Pulse Response As A Free Response

CEE 541. Structural Dynamics
Department of Civil and Environmental Engineering
Duke University
Henri P. Gavin
Fall, 2014

1 A pulse “wavelet” equation

The response of a linear time-invariant (LTI) dynamic system to a pulse with a characteristic pulse period, pulse amplitude, and having a number of cycles of pulse motion, may be computed as the free response of two cascaded LTI systems. In this presentation the input will be represented as an acceleration, $\ddot{x}_g$, which could be applied to the base of a viscously-damped elastic oscillator. Such a pulse may be represented by a function

$$\ddot{x}_g(t; T, \tau, n) = \left(\frac{t}{\tau}\right)^n \exp\left[-\frac{t}{\tau}\right] \cos\left(2\pi\frac{t}{T} - \phi\right),$$

where the parameters are the envelope decay time $\tau$, a positive exponent $n$ that governs the rise-time, and the period of oscillation $T$. For a given value of the exponent, $n$, the number of cycles in the pulse is roughly $4\tau/T$. Equation (1) may be written in terms of the sum of a pair of complex-conjugate exponentials

$$\ddot{x}_g(t; T, \tau, n) = \left(\frac{t}{\tau}\right)^n \exp\left[-\frac{t}{\tau}\right] \cos\left(2\pi\frac{t}{T} - \phi\right)$$

$$= \left(\frac{t}{\tau}\right)^n e^{-t/\tau} \left(\cos(\omega t) \cos(\phi) + \sin(\omega t) \sin(\phi)\right)$$

$$= \left(\frac{t}{\tau}\right)^n e^{-t/\tau} \left(a \cos(\omega t) + b \sin(\omega t)\right)$$

$$= \left(\frac{t}{\tau}\right)^n \left(X_g e^{i\omega t} + X_g^* e^{-i\omega t}\right)$$

where $\omega = 2\pi/T$, $a = \cos \phi$, $b = \sin \phi$, $X_g = (a - ib)/2$, and $\lambda = -1/\tau + i\omega$.

2 Zero terminal velocity

A realistic base acceleration pulse will have a corresponding velocity, $\dot{x}_g(t)$, that is zero for $t \gg \tau$. This pulse equation can be made to satisfy the zero final velocity condition by adjusting the phase angle, $\phi$, and it will be shown that the phase angle that satisfies the zero terminal velocity condition depends only on $\tau/T$.

Integrating the two terms in equation (6), for $n = 2$,

$$\frac{X_g}{\tau^2} \int_0^{t_f} t^2 e^{\lambda t} dt = \frac{X_g}{\tau^2} \left(\frac{1}{\lambda} (t_f^2 - 2t_f/\lambda + 2/\lambda^2) e^{\lambda t_f} - \frac{2}{\lambda^2}\right)$$

$$\frac{X_g^*}{\tau^2} \int_0^{t_f} t^2 e^{\lambda^* t} dt = \frac{X_g^*}{\tau^2} \left(\frac{1}{\lambda^*} (t_f^2 - 2t_f/\lambda^* + 2/\lambda^* + 2) e^{\lambda^* t_f} - \frac{2}{\lambda^{*2}}\right)$$
summing, and taking the limit as \( t_f \to \infty \), and setting this final velocity to zero,

\[
\lim_{t_f \to \infty} \left[ \frac{X_g}{\tau^2} \int_0^{t_f} t^2 e^{\lambda t} dt + \frac{X^*_g}{\tau^2} \int_0^{t_f} t^2 e^{\lambda^* t} dt \right] = -\frac{2}{\tau^2} \left( \frac{X_g}{\lambda^3} + \frac{X^*_g}{\lambda^*_3} \right) = 0 \quad (9)
\]

leads to

\[
\frac{X_g}{X^*_g} + \left( \frac{\lambda}{\lambda^*} \right)^3 = 0 \quad (10)
\]

\[
\frac{a - ib}{a + ib} + \left( \frac{-1/\tau + i\omega}{-1/\tau - i\omega} \right)^3 = 0 \quad (11)
\]

\[
\frac{1 - i \tan \phi}{1 + i \tan \phi} + \left( \frac{-1 + i2\pi \tau/T}{-1 - i2\pi \tau/T} \right)^3 = 0, \quad (12)
\]

