# Demo: Simulating Fading Channels

MATLAB has many excellent routines for simulating various multi-path fading models. In going through this demo you will learn to:

- Simulate fractional delays using the DSP toolbox
- Describe multi-path channels with arrays of delays and angles
- Plot the time-varying frequency response for a multi-path channel
- Simulate narrowband statistical fading models

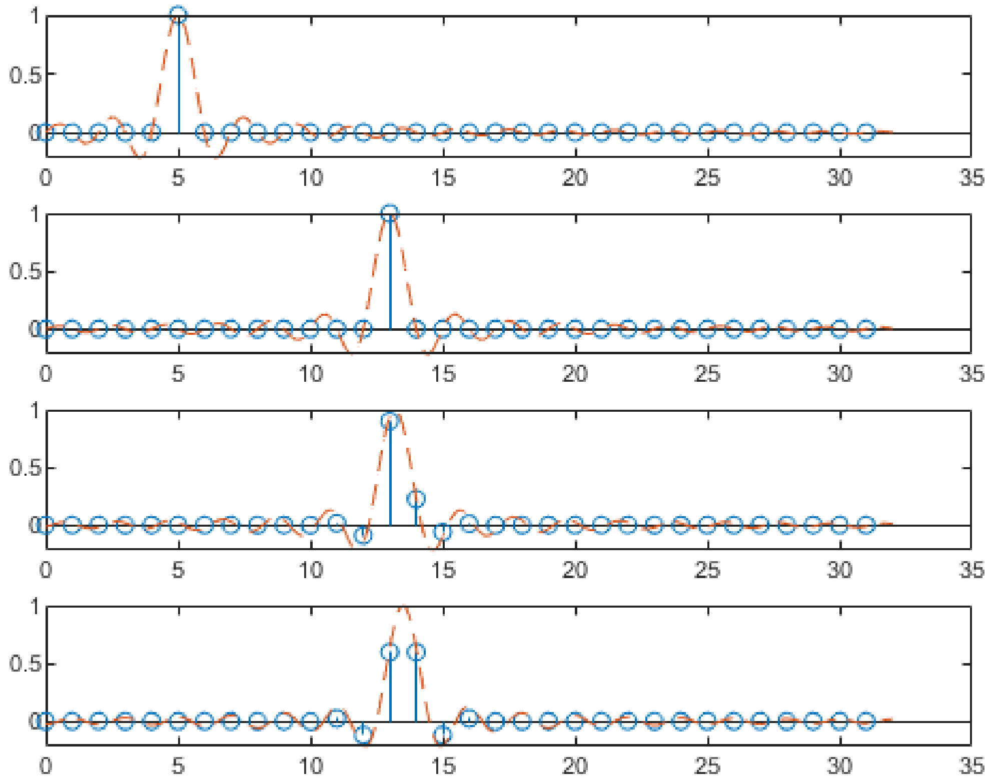
## Contents



## Simulating a fractional delay

Simulating wireless channels requires simulating fractional delays. This can be done easily in MATLAB as follows.

```
tau = [0,8,8.2,8.5];  % Delays in fractions of a sample

% Create a fractional delay object from the DSP toolbox
% We select the Farrow interpolation, which is a fast
% and accurate method.  It is important to select the options
% correctly
dly = dsp.VariableFractionalDelay(...
    'InterpolationMethod', 'Farrow','FilterLength',8,...
    'FarrowSmallDelayAction','Use off-centered kernel');

% Generate a sequence of length nt with an impulse at t0
nt = 32;
t0 = 5;
x = zeros(nt,1);
x(t0+1) = 1;
tx = (0:nt-1)';

% Create delays of the sequence
y = dly.step(x,tau);

% Plot the results along with the theoretical sinc filter
ntau = length(tau);
for i = 1:ntau
    subplot(ntau,1,i);
    stem(tx,y(:,i));
    hold on;
    t = linspace(0,nt,1000)';
    plot(t, sinc(t-tau(i)-t0), '--');
    hold off;
end
```

