Digital Signal Processing Lab

Demo 4 - Exercise 2 (Matlab GUI)

Saad Zubairi
shz2020

September 24th, 2025

Solution

This solution contains a Matlab GUI app script that lets us visualize a low pass filter along with the pertaining Frequency Response (Magnitude) and the impulse response of the filter. The solution uses the uifigure function to add multiple ui elements such as a nested plot (which contains all the three plots) and a slider that controls the cut-off frequency.

Key changes made from the included file:

- A subplot has been added (using `uigridlayout` function)

```
    % grid layout
    grid = uigridlayout(my_fig,[3 1]);
    grid.RowHeight = {'1x', 'fit', 'fit'};
    grid.ColumnWidth = {'1x'};

    % sub-grid for plots
    plots = uigridlayout(grid,[1 3]);
    plots.RowHeight   = {'1x'};
    plots.ColumnWidth = {'1x','1x','1x'};

```

Snippet 1: plot changes

- The cut off frequency is controlled in a state variable defined as:

```
    fc = slider.Value;
    update_plot()

```

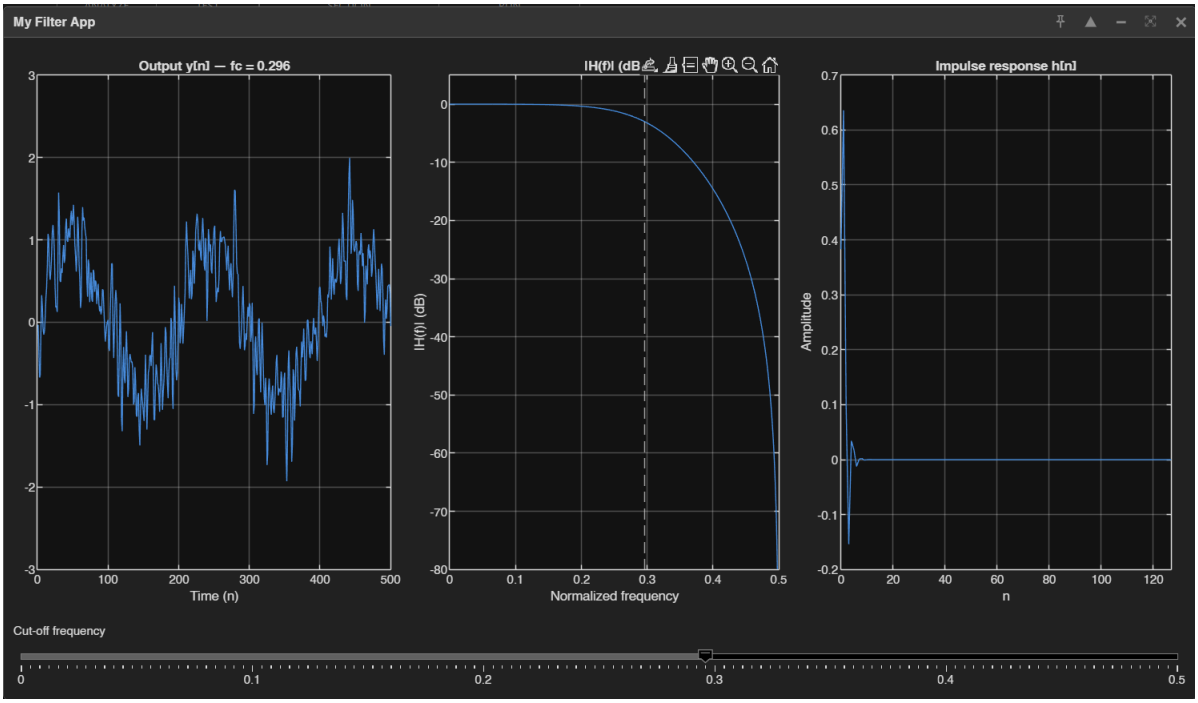
Snippet 2: state variable

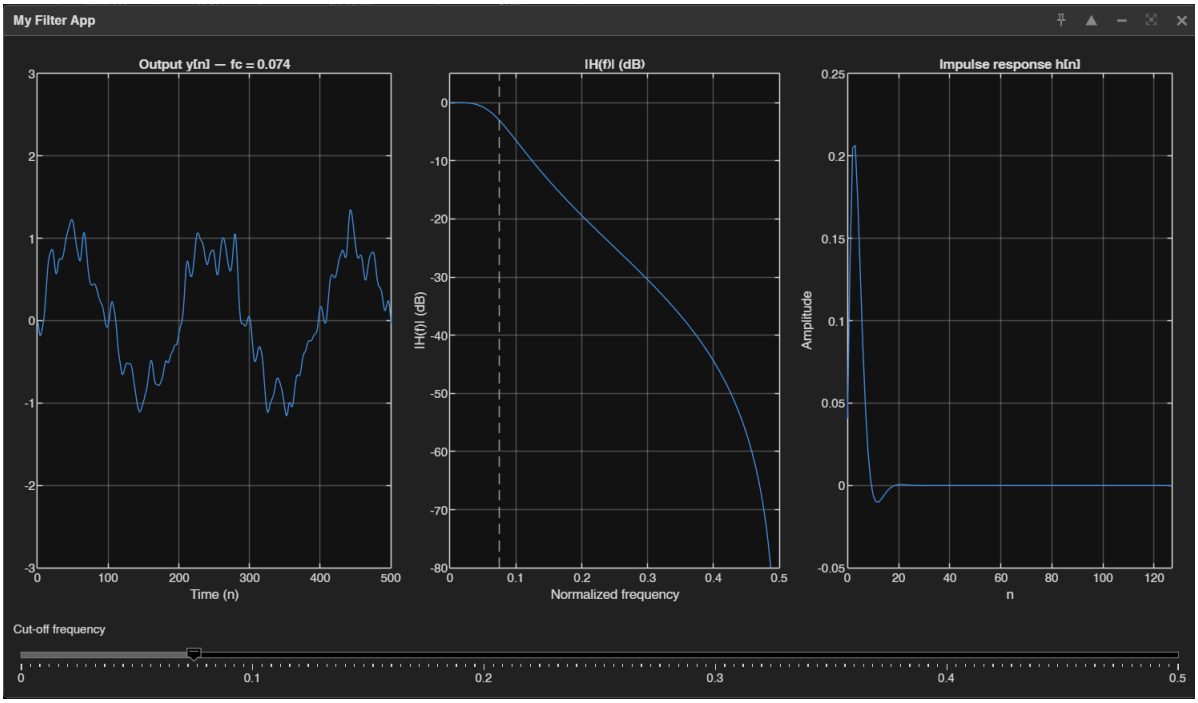
- The `update_graph` function accomodates the additional two graphs

```
    function update_plot()
        % design
        [b,a]=butter(2,2*fc);
        y=filtfilt(b,a,x);

        % time
        set(line_handle,'YData',y);

        % |H(f)| dB
        Nfft = 2048;
        [H,w]=freqz(b,a,Nfft,'half'); f=w/(2*pi);
        magdB = 20*log10(max(abs(H),1e-6));

        set(freq_line,'XData',f,'YData',magdB); ax2.Children(1).Value = fc;
        title(ax2,'|H(f)| (dB)');

        % h[n]
        Lh=128; h=filter(b,a,[1 zeros(1,Lh-1)]);
        set(imp_line,'XData',0:Lh-1,'YData',h); xlim(ax3,[0 Lh-1]); title(
    ax3,'Impulse response h[n]');
    end

```

Snippet 3: state variable

Screenshots





Code

```
function filter_app_ver2

% Set up figure and controls
my_fig = uifigure('Name',"My Filter App");