Taking another approach, equation (9) leads to

\[
\Re \left( \frac{X_g}{\lambda^3} \right) = 0 \quad (13)
\]

\[
\Re \left( \frac{\tau^3 (\cos \phi - i \sin \phi)}{(i\omega \tau - 1)^3} \right) = 0 \quad (14)
\]

\[
\frac{\tau^3}{(\omega^2 \tau^2 + 1)^3} \left( \cos \phi (3\omega^2 \tau^2 - 1) - \sin \phi (3\omega \tau - \omega^3 \tau^3) \right) = 0 \quad (15)
\]

\[
\tan \phi = \frac{3(2\pi \tau/T)^2 - 1}{3(2\pi \tau/T) - (2\pi \tau/T)^3} \quad (16)
\]

So, the condition that \( \dot{x}_g(t_f) = 0 \) for \( t_f \gg \tau \) implies a functional relationship between the phase shift, \( \phi \), and the number of cycles in the pulse, \((4\tau/T)\). The solution, equation (16) may be checked by computing the phase angles that minimize the absolute terminal velocity for a range of values for \( \tau/T \), and comparing to the values of \( \phi \) from equation (16).

### 3 Peak velocity amplitude

The peak velocity amplitude of the pulse scales linearly with \( T \) and is a function of \( \tau/T \). The peak velocity can be approximated as:

\[
V_{\text{max}} = T \max \left[ 10.00(\tau/T)^2 \exp(-7.4\tau/T), \right.
\]

\[
2.02(\tau/T)^2 \exp(-3.5\tau/T), \quad 0.93(\tau/T)^2 \exp(-2.4\tau/T), 0.085 \right] \quad (17)
\]

For \( \tau/T > 1 \) use \( V_{\text{max}} = 0.085T \). The spectrum of peak velocities, as a function of \( \tau/T \) along with the approximation of equation (17) are plotted in figure The two models are virtually identical.
Pulse Response as a Free Response

4 Linear Time-Invariant Model

If the exponent $n$ is an integer, the acceleration pulse may be modeled as the output of a finite dimensional LTI. Equation (6) separates the pulse expression into a sum of complex conjugates. Repeatedly differentiating the first term in the sum, for $n = 2$, and denoting $u(t) = \ddot{x}_g(t)$,

\begin{align*}
u(t) &= (t/\tau)^2 e^{\lambda t} \\
\dot{u}(t) &= (t/\tau)^2 \lambda e^{\lambda t} + 2t/\tau^2 e^{\lambda t} \\
\ddot{u}(t) &= (t/\tau)^2 \lambda^2 e^{\lambda t} + 4t/\tau^2 \lambda e^{\lambda t} + 2/\tau^2 e^{\lambda t} \\
\dddot{u}(t) &= (t/\tau)^2 \lambda^3 e^{\lambda t} + 6t/\tau^2 \lambda^2 e^{\lambda t} + 6/\tau^2 \lambda e^{\lambda t}
\end{align*}

The highest order derivative can be expressed in terms of the lower order derivatives.

\begin{align*}
\dddot{u} - 3\lambda \ddot{u} &= -2(t/\tau)^2 \lambda^3 e^{\lambda t} - 6t/\tau^2 \lambda^2 e^{\lambda t} \\
\dddot{u} - 3\lambda \ddot{u} + 3\lambda^2 \dot{u} &= (t/\tau)^2 \lambda^2 e^{\lambda t} \\
\dddot{u} - 3\lambda \ddot{u} + 3\lambda^2 \dot{u} - \lambda^3 u &= 0
\end{align*}

Note that these expressions give only one part of the complex conjugate pair, $u = u_r + iu_i$, and recall $\lambda = -1/\tau + i\omega$. Substituting into the terms of equation (24),

