Part A Introduction to cochlear-implant signal processing

a.

```
% Sampling rate according to Shannon et al's study followed
fs = 10000; % Hz

% Filter bank parameters
nb = [1, 2, 4, 8, 16]; %Frequency Bands
% Spread across same total frequency range 0 to 4 kHz
f_low = 0; % Hz
f_high = 4000; % Hz

filter_order = 3;   %Mentioned in original study that signal was split into frequency bands using third-order elliptical filter
filter_ripple = 15; %Adjacent filters overlapping at the point where the output from each filter was 15 dB down from the level

env_filter_order = 2;
env_filter_ripple = 6;  %Filter ripple of -6 dB per octave

% Pre-emphasis filter constructed using butter (applied according to original study)
pre_emphasis_fc = 1200; % Hz of low pass filter
[b_pre,a_pre] = butter(1, pre_emphasis_fc/(fs/2), 'high');

% Bandpass filter bank
for n = 1:length(nb)
    % Compute center frequencies for each band
    fc = linspace(f_low, f_high, nb(n)+2);
    fc = fc(2:end-1);

    wn = [fc(1)/(fs/2) fc(end)/(fs/2)];
    % Compute filter parameters for each band using third-order elliptical filters
    [b, a] = ellip(filter_order, filter_ripple, 40, wn, 'bandpass');

    % Save filter coefficients for each band
    filter_bank(n).b = b;
    filter_bank(n).a = a;
end
```

As seen in the code, the filter-bank reconstruction is designed with the specified number of frequency bands (nb) equal to 1, 2, 4, 8, 16 spread across the same total frequency range (from 0 to 4 kHz) and uses the same basic filter parameters as in the Shannon et al. study

```
% Generate noise carrier through using white noise (according to original study)
t = 0:1/fs:1;
noise_carrier = randn(length(t), 1);
```

The white noise carrier that was referenced in Shannon et al.'s study

b. Various Envelope Extractions:
   a. 16 Hz (see below)
   b. 160 Hz (see below)

```
% Envelope extraction filters as specified
env_fc = [16, 160];
% For each cutoff frequency
for n = 1:length(env_fc)
    [env_b, env_a] = ellip(env_filter_order, env_filter_ripple, 40, env_fc(n)/(fs/2), 'low');

    % Save filter coefficients for each filter
    env_filter(n).b = env_b;
    env_filter(n).a = env_a;
end
```

