EEE 202, Design HW **NAME:___SOLUTIONS___**

Problem 1: An RC circuit is to be designed to filter an audio signal with a DC bias so that the DC is eliminated. Since large filter transients are in general undesirable, the RC time constant should not be too large.

To create a more specific objective we consider the sum of:

1. A useful 40Hz signal $x_0(t) = \sin 40 \times 2\pi t$
2. A drift 0.1Hz signal $x_n(t) = \sin 0.1 \times 2\pi t$

We would like to select suitable R, C such that the RC circuit with transfer function

$$H(s) = \frac{RCs}{RCs + 1}$$

lets the useful signal through unchanged and stops the drift signal, as much as possible. For example, we would like to minimize the power of the error signal

$$H[x_0 + x_n] - x_0 = (1 - H)[x_0] + H[x_n]$$

As a first approximation, we can try to choose RC such that the steady state amplitudes of the two components are the same. These amplitudes can be computed from the Bode plots of (1-H(s) and H(s).

1. Use MATLAB to perform the necessary computations to obtain a coarse solution for the filter time constant RC;
2. select common values for a Resistor and a Capacitor to implement the filter;
3. simulate the time response of the filter and estimate approximately how long it would take to reach steady-state;
4. verify that the simulated signal amplitudes and time to steady-state are in agreement with the predictions from the Bode plots and the transient response of the filter.

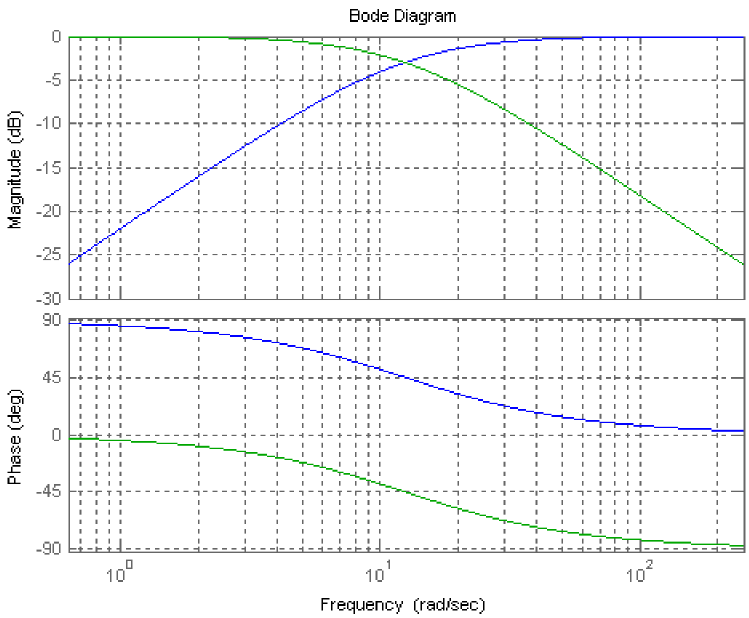
Relevant MATLAB commands:

```
t=[0:.001:20]';  % time vector definition
x0=sin(40*6.28*t);  % signal definition
xn=sin(0.1*6.28*t); % drift definition
RC=1/12.5; H=tf([RC 0],[RC 1]);  % filter definition
Bode(1-H,H) % Bode plots
plot(t,x0+xn, t,x0,t,lsim(H,x0+xn,t)) % plot of noisy signal,
                              % clean signal, and filtered signal
plot(t,lsim(H,x0+xn,t)-x0) % plot of error signal
```

