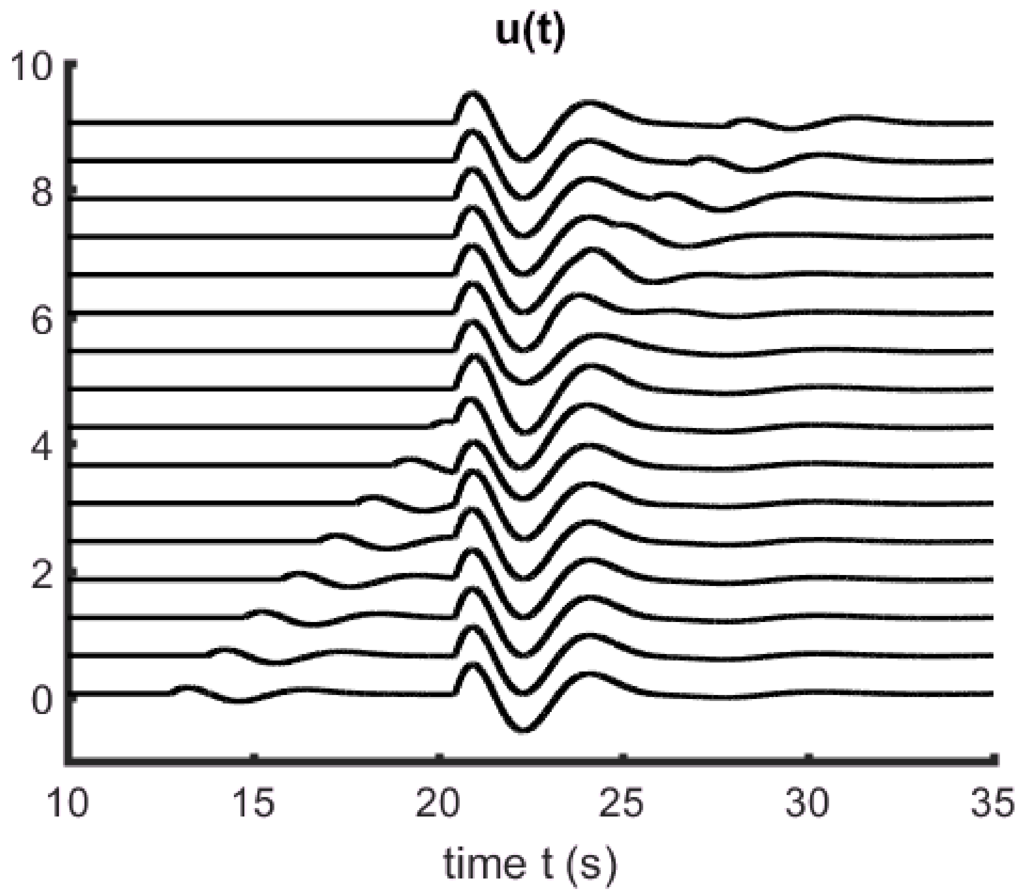
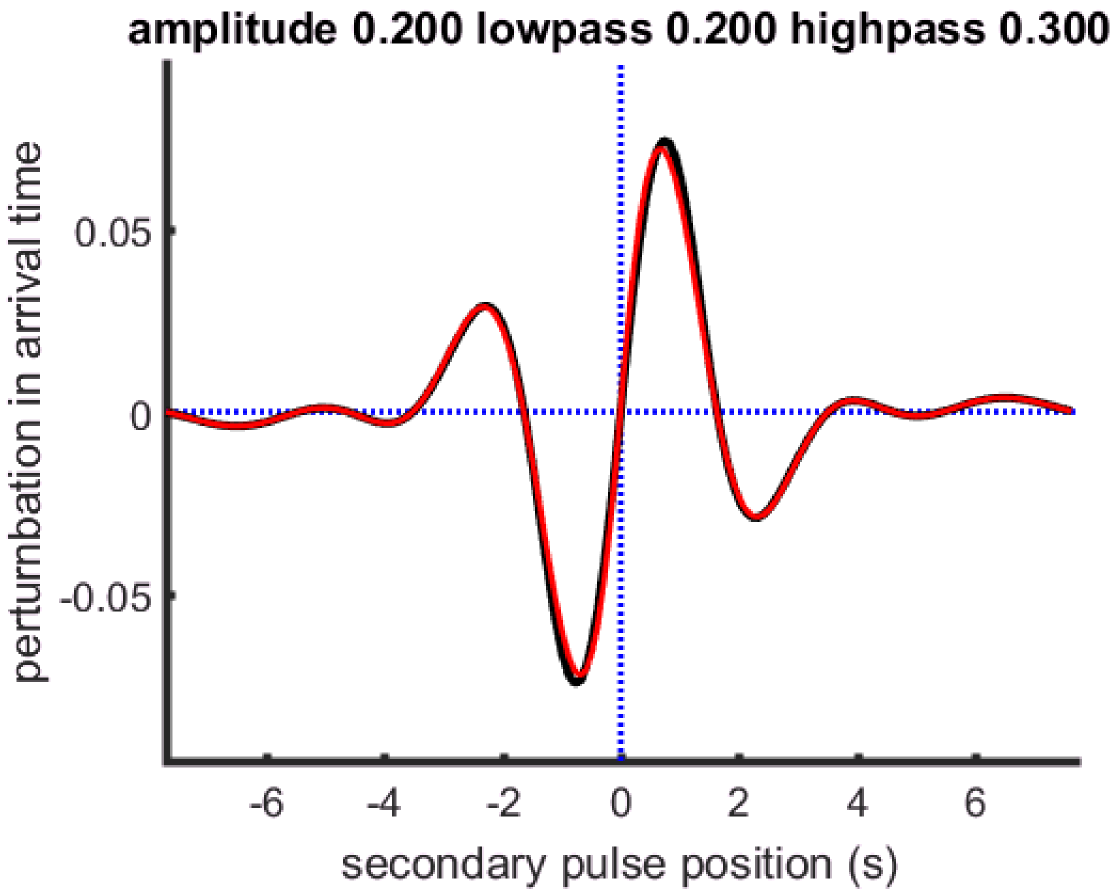
```
% gda13_03
% Supports Figure 13.2
% How much does a small secondary pulse near a main pulse
% affect differential traveltime as determined by cross
% correlation?  Both timeseries are bandpass filtered before
% cross correlation.  Calculation is performed both by brute
% force and Marquering et al (1999) formula

% Note: This script uses a function gda_timedelay() that is
% not described in the text (sorry about that) that does a
% standard cross-correlation to find the delay between two
% signals, and then refines that delay by fitting a parabola
% to the  peak of the cross-correlation and finding the
% peak in the parabola.  This process allows the delay to
% be estimated to a resolution much better than 1Dt (often
% several orders of magnitude better).

