Chalmers University of Technology

SSY130 Signal Processing

Hand in Problem 2

Yu-hsi LI

student_id: 20021026   *secret_key: 'Woobat'*

Questions

1. The code in FIG.1 shows a FIR lowpass filter that performs approximate differentiation for low frequencies up to frequency 0.05 Hz(f_pass), while blocking the frequency above 0.1 Hz(f_stop) to reduce the influence of the notice.
   Specifying the `firpm` Matlab command, the frequency vector `f` given the frequency points which the values are normalized by dividing by half the sampling frequency `f_s/2` (Nyquist frequency) to map them to the range [0, 1]. The passband frequency is in the range of [0, f_pass], and stopband is ranges from [f_stop, f_s/2]. Additionally, to differentiate the passband, we define the amplitude response vector `a`, which specifies the desired response at each frequency point. The values indicate that the filter should have a passband gain of 1 and a stopband gain of 0.
   Also, to ensure the amplitude vector is in the right unit, we multiply `f * ` 2π ` * f_s/2` result in the unit of radians per second.
   Lastly, the `'differentiator'` parameter will force the filter's phase to be $+90°$

2. See FIG. 2 and FIG. 3

3. We design the linear phase FIR filter by `firpm` command in MATLAB, which uses the Parks-McClellan algorithm, the filter's taps M introduces the delay, and the delay can be calculated by $(M-1)/(2*f_s) = (61-1)/(2*1) = 30$ sample since the sampling frequency is 1, it also means that it delay 30 seconds, which is because that the phase response is a linear function of frequency, and impulse respond is truncated to M samples than it delays $(M-1)/(2*f_s)$ to align the output with the input signal and maintain causality.

```
function h = gen_filter()

    %h = 0; %TODO: This line is missing some code!

    % Define the desired frequency response
    f_s = 1;         % Sampling frequency (1 Hz)
    f_pass = 0.05;   % Passband frequency (0.05 Hz)
    f_stop = 0.1;    % Stopband frequency (0.1 Hz)

    % Calculate the filter order (number of coefficients)
    filter_order = 60;

    f = [0, f_pass, f_stop, f_s/2] /(f_s/2);       % *pi [rad/sample]
    a = [1, 1, 0, 0] .*f*pi *f_s;                  % filter amplitude
    h = firpm(filter_order,f,a,'differentiator');

end
```