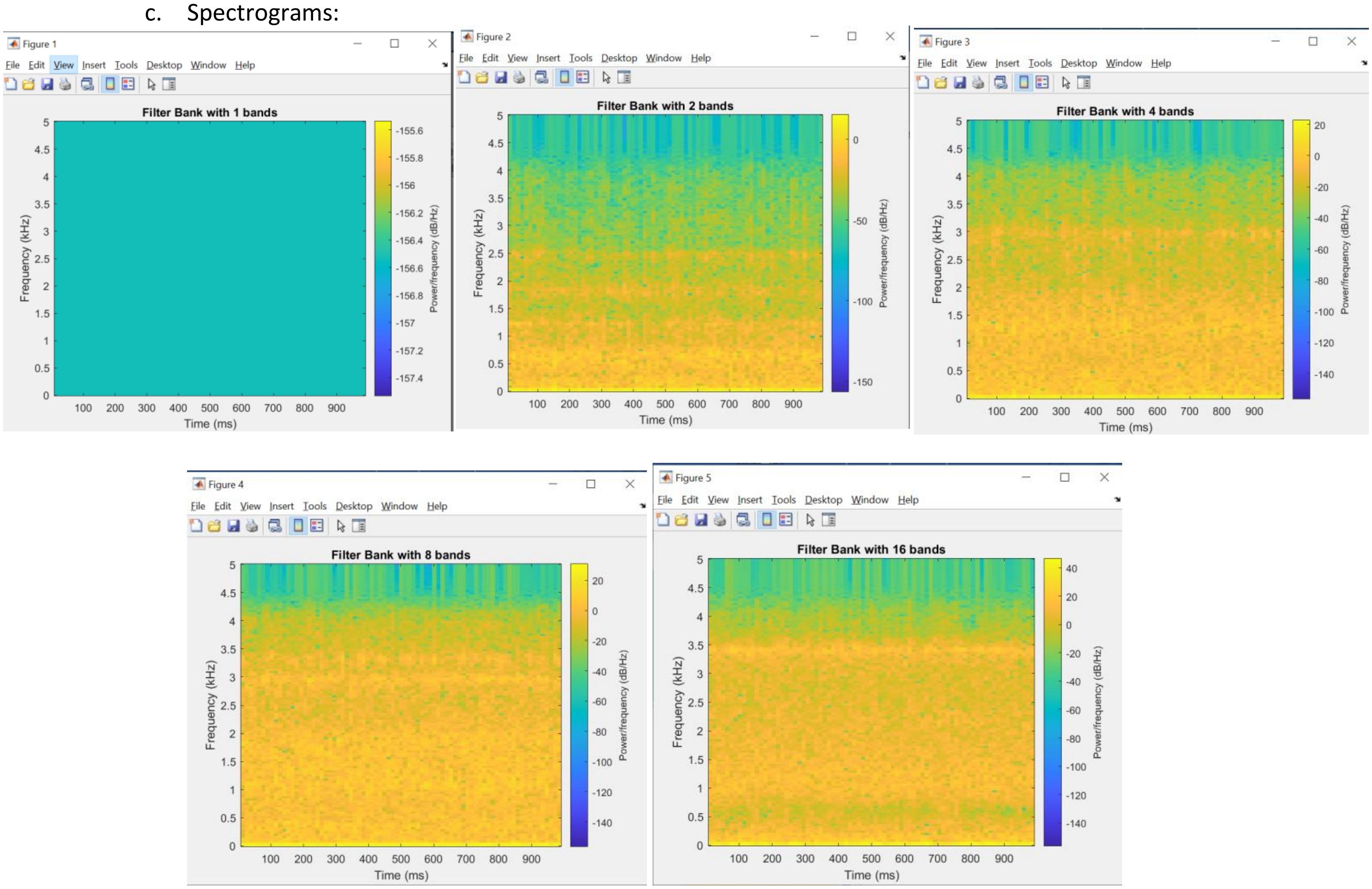
I created an array for the different rectification/low pass filters through incorporating them within a for loop instead of doing them individually
The two filter coefficients (outputs of the elliptical filter function) for each cutoff frequency is saved in the array "env_filter", respectively

c. Hilbert Transform

```
% Apply filter bank and Hilbert transform envelope extraction
y = filter(filter_bank(n).b, filter_bank(n).a, noise_carrier);
env = abs(hilbert(y));
```

For the Hilbert transform, instead of looping over an array for different cutoff frequencies and then applying the envelope extractions individually I was able to incorporate it within the number of bands loop, apply the filter bank and perform the Hilbert transform envelope extraction by taking the absolute value of the Hilbert function output, giving me the magnitude of the complex number

c. Spectrograms:



Spectrogram figures for envelope extractions using 16 & 160 Hz low-pass filters (cochlear_replicate_study_lowpass)

As evident on the figures, the spectrograms for each band shows that the higher frequency bands have more energy at higher frequencies (represented through the brighter yellow colors). Additionally, as the number of bands increases there is a smoother input signal, which makes sense because having more frequency bands would allow for better frequency resolution

d. Simple Perceptual Study Results:

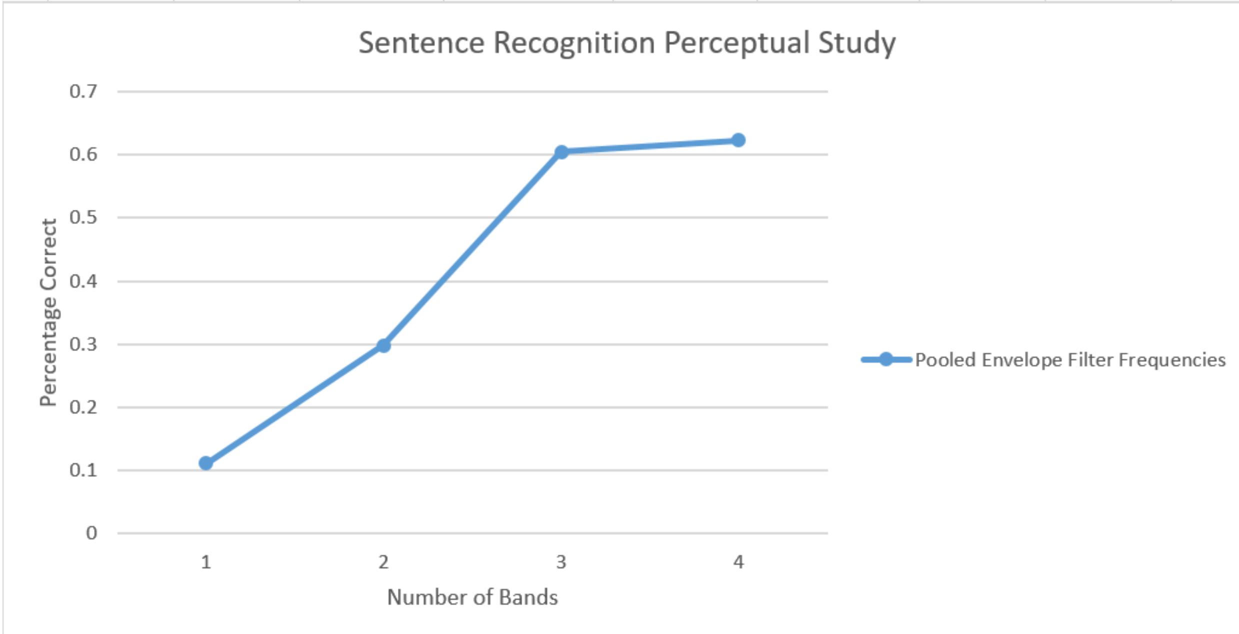


Figure for Perceptual Study Sentence Recognition
(using the files from SPINsents.zip)

Part of Study Data:

Files	Words	First	Second	Third	Fourth	First_Avg	Second_Avg	Third_Avg	Fourth_Avg
101	7	3	4	2	1	0.4285714	0.571428571	0.2857143	0.14285714
102	6	6	6	3	2	1	1	0.5	0.33333333
103	7	4	4	2	1	0.5714286	0.571428571	0.2857143	0.14285714
104	6	5	4	3	2	0.8333333	0.666666667	0.5	0.33333333
105	7	0	0	0	0	0	0	0	0
106	7	5	4	3	2	0.7142857	0.571428571	0.4285714	0.28571429
107	7	6	5	3	1	0.8571429	0.714285714	0.4285714	0.14285714
108	5	3	3	1	1	0.6	0.6	0.2	0.2
109	6	4	6	3	0	0.6666667	1	0.5	0
110	5	5	5	3	0	1	1	0.6	0
111	5	5	4	2	0	1	0.8	0.4	0
112	5	3	2	1	0	0.6	0.4	0.2	0
113	6	4	4	3	3	0.6666667	0.666666667	0.5	0.5

As seen in the perceptual study conducted using the code written (located in the “perceptual_study” file), the results above resemble that of Shannon et al’s results. This would indicate that the number of noise bands possesses a positive correlation with speech recognition. The higher the number of noise bands within the filter bank the higher the percentage of speech recognition. Although the perceptual study conducted in our case was much simpler, the results agree.

Part B Exploring the potential for frequency modulation information to improve cochlear-implant signal processing

a. Code understanding:

```
zfilt1 = fftfilt(b(:,k), x1);
zfilt2 = fftfilt(b(:,k), x2);
```

The Hilbert transform is done here through applying the complex filter to the original signals

```
% interchange envelope and fine structure
e1_fs2_filt = abs(zfilt1) .* cos(angle(zfilt2));
e2_fs1_filt = abs(zfilt2) .* cos(angle(zfilt1));
```

The absolute value of the output is then used to obtain the Hilbert envelope in each band
Note that the absolute value of the complex filter output gives the magnitude of the complex number at each time point. Multiplying by the cosine of the phase angle gives the real part of the complex number at that specific point in time. If I am correct this should remove the imaginary part (fine structure) and leave only the magnitude (envelope).