\begin{align*}
\lambda \dddot{u} &= (-1/\tau + i\omega)(\dddot{u}_r + i\dddot{u}_i) \\
&= -1/\tau \dddot{u}_r + i\omega \dddot{u}_r - i/\tau \dddot{u}_i - \omega i \dddot{u}_i \\
\lambda^2 \dddot{u} &= (-1/\tau + i\omega)(-1/\tau + i\omega)(\dddot{u}_r + i\dddot{u}_i) \\
&= (1/\tau^2 - 2i\omega/\tau - \omega^2)(\dddot{u}_r + i\dddot{u}_i) \\
&= 1/\tau^2 \dddot{u}_r - 2i\omega/\tau \dddot{u}_r - \omega^2 \dddot{u}_r + i/\tau^2 \dddot{u}_i + 2\omega/\tau \dddot{u}_i - i\omega^2 \dddot{u}_i \\
\lambda^3 u &= (1/\tau^2 - 2i\omega/\tau - \omega^2)(-1/\tau + i\omega)(u_r + iu_i) \\
&= (-1/\tau^3 + 3i\omega/\tau^2 + 3\omega^2/\tau - i\omega^3)(u_r + iu_i) \\
&= -1/\tau^3 u_r + 3i\omega/\tau^2 u_r + 3\omega^2/\tau u_r - i\omega^3 u_r - i/\tau^3 u_i - 3\omega/\tau^2 u_i + 3i\omega^2/\tau u_i + \omega^3 u_i
\end{align*}
Now, separating the complex-valued o.d.e. of equation (24) into two real valued o.d.e’s for $u_r$ and $u_i$,

$$\ddot{u}_r + 3\omega\dot{u}_r + 3/\tau\ddot{u}_i + 6\omega/\tau\dot{u}_i + 3(1/\tau^2 - \omega^2)\dot{u}_r + (3\omega/\tau^2 - \omega^3)u_i + (1/\tau^3 - 3\omega^2/\tau)u_i = 0 \quad (28)$$

$$\ddot{u}_i - 3\omega\ddot{u}_r + 3/\tau\ddot{u}_i - 6\omega/\tau\dot{u}_r + 3(1/\tau^2 - \omega^2)\dot{u}_r - (3\omega/\tau^2 - \omega^3)u_r + (1/\tau^3 - 3\omega^2/\tau)u_i = 0 \quad (29)$$

These coupled third order o.d.e’s may be put into state-space form as follows:

$$\frac{d}{dt} \begin{bmatrix} u_r \\ u_i \\ \dot{u}_r \\ \dot{u}_i \\ \ddot{u}_r \\ \ddot{u}_i \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -(1/\tau^3 - 3\omega^2/\tau) & -(3\omega/\tau^2 - \omega^3) & -3(1/\tau^2 - \omega^2) & -6\omega/\tau & -3/\tau & -3\omega \\ -(1/\tau^3 - 3\omega^2/\tau) & -(3\omega/\tau^2 - \omega^3) & -3(1/\tau^2 - \omega^2) & 6\omega/\tau & -3/\tau & -3\omega \end{bmatrix} \begin{bmatrix} u_r \\ u_i \\ \dot{u}_r \\ \dot{u}_i \\ \ddot{u}_r \\ \ddot{u}_i \end{bmatrix} \quad (30)$$

with initial conditions,

$$\begin{bmatrix} u_r \\ u_i \\ \dot{u}_r \\ \dot{u}_i \\ \ddot{u}_r \\ \ddot{u}_i \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 2/\tau^2 \\ 0 \end{bmatrix} \quad (31)$$

and the total solution is given by $\ddot{x}_g(t) = X_g u + X_g^* u^* = \cos(\phi)u_r + \sin(\phi)u_i$,}

$$\ddot{x}_g(t) = \begin{bmatrix} \cos(\phi) & \sin(\phi) & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_r \\ u_i \\ \dot{u}_r \\ \dot{u}_i \\ \ddot{u}_r \\ \ddot{u}_i \end{bmatrix}. \quad (32)$$

As a comparison of the closed form model of equation (6) and of the state-variable model of equations (30) - (32), pulse acceleration records and corresponding velocity records are plotted on the same axes in Figure 3 The two models are virtually identical.
Given a state variable model that computes a desired pulse function as a free response, the response of a dynamical system to such a pulse may also be computed as a free response. The pulse dynamics are described by a state variable system \(\dot{u}(t) = A_u u(t)\) with an initial state \(u(0) = u_0\) and the output \(y_u(t) = C_u u(t)\). To compute the response of a primary system to such a pulse, the output of the pulse system is the input to the primary system, \(\dot{x}(t) = Ax(t) + By_u(t) = Ax(t) + BC_u u(t)\), giving the cascade system dynamics,

\[
\frac{d}{dt} \begin{bmatrix} u(t) \\ x(t) \end{bmatrix} = \begin{bmatrix} A_u & 0 \\ BC_u & A \end{bmatrix} \begin{bmatrix} u(t) \\ x(t) \end{bmatrix}, \quad \begin{bmatrix} u(0) \\ x(0) \end{bmatrix} = \begin{bmatrix} u_0 \\ 0 \end{bmatrix}
\]