Figure 1: written code

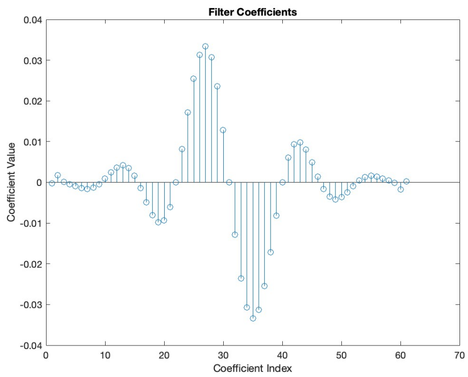

Figure 2: resulting filter coefficient

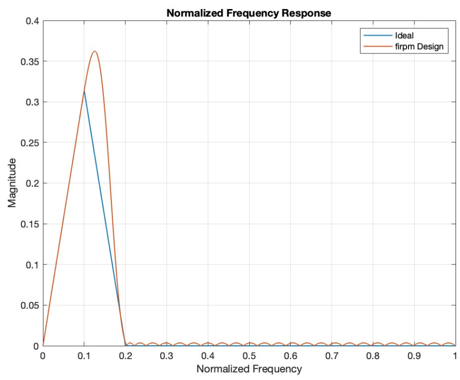

Figure 3: magnitude of frequency function with ideal filter

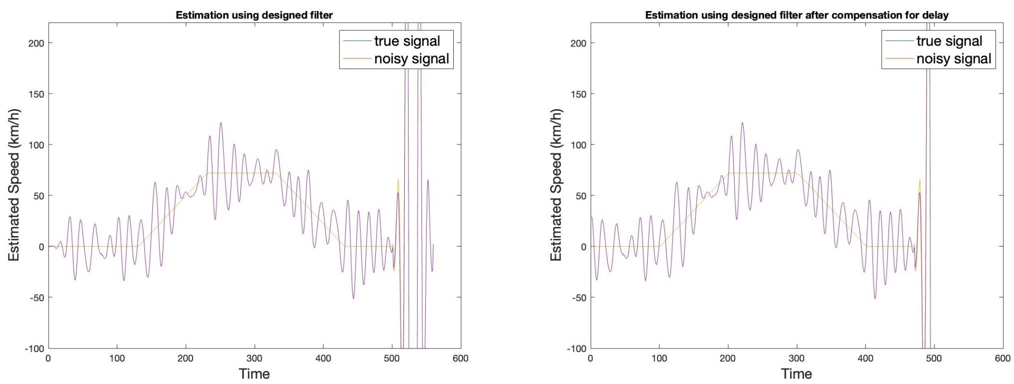

Figure 4: filtered true_position and noisy_position

4. The FIG.4 shows that filtered true_position and noisy_position signals, each with before and after delay compensation. The reason why at the end of the filter output, there are very large oscillations occur is due to the effect of convolution $v(n) = h(n)*s(n)$. If the impulse response $h(n)$ has size M and signal $s(n)$ has size N, using the finite signal strategy, each block will generate N+M-1 long output, at least M-1 element is calculated by the zero-padding s(n) in M-1 element. Then the last element in s(n) will be very big and due to zero-padding, it will also create a huge displacement with a large velocity.

5. The figure FIG.5 shows that filtered true_position and noisy_position signals compare with the Euler filter, each with before and after delay compensation. And FIG.6 shows the two signals filtered with Euler filter.

   The figures show that applying the Euler filter will result in velocity affected by noise since taking the signal that is corrupted by noise followed by a higher frequency component will have a large amplified magnitude.

   In the frequency domain derivative becomes multiplication. So higher frequency components will be amplified more and affect the output signal significantly.

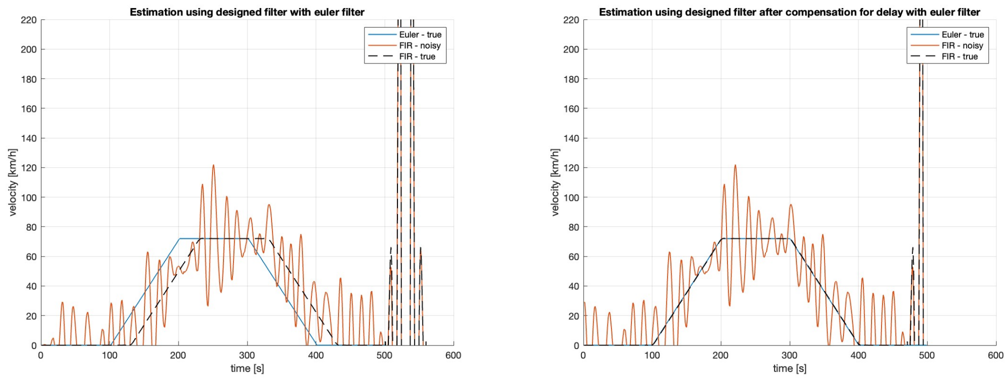

Figure 5: filtered true_position and noisy_position with Euler filter

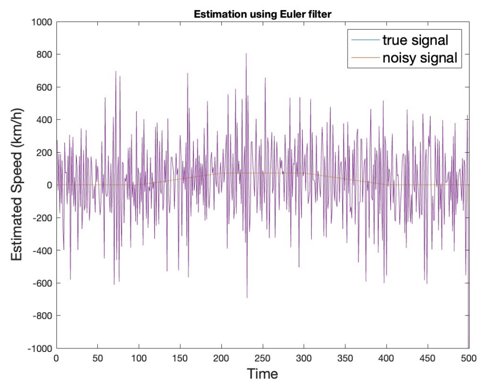

Figure 6: true_position and noisy_position with Euler filter

If we can filter the noise and take the derivative, amplified noise will decrease. That is to say, the risk of reducing the amplitude will reduce the gain of the true signal that is close to this frequency. Thus, each differentiator filter must be adjusted to the desired frequency range.

6. According to the setup above, the maximum velocity we can achieve is:
`the maximum of the vehicle found from the observed signal(noisy) is:122.0101 km/h`
`the maximum of the vehicle found from the true signal(true) is:72.1932 km/h`
If we wish to further reduce the noise, we can adjust the parameter by reducing the pass-band frequency (narrowing the noise) or increasing the order to lower ripple (but will also introduce more delay).