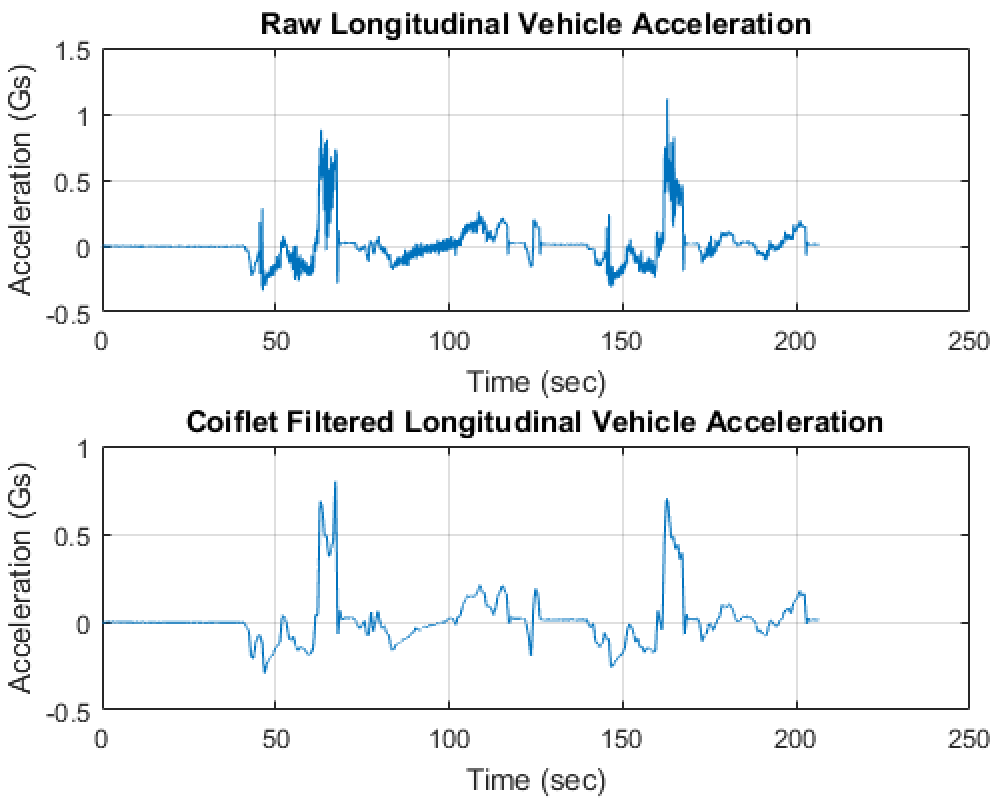
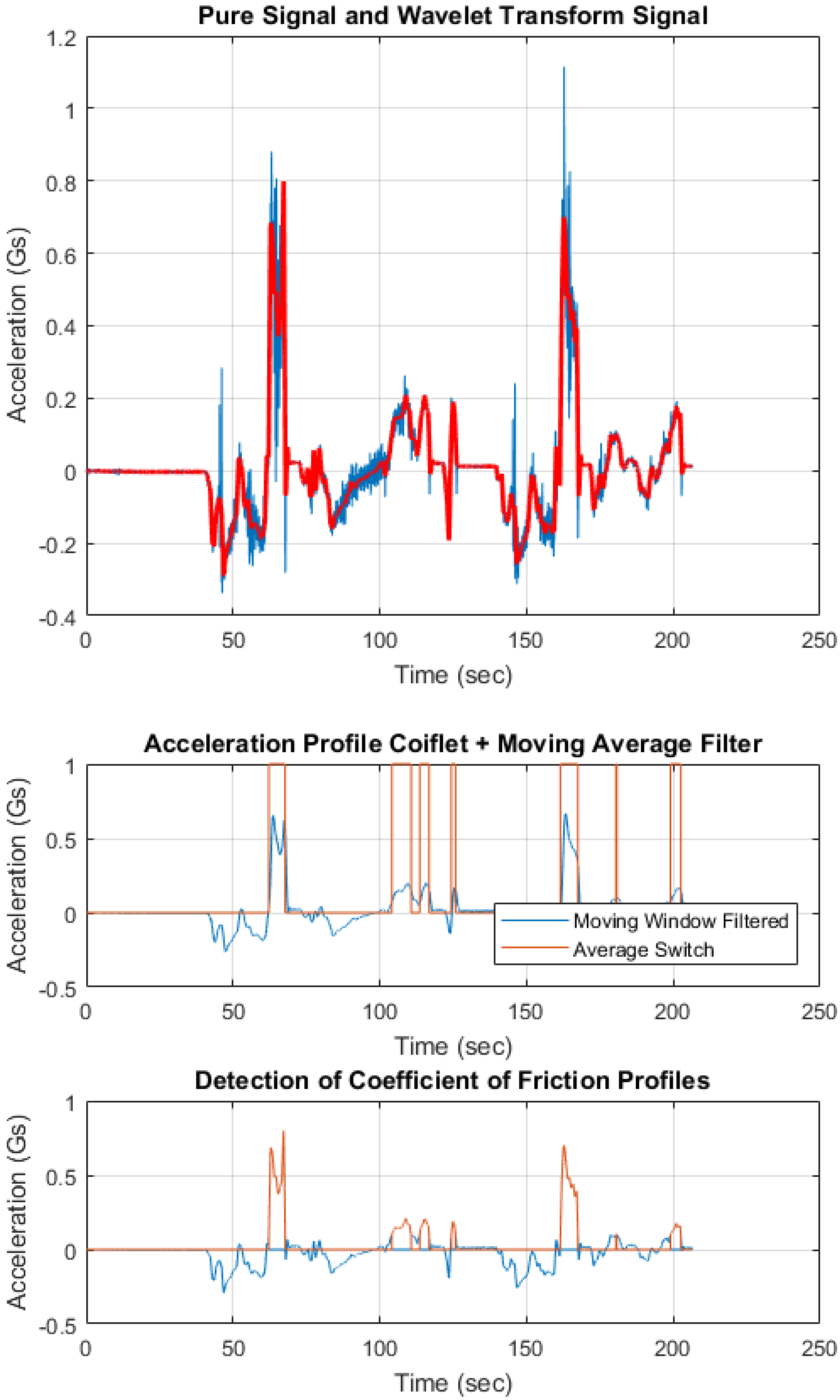
```matlab
clc; clear all; close all
Thrs = 0.1; WdwSize = 100;
% ----------------------------------------
% Full Gravel Testing FGA et FGT
load('ApproxCoif2L6CompleteSignalFGA.mat');
load('FGT.mat'); load('FGA.mat')
t = FullGravelTime;  sig = DecomCoif2L6CompFGA;
a  = FullGravelAccel;
CoifPlotter(t,sig);
[FricCoeff] = FilterSwitch(Thrs,WdwSize,sig,t);
% ----------------------------------------
% Butterworth Filter
% Attempt to Remove Noise to Avoid
% Drifting of Speed Integrated Signal
fc = 0.5;
[ba] = FFTButter(t,a,fc);
% ----------------------------------------
% Splined Signal from Either Coiflet +
% Butterworth filter or original Coiflet Signal
% To Activate Spline, Uncomment Section Below
yy = a; tq = t;
% tq = 0:.1:t(end);
% % Choose to Spline True Signal or Butterworth Filtered Signal
% % yy = spline(t,sig(1,:),tq);
% yy = spline(t,ba,tq);
% figure; plot(tq,yy)
% xlabel('Time (sec)'); ylabel('Acceleration m/s2');
% title('Splined Butterworth Filtered Signal')
% ----------------------------------------
% Note that the splined was obtained from
% the Coiflet filtered Signal (not necessary). Thus
% Simpsons Rule gives an exact interpolation of
% the Splined signal. At this point,
% the drifted signal is attributed completely to
% noise in the data before interpolating.
% ----------------------------------------
[v,d,n] = DiscIntegSims(tq,yy*9.81);
figure; subplot(211); plot(tq,v); grid on;
title('Velocity Profile')
xlabel('Time (sec)'); ylabel('Velocity m/s');
subplot(212); plot(tq,d); grid on;
title('Displacement Profile')
xlabel('Time (sec)'); ylabel('Distance m')
% ----------------------------------------
% From Literature,
% Gravel Friction Coeff Value = 0.55
% Select Appropiate Ranges for Braking Incident
% First Event
idx1 = find(t ==61); idx2 = find(t ==70);
% Second Event
idx3 = find(t ==160); idx4 = find(t ==176);
% The following inputs, the Range of Braking Incident
```

