generate Gaussian pulse (frequency d

gauss_pulse=gauss_pulse.*exp(-j*w*tdel); % Apply timed delay so 0 for t<0

gauss_t=real(ifft(gauss_pulse)); % Time domain of base waveform for ref
env_gauss_t=abs(hilbert(gauss_t)); % Envelope calculation
tstep=1./max(f); % Time steps after using Inverst FFT
t=[1:ns].*tstep; % Define time axis

weight=ones(num_elems,1); % Define the weighting function (you ca

xres = 0.050;
% Increase resolution for calculation in the x field.
x_pts=[-50:xres:50].*1e-3; % Define X-direction field locations
z_pts=[0.01:2.0:50.01].*1e-3; % Define Z-direction field locations

% Create focal delays
for i=1:num_elems
    x_elem(i)=((i-1)-(num_elems-1)./2).*pitch; % Calculate locations of each arra
    foc_del(i)=(sqrt(x_elem(i).^2+z_foc^2)-z_foc)./vel; % Calculate focusing delay
    steer_del(i) = i*pitch*sin(theta_steer)/vel;
    foc_del(i)=foc_del(i)+steer_del(i); % You can included steering and focusi
end

for j=26:length(z_pts) % Loop over field locations
    for i=1:length(x_pts)
        sum_pulse=zeros(size(gauss_pulse)); % Initialize sum of waveforms to zero
        for k=1:num_elems
            prop_del=(sqrt((x_elem(k)-x_pts(i)).^2+z_pts(j)^2))./vel;
            % prop_del = ??? for each elem location, calculate actual
            % propagation time out to current field point
            sum_pulse=sum_pulse+weight(k).*gauss_pulse.*exp(-sqrt(-1)*w.*(prop_del
        end

        sum_pulse_t=real(ifft(sum_pulse)); % Convert to time domain

```

---

---

```

        field_val(j,i)= max(abs(hilbert(sum_pulse_t)));
    end
    field_val(j,:)=field_val(j,:)./max(field_val(j,:));
    field_db(j,:)=20*log10(field_val(j,:));
end

% Find -6 dB Points
mask = field_db(26,:) > (-6);
mask = mask*1;
six_dB_width = sum(mask)*xres

figure(1)
hold on
plot(x_pts,field_db(26,:), 'k');
xlabel('x');
ylabel('Field Strength dB');
title('Unity-Wavelength Spacing - Field at z = 50 mm');
axis([-0.01 0.01 -40 0])

UltrasoundHomework_Task2(1);
legend('Unity-Wavelength Spacing (f_c = 5 MHz)', 'No Window (f_c = 10 MHz)', 'Hann

beep

six_dB_width =

    4.5000e-001

six_dB_width =

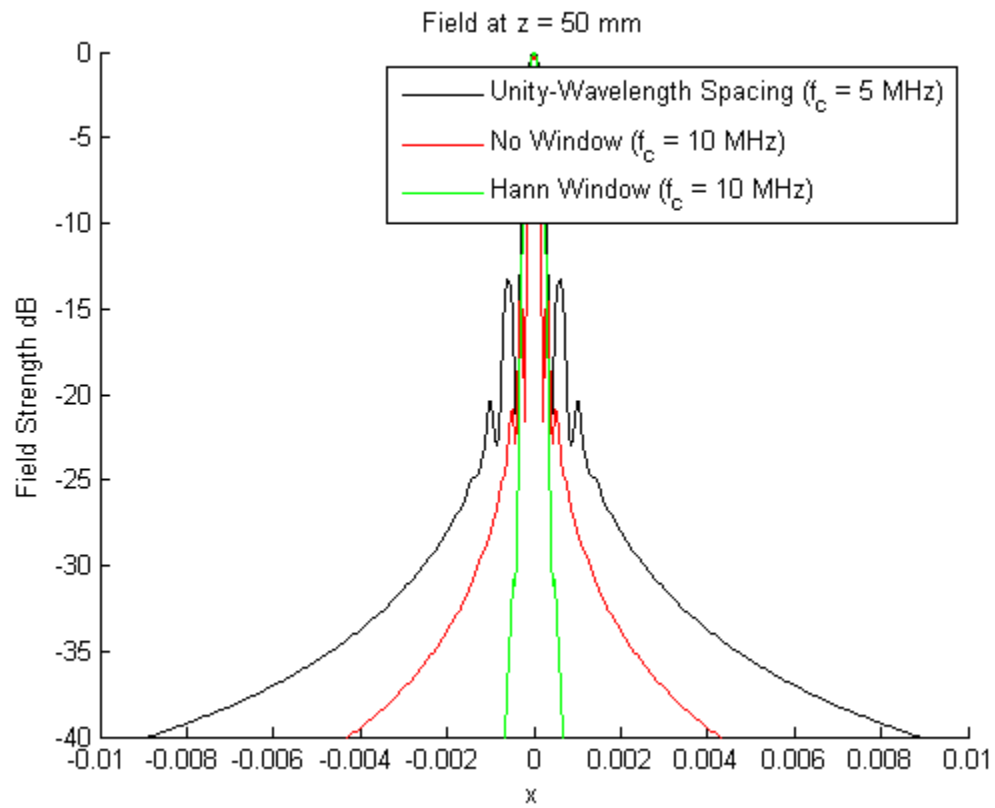
    2.5000e-001

six_dB_width =

    4.5000e-001

```

---



*Published with MATLAB® 7.11*