% Note: This script uses a function gda_chebyshevfilt() that is
% not described in the text (sorry about that) to bandpass
% filter a pulse.

clear all;

% narrow bandpass frequencies
flow = 0.2;
fhigh = 0.3;

pulsesize = 0.2;  % secondary pulse size (main pulse is 1)

% time setup
Dt = 0.01;
N=4096;
Na = floor(5*N/16); % left limit of secondary pulse position
Nb = floor(11*N/16); % left limit of secondary pulse position
No2 = floor(N/2);
t = Dt*[0:N-1]';
t0 = t(No2);
ta = t(Na);
tb = t(Nb);

% figure 1 is plot of delays
figure(1)
clf;
set(gca,'LineWidth',3);
set(gca,'FontSize',14);
hold on;
scale = 80;
axis( [ta-t0, tb-t0, (ta-t0)/scale, (tb-t0)/scale] );
% reference lines
plot( [0,0]', [(ta-t0)/scale, (tb-t0)/scale]', 'b:', 'LineWidth', 2 );
plot( [ta-t0, tb-t0]', [0,0]', 'b:', 'LineWidth', 2 );
xlabel('secondary pulse position (s)');
ylabel('perturnbation in arrival time');
title(sprintf('amplitude %.3f lowpass %.3f highpass %.3f', pulsesize, flow, fhigh) );

% figure 2 is plot of time series
figure(2)
clf;
set(gca,'LineWidth',3);
```

```
set(gca,'FontSize',14);
hold on;
axis( [10, 35, -1, 10] );
title('u(t)');
xlabel('time t (s)');

% reference signal u0 has just one main pulse
u0 = zeros(N,1);
u0(No2) = 1.0;
u0f = gda_chebyshevfilt(u0, Dt, flow, fhigh); % bandpass filter

ilist = [Na:10:Nb]';   % times at which delay is calculated
ilist2 = [Na:100:Nb]'; % times at which time series is plotted
Nlist = length(ilist);
tpulse = zeros(Nlist,1); % save calculation in array
tau = zeros(Nlist,1);
tau2 = zeros(Nlist,1);

j=0; % counts iterations
for i=ilist' % loop over pulse positions
    % pulse position
    j=j+1;
    tpulse(j)=Dt*(i-No2);

    % another signal u has two pulses
    du = zeros(N,1);
    du(i) = pulsesize;
    duf = gda_chebyshevfilt(du, Dt, flow, fhigh); % band pass filtered perturbation
    u = u0+du;
    uf = u0f + duf; % bandpass filtered signal

    if( ~isempty(find(ilist2==i,1)) )
        figure(2);
        plot( t, 100*uf+0.06*j, 'k-', 'LineWidth', 2 );
    end

    % brute force calculation of the delay by
    % cross-correlation followed by parabolic refinement
    % see the comments in gda_timedelay() for details.
    [ tBmA_est, Cmax_est ] = gda_timedelay( u0f, uf, Dt );
    tau(j) = tBmA_est;
    cmax = Cmax_est;

    % formula from Marquering et al, GJI 137, 805-815, 1999
    u0fd = [diff(u0f); 0]/Dt;  % 1st derivative
    u0fdd = [0; diff(u0f,2); 0]/(Dt^2);  % 2nd derivative
    tau2(j) = (Dt*u0fd'*duf)/(Dt*u0fdd'*u0f);

end
```




```
% Figure 13.2A. Finite frequency travel times. A suite of signals u(t) (black
% curves) are created by adding a small perturbation (blue) to a reference pulse
% u_0 (red), where the signals differ only be the delay t_0 of the perturbation.

% plot time lag against position of secondary pulse
figure(1);
plot( tpulse, tau, 'k-', 'LineWidth', 3);
plot( tpulse, tau2, 'r-', 'LineWidth', 2);
```




```
% Figure 13.2B. Cross-correlation (black curve) and the first order approximation to it
% (Equation (13.??), red curve) are used to estimate the travel time perturbation δτ.
% Both yield similar results. Owing to wave interference, δτ(t_0 ) behaves like
% δ τ ∝ t_0 only for very small delays.  It is oscillatory at larger delays.
% MatLab script gda13_03.m.
```