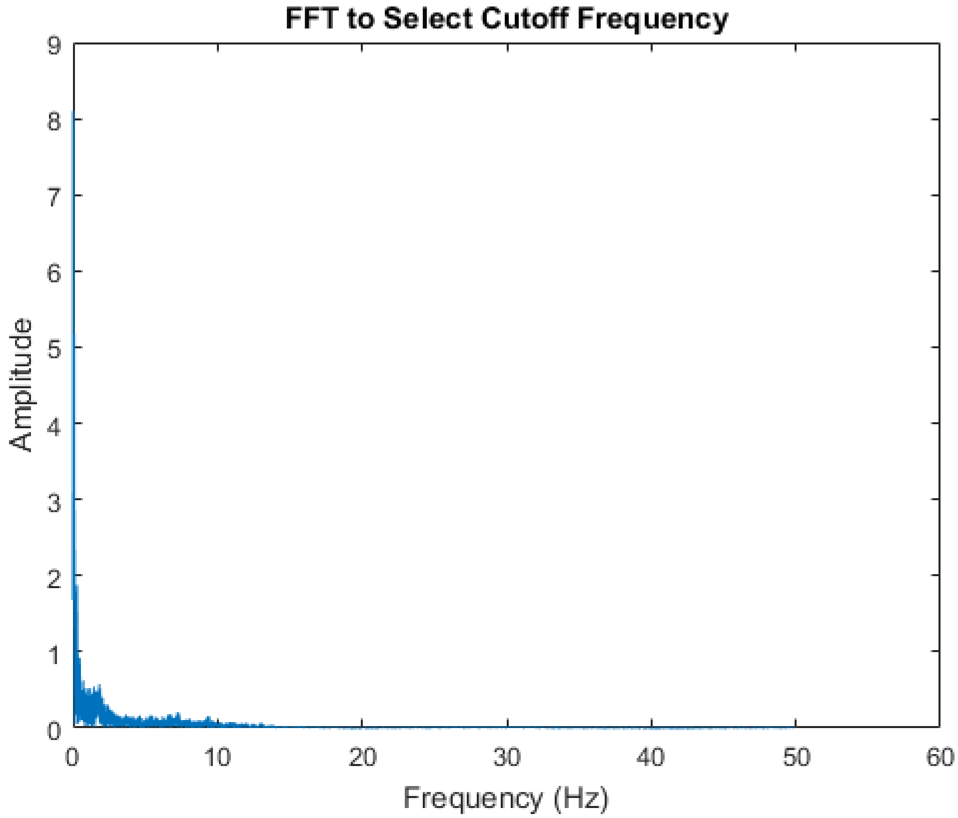
```
% The Integrated Velocity, and the FricCoeff from Filter Switch
[FGmu1,FGmu2] = FrictionEst(FricCoeff,v,idx1,idx2,idx3,idx4)
LitValue = 0.55;
Relative_Error1 = abs((FGmu1 - LitValue))/LitValue*100
Relative_Error2 = abs((FGmu2 - LitValue))/LitValue*100
```