The spectrally matched noise is created through the function psd_matched_noise. It implements a spectrally matched noise by randomizing the phase of the Fourier spectrum while preserving the magnitude. This technique ensures that the synthesized noise has the same power spectrum as the original signal as seen below.

```
function n = psd_matched_noise(x)
% PSD_MATCHED_NOISE - Synthesize noise N having the same power spectrum
%             and duration as original signal X.
%             Done by randomizing phase of Fourier spectrum.
% Usage:  n = psd_matched_noise(x)
%
%   Copyright Bertrand Delgutte, 1999-2000
%
n = real(ifft(abs(fft(x)) .* exp(j*2*pi*rand(size(x)))));
```

The function first takes the FFT of the original signal to obtain its complex Fourier spectrum, it then multiplies this with a randomly generated phase spectrum in the frequency domain; finally the inverse FFT is taken to get the synthesized noise signal in the time domain.

b. Speech to Speech Chimera Reception

101.wav	_env+102.wav	_fts-nb1
101.wav	_env+102.wav	_fts-nb8
101.wav	_env+102.wav	_fts-nb16
101.wav	_env+102.wav	_fts-nb32
102.wav	_env+101.wav	_fts-nb1
102.wav	_env+101.wav	_fts-nb8
102.wav	_env+101.wav	_fts-nb16
102.wav	_env+101.wav	_fts-nb32
103.wav	_env+104.wav	_fts-nb1
103.wav	_env+104.wav	_fts-nb8
103.wav	_env+104.wav	_fts-nb16
103.wav	_env+104.wav	_fts-nb32
104.wav	_env+103.wav	_fts-nb1
104.wav	_env+103.wav	_fts-nb8

Files obtained through running “make_band_chimeras_modified” code
(Note the envelope of the sound is combined with the fine structure of another file in each case)

Speech Reception Results:

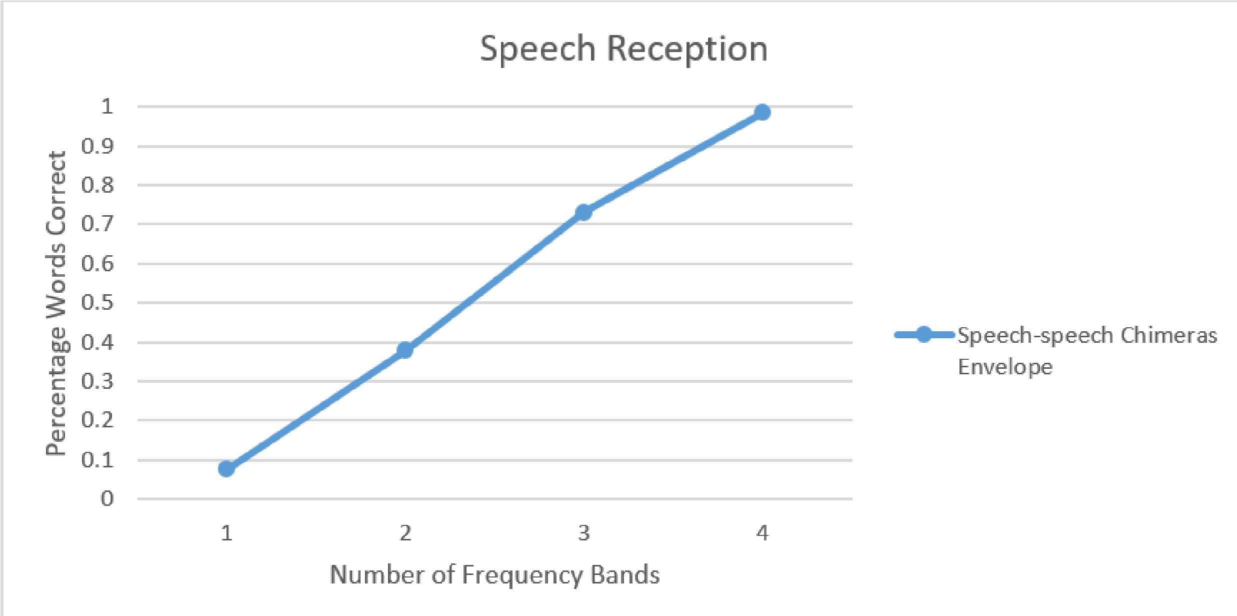
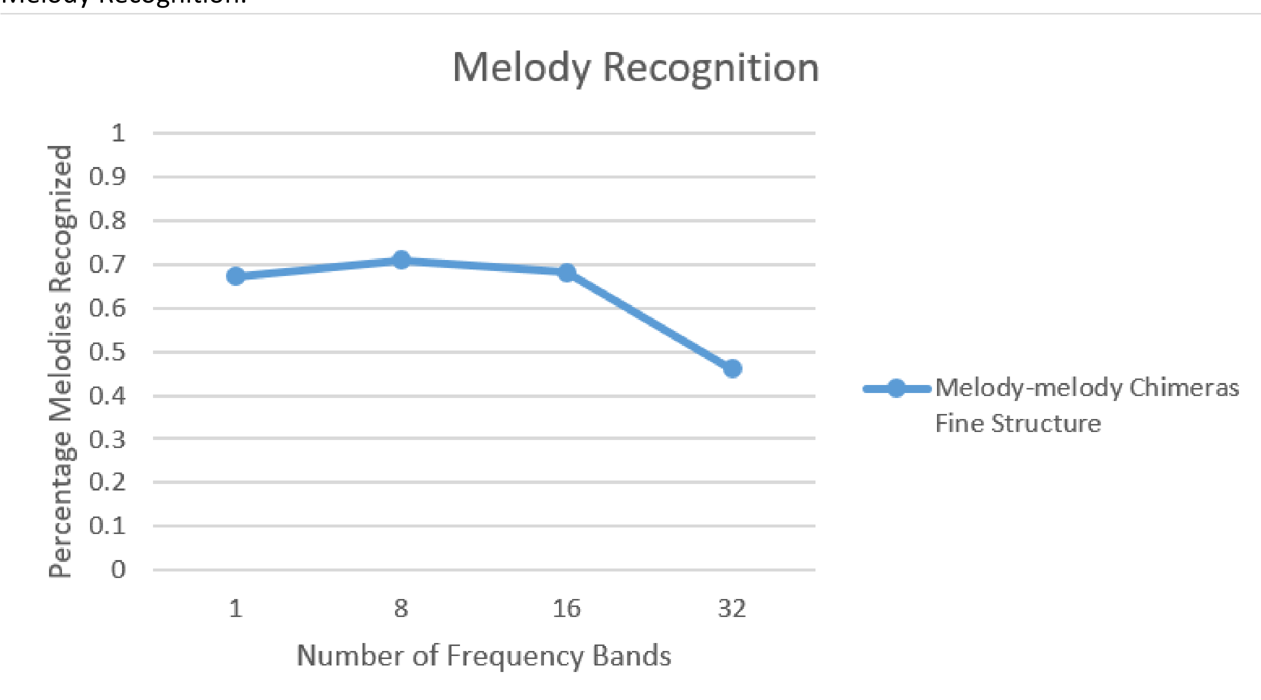


Figure for Speech-Speech Chimera Reception

Data:

Number of Bands	1	8	16	32
Average	0.07619	0.379524	0.731429	0.985714

Melody Recognition:



*Note that for melody recognition, the actual corresponding notes for the individual melodies (saved as .wav files in the 'melodies' directory) were available. I then attempted to use my perfect pitch to identify the notes to the best of my ability

c. The change could potentially affect the chimeric stimuli, but the significance of the effect depends on the specific characteristics of the signals being processed and the listener's sensitivity to the changes, this could happen because:
    a. The rectifier/low-pass filter approach might produce less accurate or less detailed envelopes, leading to potential perceptual differences in the chimeric stimuli, especially if the original signals contain fast amplitude fluctuations
    b. The low-pass filter used for envelope extraction in the rectifier/low-pass filter approach introduces a delay and possible phase distortion, which could alter the phase relationship between the extracted envelope and the fine structure, affecting the resulting chimeric signals
    c. The choice of the low-pass filter's cutoff frequency is important for accurately capturing the envelope information. If the cutoff frequency is too low or too high, it may not accurately represent the desired envelope