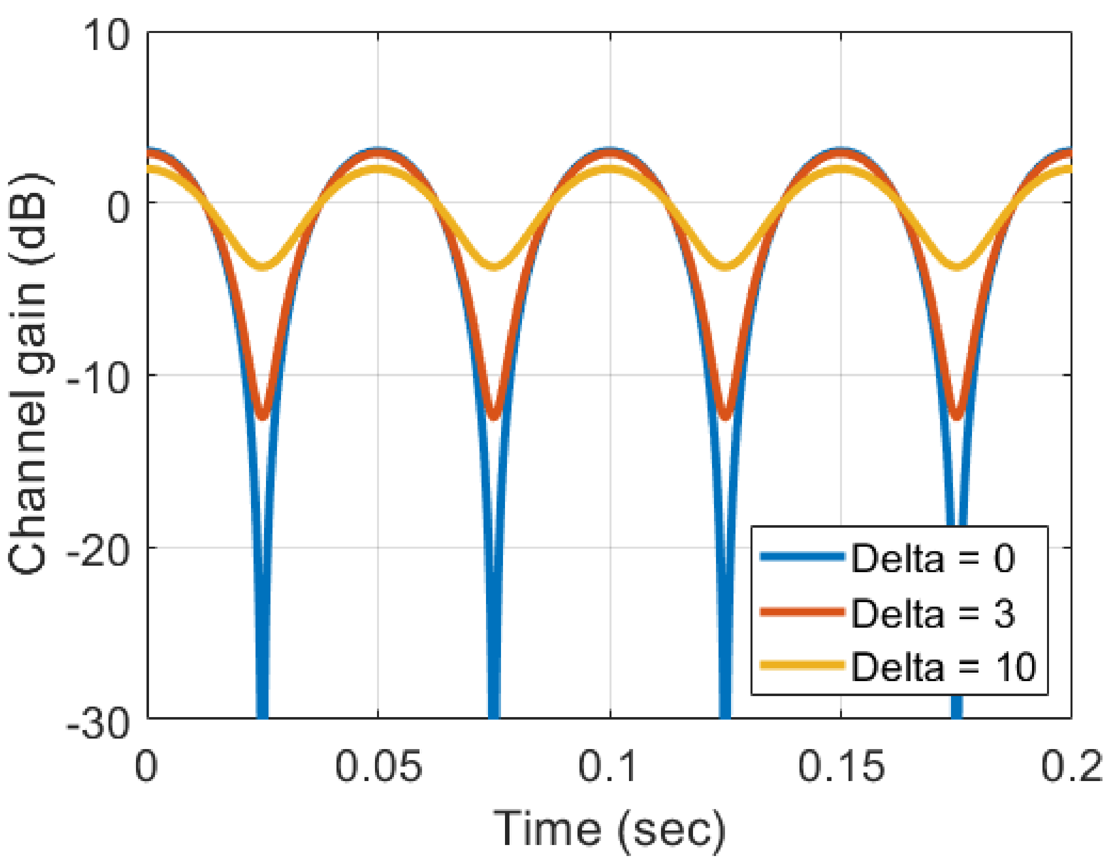



## Two path channel: Variations in Time

To illustrate the concepts of multipath fading, we consider a simple two path channel.

```
% Parameters
dly = [0,0.2]'*1e-6;  % path delays in us
fdmax = 10;          % max doppler shift
theta = [0,pi]';     % angles of the paths
deltaTest = [0,3,10];  % difference in dB between paths

% We first compute the narrowband response over time
% for a fixed frequency
fd = fdmax*cos(theta); % doppler shifts of the paths
ndel = length(deltaTest);

% Times to plot
nt = 1000;
t = linspace(0,0.2,nt)';

% Compute the channel response over time for each delta value
H = zeros(nt,ndel);
legStr = cell(ndel,1);
for i = (1:ndel)

    % Compute path gains
    del = deltaTest(i);
    hpow = [1 10.^(-0.1*del)];
```

```
    hpow = hpow / sum(hpow);
    h = sqrt(hpow)';

    % Compute the channel response
    H(:,i) = exp(2*pi*1i*t*fd')*h;

    % Add to legend
    stri = sprintf('Delta = %d', del);
    legStr{i} = stri;
end

subplot(1,1,1);
P = 10*log10(abs(H).^2);
plot(t,P,'-','LineWidth',3),
grid on,
ylim([-30 10]);
set(gca,'Fontsize', 16);
xlabel('Time (sec)');
ylabel('Channel gain (dB)');
legend(legStr, 'Location', 'SouthEast');
```