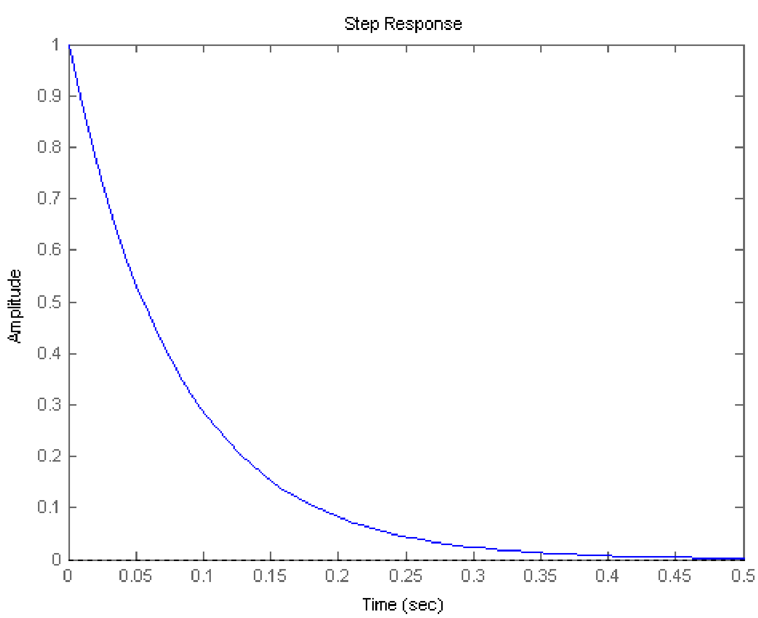
SOLUTION:

Using a quick trial-and-error iteration, we find the filter time constant that produces the same amplitude of distortion (1-H) and noise/drift (H) effects. (The theoretical optimization shows that the optimum is at the geometric mean of the two frequencies. This simple result is due to symmetry. It will change and require adjustment, if the noise and the signal have different amplitudes or the filters have different slopes.)

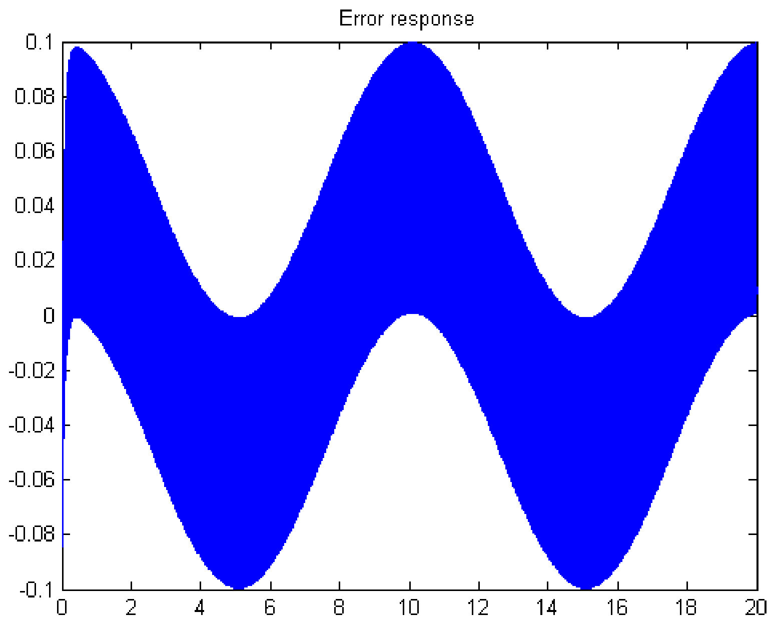
RC_opt = 1/12.5. This filter can be implemented with C = 1uF and R = 80kΩ, which are reasonably common values.



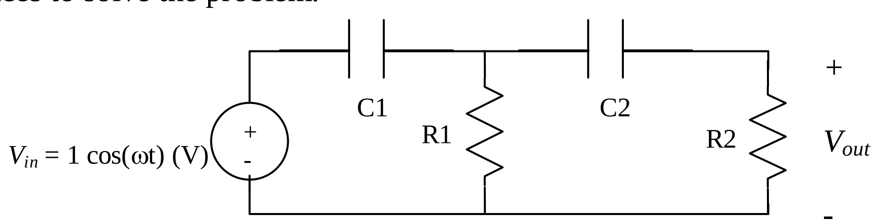
This filter has a response that stabilizes its transients in 4-time constants, or, roughly 0.3-0.4 seconds.