movegui(my_fig,'center');

% grid layout
grid = uigridlayout(my_fig,[3 1]);
grid.RowHeight = {'1x', 'fit', 'fit'};
grid.ColumnWidth = {'1x'};

% sub-grid for plots
plots = uigridlayout(grid,[1 3]);
plots.RowHeight   = {'1x'};
plots.ColumnWidth = {'1x','1x','1x'};

% Axes
ax  = uiaxes(plots);
ax.XGrid='on'; ax.YGrid='on'; xlabel(ax,'Time (n)'); xlim(ax,[0 500]); ylim(ax
    ,[-3 3]); box(ax,'on')

ax2 = uiaxes(plots);
ax2.XGrid='on'; ax2.YGrid='on'; xlabel(ax2,'Normalized frequency'); ylabel(ax2,'|
    H(f)| (dB)');
xlim(ax2,[0 .5]); ylim(ax2,[-80 5]); box(ax2,'on')

ax3 = uiaxes(plots);
ax3.XGrid='on'; ax3.YGrid='on'; xlabel(ax3,'n'); ylabel(ax3,'Amplitude'); box(ax3
    ,'on')

% Slider
slider_label = uilabel(grid);
slider_label.Text = 'Cut-off frequency';

slider = uislider(grid);
slider.Value = 0.3;
slider.Limits = [0 0.5];
slider.MajorTicks = 0:0.10:0.5;
slider.ValueChangingFcn = @slider_callback;

% Signal
N = 500; n = 1:N; x = sin(5*pi*n/N) + 0.5*randn(1,N);

line_handle = line(ax,n,x);

freq_line = plot(ax2,nan,nan); hold(ax2,'on')
xline(ax2,slider.Value,'--'); hold(ax2,'off')

imp_line  = plot(ax3,nan,nan);

% state
fc = slider.Value;
update_plot()

    function slider_callback(~,evt)
        fc = max(0.01,min(0.49,evt.Value));
```

```
        update_plot()
    end

    function update_plot()
        % design
        [b,a]=butter(2,2*fc);
        y=filtfilt(b,a,x);

        % time
        set(line_handle,'YData',y);

        % |H(f)| dB
        Nfft = 2048;
        [H,w]=freqz(b,a,Nfft,'half'); f=w/(2*pi);
        magdB = 20*log10(max(abs(H),1e-6));

        set(freq_line,'XData',f,'YData',magdB); ax2.Children(1).Value = fc;
        title(ax2,'|H(f)| (dB)');

        % h[n]
        Lh=128; h=filter(b,a,[1 zeros(1,Lh-1)]);
        set(imp_line,'XData',0:Lh-1,'YData',h); xlim(ax3,[0 Lh-1]); title(ax3,'
    Impulse response h[n]');
    end
end
```

Snippet 4: example code