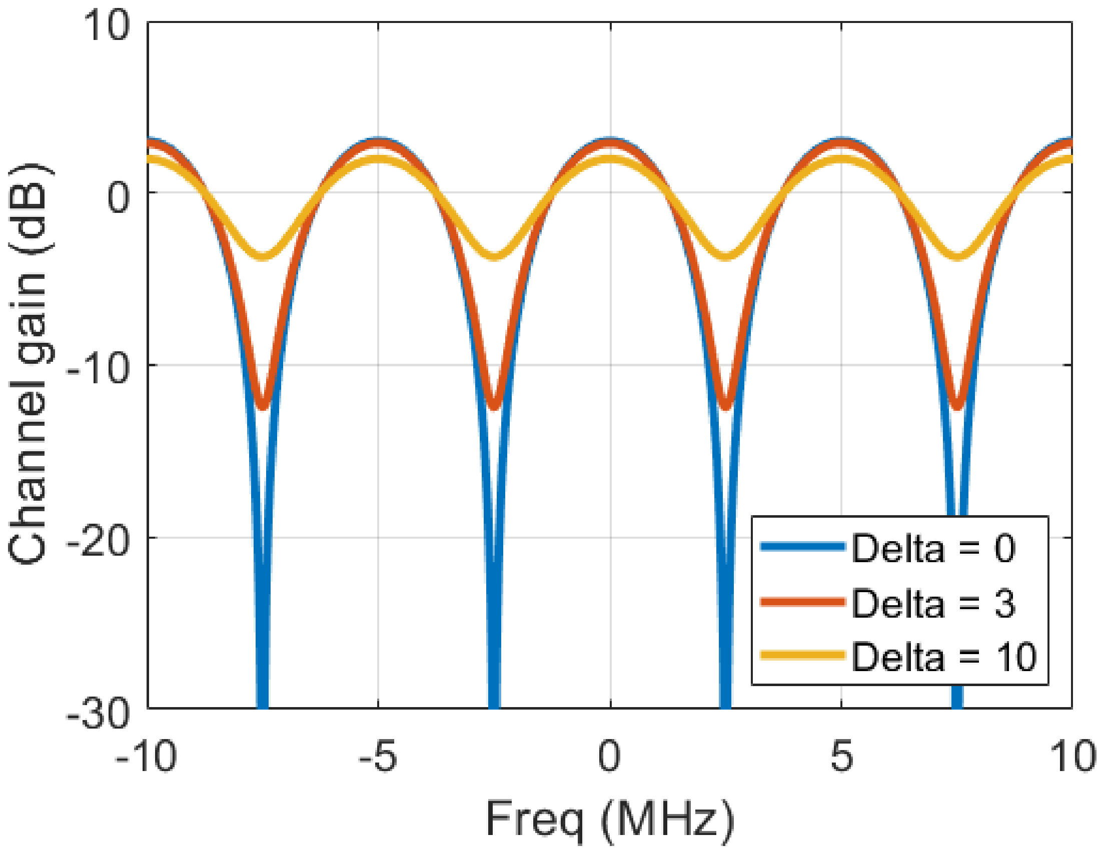



## Two-path channel: Variations in frequency

We can now fix the time and plot the variations in frequency

```
% Frequencies to plot
nf = 1000;
f = linspace(-10,10,nf)'*1e6;
```

```
% Compute the channel response over time for each delta value
H = zeros(nf,ndel);
legStr = cell(ndel,1);
for i = (1:ndel)

    % Compute path gains
    del = deltaTest(i);
    hpow = [1 10.^(-0.1*del)];
    hpow = hpow / sum(hpow);
    h = sqrt(hpow)';

    % Compute the channel response
    H(:,i) = exp(2*pi*1i*f*dly')*h;

    % Add to legend
    stri = sprintf('Delta = %d', del);
    legStr{i} = stri;
end

subplot(1,1,1);
P = 10*log10(abs(H).^2);
plot(f/1e6,P,'-','LineWidth',3),
grid on,
ylim([-30 10]);
set(gca,'Fontsize', 16);
xlabel('Freq (MHz)');
ylabel('Channel gain (dB)');
legend(legStr, 'Location', 'SouthEast');
```