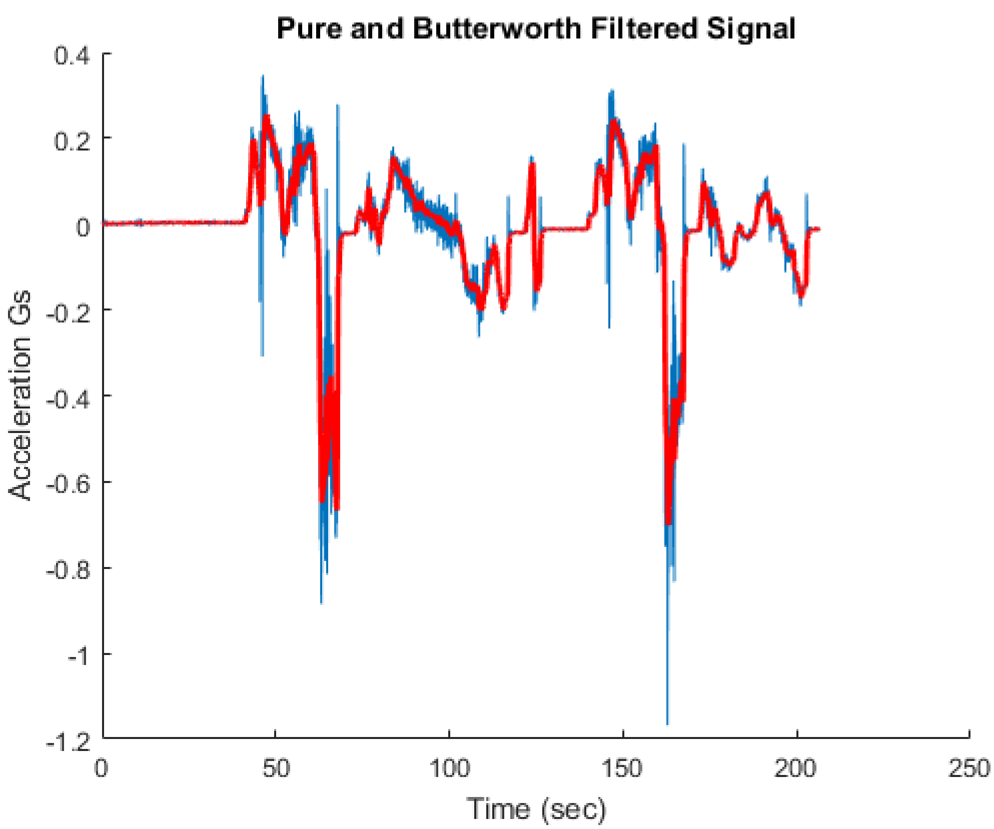
```
FGmu1 =
    0.5206
FGmu2 =
    0.4709
Relative_Error1 =
    5.3372
Relative_Error2 =
   14.3731
```