d. Our findings showed that for speech reception, conflicting speech information in the envelope and fine structure and increasing number of frequency bands resulted in a positive correlation with the percentage words correct. On the other hand, for melody recognition with conflicting envelope and fine structure information there was a slight negative correlation between the number of frequency bands and percentage melodies recognized when the number of frequency bands exceeded 16.
   This would support and complement the conclusions made by Smith et al, based on the two figures which show:
    i. Speech performance was better when speech information was present in the envelope (solid blue-line) as opposed to the fine-time structure (solid red-line). These findings suggest that envelope information is critical for speech perception
    ii. Melody recognition was better when music information was present in the fine-time structure. The melody-melody chimeras (dashed red-line) containing conflicting melodies in the envelope and fine structure showed that listeners could still recognize the melody in the fine structure, suggesting that fine-time information is more important for music perception

e. Given that current CI processing primarily focuses on encoding and transmitting the envelope information of input signals. The Continuous Interleaved Sampling approach (CIA) as mentioned in Loizou's paper prioritizes the preservation of the envelope component to aid in speech perception. However, current strategies like these tend to sacrifice fine-time information, which may be crucial for music perception, as evident in the Smith et al study we replicated.
   Personally, I think improvements could be made based on personalizing CI processing, which would allow the process to be customized based on each individual's needs and preferences; through tailoring signal processing it might be possible to optimize speech and music perception for a wider range of CI users
   In addition, if it would be possible to develop two separate approaches, one optimized for envelope preservation and the other fine-time information if it would be possible to separate these 2 processing pathways and be able to adaptively switch between them, this could solve the problem present in current CI processing technology.

Part C Considering a technical artifact with chimera synthesis and potential implications

a. Zeng et al.'s technical issue concerning envelopes obtained by cochlear filtering was explored by Gilbert and Lorenzi and Heinz and Swaminathan. Both studies addressed potential chimera synthesis artifacts and their effects on perceptual and physiological outcomes using different strategies. Gilbert and Lorenzi used a perceptual approach, employing stimuli with varying frequency bands to assess the listeners' ability to distinguish genuine and fake stimuli. They used a signal-detection theory approach to measure listeners' sensitivity to envelope and fine structure variations, evaluating whether broad chimera filters influenced perceptual salience.

   Heinz and Swaminathan on the other hand, investigated the issue using a more physiological approach. They simulated auditory nerve responses to chimera stimuli using a computational model with realistic cochlear filtering, nonlinear transduction, and auditory nerve fiber characteristics. The model's output was analyzed to assess the relationship between acoustic stimuli and auditory nerve responses using metrics like cross-correlation coefficients and mutual information. Both studies discovered that broad chimera filters could affect envelope recovery by cochlear filtering, but not as significantly as Zeng et al. claimed. The reversal in envelope/fine-time salience for speech seen in Smith et al's Figure 2 is unlikely due to this artifact alone. The problem stems from a difference in bandwidth between chimera and auditory nerve filters, which could affect the brain's representation of the stimuli and influence perceptual and physiological outcomes.

b. If Zeng et al.'s artifact is correct, it could affect the interpretation of Smith et al.'s results and conclusions about the relative importance of envelope and fine-time information in speech and music perception. The artifact suggests that the observed transitions between envelope and fine-time salience may be due to stimulus artifacts rather than perceptually based transitions. This implies that the chimera synthesis algorithm may fail to generate AN responses with the desired envelope and fine-time structure, casting doubt on Smith et al's conclusions about the importance of these features in speech and music perception.

   The potential impact of the artifact on cochlear implant (CI) processing may necessitate a reconsideration of the relative importance of envelope and fine-time information in CI processing. However, studies by Gilbert and Lorenzi and Heinz and Swaminathan indicate that the artifact may not be as important as Zeng et al. claimed. While the artifact may have an impact on CI processing strategies, it may not be the only factor determining the optimal balance of envelope and fine-time information, implying that Smith et al's findings may still be applicable to CI processing.