(33)

from which any response of the primary system \(y(t) = Cx(t)\) is

\[
y(t) = C_v e^{A_v t} v_0
\]

(34)

where \(v^T = [u^T \ x^T]\), \(C_v = [0 \ C]\), and

\[
A_v = \begin{bmatrix} A_u & 0 \\ BC_u & A \end{bmatrix}.
\]

(35)

5 Sigmoidal Baseline Correction
function [accel, veloc, displ] = pulseVx ( Vp, Tp, Nc, t, t0, fig )

% Computes an earthquake-like acceleration, velocity, and displacement pulse

% INPUT
% Vp  - max velocity of pulse  1.0
% Tp  - time period of pulse  1.0
% Nc  - number of cycles in pulse ... approximate  1.0
% t   - time record ... [1:points] * delta_t  [1:1000]*0.01
% t0  - time at which the pulse starts  0.25
% fig - figure number for plotting, 0: no plots  1

% OUTPUT
% accel - earthquake ground acceleration
% veloc - earthquake ground velocity
% displ - earthquake ground displacement

if nargin < 1, Vp = 1.0; end
if nargin < 2, Tp = 1.0; end
if nargin < 3, Nc = 1.0; end
if nargin < 4, t = [1:1000]*0.01; end
if nargin < 5, t0 = 0.25 ; end
if nargin < 6, fig = 1; end

points = length(t); % number of points
delta_t = t(2) - t(1); % time step interval
p0 = floor((t0 -t(1))/ delta_t); % number of initial points w/o motion
tau = Nc*Tp/4.0 % decay time constant of pulse
tT = tau/Tp; % Nc / 4
n = 2.0; % rise-time value for pulse

% set phase of pulse such that terminal velocity is zero, or close to it.
phi = atan2( (3*(2*pi*tT)^2 -1) , (3*2*pi*tT - (2*pi*tT)^3) )
accel = ((t-t0)/tau).^n .* exp((-t-t0)/tau) .* cos(2*pi*(t-t0)/Tp - phi);
accel(1:p0) = 0; % shift the offset

% baseline correction for zero terminal velocity and displacement
z = 0.5; % correction over duration z
ts = (t-n*tau-t0)/(z*tau); % scaled time
exp_ts = exp(-ts);
dc = 1./(1+exp_ts); % sigmoidal displacement correction
vc = exp_ts .* (1+exp_ts).*(-2.0)/(z*tau); % d(dc)/dt
ac = exp_ts .* (1+exp_ts).*(-3.0).* (exp_ts -1)/(z*tau)^2; % d(vc)/dt

maxIter = 50;
velT = zeros(maxIter,1);
dspT = zeros(maxIter,1);
for iter = 1:maxIter
velT(iter) = sum(accel)*delta_t;
dspT(iter) = [points:-1:1]*accel'*delta_t^2;
accel = accel - velT(iter)*vc - dspT(iter)*ac;
if abs(dspT(iter)) < 1e-3*Tp )
break;
end
end
iter
veloc = cumtrapz(accel)*delta_t; % trapezoidal rule for the pulse veloc.

%veloc(find(t>0+12*tau)) = 0.0; % knock-out round-off error

displ = cumtrapz(veloc)*delta_t; % trapezoidal rule for the pulse displ.

if Nc > 5,
    maxV = 0.086 * Tp;
else
    maxV = max([ 4.063*Nc^-2.165*exp(-4.403/Nc) ;
                  2.329*Nc^-1.336*exp(-5.693/Nc) ]) * Tp;
end

scale = Vp / maxV;

accel = accel * scale;
veloc = veloc * scale;
displ = displ * scale;

if fig > 0 % plot the pulse data

    figure(fig)
    subplot(3,1,1)
    plot(t,accel,t, scale*dspT*ac);
    axis tight
    ylabel('accel, cm/s^2')
    grid on
    subplot(3,1,2)
    plot(t,veloc,t,vc)
    axis tight
    ylabel('veloc, cm/s')
    grid on
    subplot(3,1,3)
    plot(t,displ,t,scale*dspT*dc);
    axis tight
    ylabel('displ, cm')
    xlabel('time, s')
    grid on

    drawnow

end

% pulseVx 2 Oct 2010; 18 Nov 2010; 24 May 2011; 19 Dec 2011