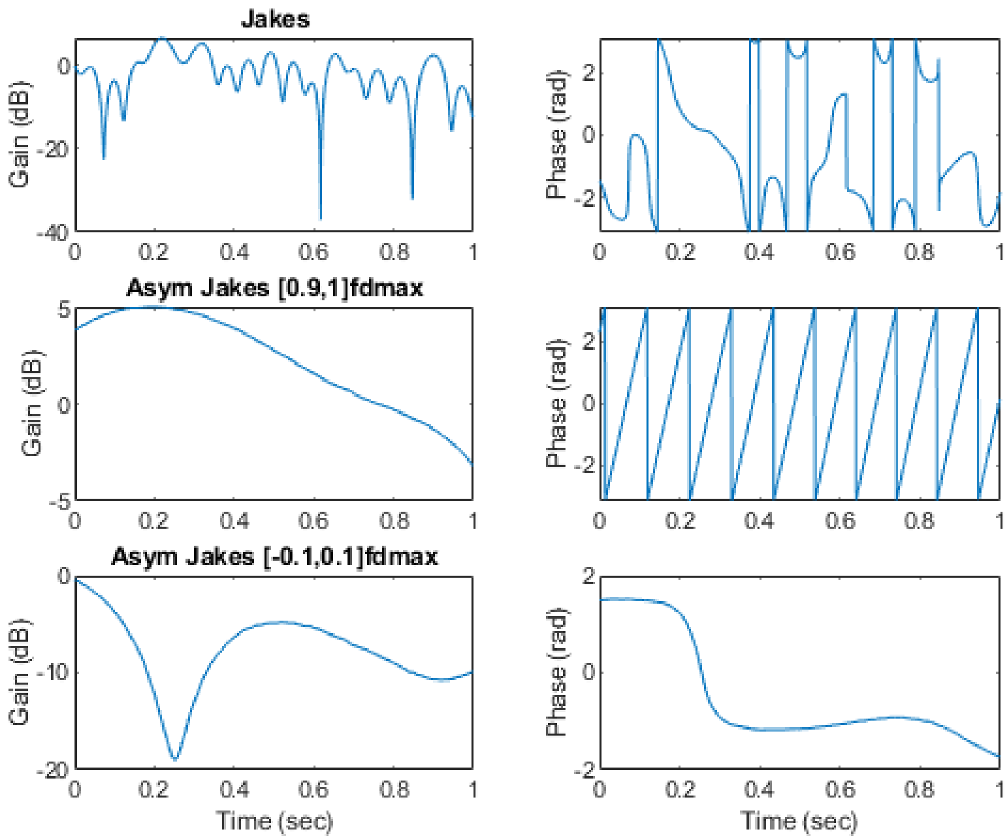

## Simulating Fading

MATLAB has excellent tools to simulate fading channels. The code below plots random samples of fading paths under various fading models. Plotted is the gain magnitude and phase over time. We can observe the following:

- Jake's spectrum has the fastest variations since it contains paths with Doppler from [-fdmax, fdmax]
- The asymmetric Jake's spectrum has paths from [a*fdmax, b*fdmax] where [a,b] are specified in the construction of the Doppler model. In models 2 and 3, [a,b] is a small interval and the Doppler spread is small. As a result, the channel gain changes slowly
- In model 2 and 3, there is a linear phase across time depending on the center Doppler shift. In model 2, the center is close to 0, so the angle does not change. But, in model 3, the angle changes linearly over time

```
% Parameters
fsym = 1e3;      % sample rate in Hz
fdmax = 10;      % max Doppler rate in Hz
nt = 1e3;        % number of samples to simulate

% Create an input sequence
t = (0:nt-1)'/fsym;
x = ones(nt,1);

% Create Doppler models
nmod = 3;
dopMod = cell(nmod,1);
dopMod{1} = doppler('Jakes');
dopMod{2} = doppler('Asymmetric Jakes', [0.9 1]);
dopMod{3} = doppler('Asymmetric Jakes', [-0.1 0.1]);
titleStr = {'Jakes', ...
```

```
            'Asym Jakes [0.9,1]fdmax', ...
            'Asym Jakes [-0.1,0.1]fdmax'};

% Simulate the channel gains for each model
for i = (1:nmod)
    % Create a Rayleigh fading object
    chan = comm.RayleighChannel(...
        'SampleRate', fsym, 'AveragePathGains', 0, ...
        'MaximumDopplerShift', fdmax,...
        'DopplerSpectrum', dopMod{i}, ...
        'PathGainsOutputPort', true);

    % Run the channel
    [y, gain] = chan.step(x);

    % Plot the results
    subplot(nmod,2,2*i-1);
    plot(t, 20*log10(abs(gain)));
    title(titleStr{i});
    if i == nmod
        xlabel('Time (sec)');
    end
    ylabel('Gain (dB)');

    subplot(nmod,2,2*i);
    plot(t, angle(gain));
    if i == nmod
        xlabel('Time (sec)');
    end
    ylabel('Phase (rad)');

end
```




*Published with MATLAB® R2019b*