This is in agreement with the observed response to the two sinusoids. The total error is approximately 0.1, which is due to both sinusoids being attenuated by -26dB, i.e., by a factor of 20. Also, the initial transient appears to be no more than 0.5sec.



Problem 2: Repeat Problem 1 for a second order filter that is a cascade of two first order RC filters. Here you can use H2 = H*H to find the effective time constant. However, due to loading effects, simply cascading two first order filters does not produce the desired transfer function. Use Nodal/Loop analysis to find the transfer function of the general second order filter shown below, and based on the result make a reasonable selection of the two resistances and two capacitances to solve the problem.



SOLUTION: We calculate first the transfer function $V_{in} \rightarrow V_{out}$:
Using the intermediate node voltage $V_1, V_{out} = \frac{R_2}{R_2+1/C_2 s} V_1 = \frac{R_2 C_2 s}{R_2 C_2 s+1} V_1$
Then $V_1 = \frac{Z}{Z+1/C_1 s} V_{in}$, where $Z = R_1 || \left(R_2 + \frac{1}{C_s s}\right) = \frac{R_1 R_2 C_2 s + R_1}{R_2 C_2 s+1+R_1 C_2 s}$. It now follows that
$V_1 = \frac{(R_1 R_2 C_2 s+R_1) C_1 s}{(R_1 R_2 C_2 s+R_1) C_1 s+R_2 C_2 s+1+R_1 C_2 s} V_{in} = \frac{(R_1 R_2 C_1 C_2 s^2+R_1 C_1 s)}{R_1 R_2 C_1 C_2 s^2+(R_1 C_1+R_2 C_2+R_1 C_2)\, s+1} V_{in}$, and substituting in the first equation,

$$V_{out} = \frac{R_2 C_2 s}{R_2 C_2 s + 1} \cdot \frac{R_1 C_1 (R_2 C_2 s + 1) s}{R_1 R_2 C_1 C_2 s^2 + (R_1 C_1 + R_2 C_2 + R_1 C_2)\, s + 1} V_{in}$$

$$V_{out} = \frac{R_1 C_1 R_2 C_2 s^2}{R_1 R_2 C_1 C_2 s^2 + (R_1 C_1 + R_2 C_2 + R_1 C_2) s + 1} V_{in}$$

Comparing with the decoupled cascade of two first order filters, where $V_{out} = \frac{R^2 C^2 s^2}{(RCs+1)^2} V_{in}$, we see the double zero at 0, but the two poles are different. They only approach their decoupled case values $(-1/R_1 C_1, -1/R_2 C_2)$, (both set to -1/RC), as $R_1 C_2 \rightarrow 0$.

In order to obtain an approximate solution, we set $R_1C_2 = \frac{R_1C_1}{10}$ and using the previous values as a ballpark estimate, we start with $R_1C_1 = R_2C_2 = 1/12$. That would be easily achieved by scaling the original components by, say, a factor of 3 (reduce the first stage resistance and increase the second stage resistance), so such a solution is feasible.

After a few quick iterations yield RC_opt = 1/3.6. This filter can be implemented with $R_1 = 28k\Omega, C_1 = 10\mu F, R_2 = 126k\Omega, C_2 = 2.2\mu F$. The second resistance is a bit large (hence, noisy) and the capacitors will probably need to be electrolytic but the values are reasonable.

On the other hand, the step response of the filter is considerably slower, since we lowered its bandwidth. It now takes about 2 seconds to stabilize, which is certainly significant for an audio application. The benefit is that we gained an additional 4dB of attenuation. This is consistent with the time response simulation which shows smaller errors but takes longer to reach steady-state.

MATLAB: (We use H0 for the ideal decoupled filter, for comparison and w as a vector with the frequency range of interest)

```
RC=1/3.6; H0=tf([RC^2 0 0],[RC^2 2*RC 1]); H=tf([RC*RC 0 0],[RC*RC 2*RC+RC/10 1]);
bode(H,1-H,H0,w),grid, pause, plot(t,lsim(H,x0+xn,t)-x0)
```