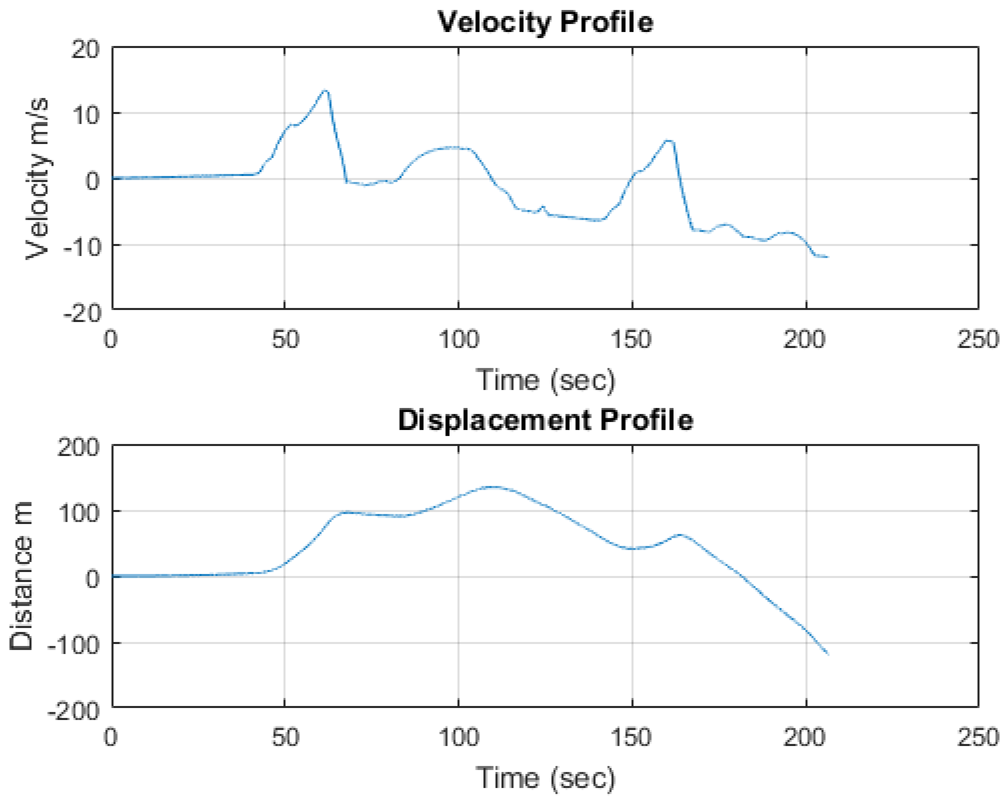


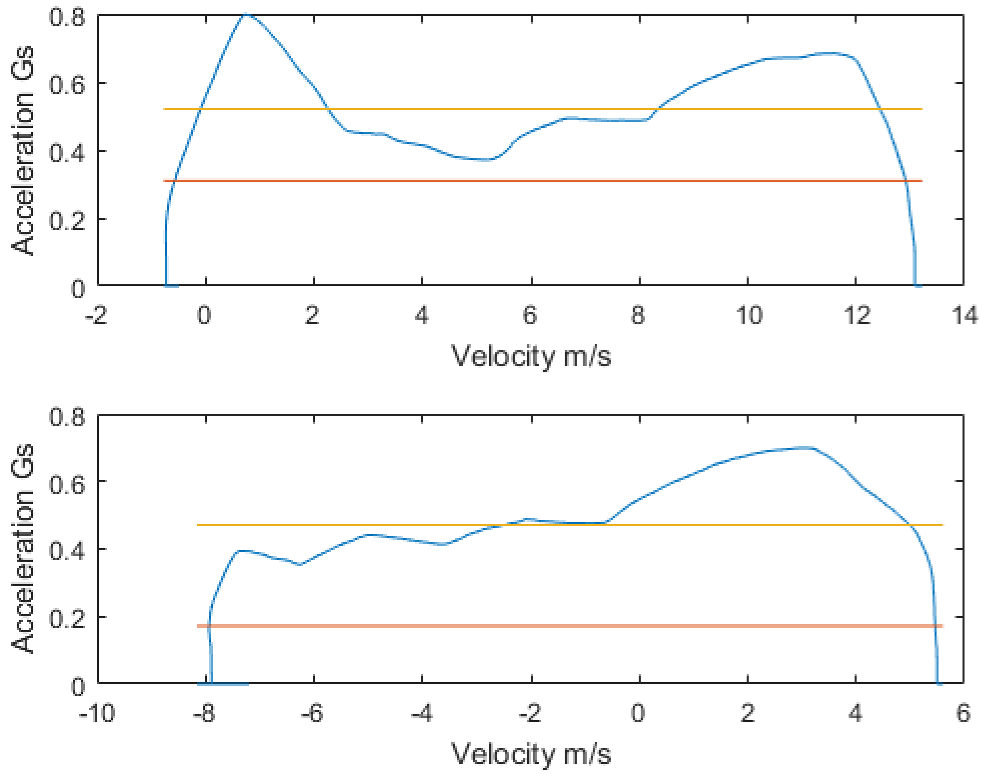


Pure Signal and Wavelet Transform Signal
Acceleration (Gs)
Time (sec)
Acceleration Profile Coiflet + Moving Average Filter
Moving Window Filtered
Average Switch
Acceleration (Gs)
Time (sec)
Detection of Coefficient of Friction Profiles
Acceleration (Gs)
Time (sec)


FFT to Select Cutoff Frequency
Amplitude
Frequency (Hz)
0
1
2
3
4
5
6
7
8
9
0
10
20
30
40
50
60


Pure and Butterworth Filtered Signal
Acceleration Gs
Time (sec)
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1.2
0
50
100
150
200
250


Velocity Profile
Velocity m/s
Time (sec)
Displacement Profile
Distance m
Time (sec)


Acceleration Gs
Velocity m/s
Acceleration Gs
Velocity m/s

Published with MATLAB® R2017a