1. **Polyphase/DFT Filterbank:**
   In this problem, you will derive the equivalence between the uniformly modulated filterbank in Fig. 1 and its polyphase/DFT implementation in Fig. 2. Assume that the impulse response lengths of \( H(z) \) and \( K(z) \) both equal \( M \), a multiple of \( N \). The impulse responses of the polyphase filters \( H^{(p)}(z) \) and \( K^{(p)}(z) \) are related to those of \( H(z) \) and \( K(z) \) as follows.

\[
\begin{align*}
    h^{(p)}_k &= h_{\ell N + p} \\
    k^{(p)}_k &= k_{\ell N + p}
\end{align*}
\]

(a) Show the equivalence between the analysis banks in Fig. 1 and Fig. 2. (Hint: Using Fig. 1, derive an expression for \( s_i(m) \) in terms of input \( x(n) \) and filter coefficients \( \{h_n\} \). Then convert to polyphase notation using \( x^{(p)}(m) \) and \( h^{(p)}_n \), and finally \( w^{(p)}(m) \).)

(b) Show the equivalence between the synthesis banks in Fig. 1 and Fig. 2. (Hint: Using Fig. 1, derive an expression for \( u(n) \) in terms of inputs \( s_i(m) \) and filter coefficients \( \{k_n\} \). Then convert to polyphase notation using \( u^{(p)}(m) \) and \( k^{(p)}_n \), and finally \( v^{(p)}(m) \).)

(c) Implement the filterbank pairs of Fig. 1 and Fig. 2 in Matlab using \( N = 8 \), filters of length \( M = 64 \), and input data created via \( x = \text{randn}(1,100) \). Using the following impulse response for both \( H(z) \) and \( K(z) \).

\[
h = \text{remez}(M-1, [0,.8/N,1.2/N,1],[\text{sqrt}(N),\text{sqrt}(N),0,0]);
\]

Plot the output from both filters as well as the \( M \)-delayed input as done in Fig. 3.

---

**Figure 1:** \( N \)-band uniformly-modulated analysis/synthesis filterbanks.
Figure 2: Polyphase/DFT implementation of $N$-band uniformly modulated analysis/synthesis filterbanks.

Figure 3: Matlab filterbanks simulation outputs.
2. **MPEG Prototype Filter Design:**

Here we focus on prototype filter design for the MPEG-style cosine-modulated filterbank. The notes derived the following expression for the transfer function of the composite system and derived \( \{a_i\} \) and \( \{c_i\} \) which result in real-valued filter coefficients and near-perfect reconstruction.

\[
Q(z) = \frac{U(z)}{X(z)} = \frac{2M - 2}{N} \left( \sum_{i=0}^{N-1} \text{Re}(a_i c_i) \cos \left( \frac{\pi (2i+1)}{2N} n \right) - \text{Im}(a_i c_i) \sin \left( \frac{\pi (2i+1)}{2N} n \right) \right) \left( \sum_{k=0}^{M-1} h_k h_{n-k} \right) z^{-n},
\]

Above, \( N \) is the number of sub-bands, \( M \) is the prototype filter impulse response length, and \( \{h_k\} \) is the prototype filter impulse response. Recall that in MPEG, \( N = 32 \) and \( M = 513 \).

(a) Assuming a unit-variance white input process \( \{x(n)\} \), derive an expression for reconstruction error variance

\[
\sigma_e^2 = \mathbb{E}\{|u(n) - x(n - M + 1)|^2\}
\]

in terms of the impulse response of the prototype filter \( \{q_n\} \).

(b) Using the MPEG prototype filter coefficients in the file\(^1\) `h.mpeg.mat`, plot in dB:

i. the prototype filter DTFT magnitude \( |H(\omega)| \) over \( 0 \leq \omega \leq \pi \),

ii. the prototype filter DTFT magnitude \( |H(\omega)| \) over \( 0 \leq \omega \leq \frac{2\pi}{N} \) (to better see the passband) superimposed on the ideal magnitude response, and

iii. the composite system DTFT magnitude \( |Q(\omega)| \) over \( 0 \leq \omega \leq \pi \),

and calculate \( \sigma_e^2 \). An example appears in Fig. 4.

(c) Using the `remez` filter design command, attempt to design a length-513 FIR filter with similar passband response but better composite response than the MPEG filter. Can you? Plot the same graphs as in (b) for your best design, and compute \( \sigma_e^2 \). (Hint: make minor adjustments to the passband and stopband cutoff frequencies so that the composite’s passband response alternates between \( \pm \epsilon \) dB for some very small \( \epsilon \).)

(d) Attempt (c) using the `firls` filter design command. Does this seem to be a better design technique?

---

\(^1\)See the course web page.

©P. Schniter, 1999
Figure 4: Prototype-filter and composite-system DTFT magnitude responses.
3. MPEG Filterbank Implementation:
Here we will implement the polyphase/DCT version of the MPEG filterbank illustrated in Fig.s 25-26 of the subband coding notes.

(a) Implement the filterbank using the prototype filter in h.mpeg.mat and an input generated by 
\( x = \text{randn}(1,10000) \). (Hints: Zero-pad the beginning of the input record so that \( x(0) \) is the only non-zero value used to code the first frame. Zero-pad the end of the input record so that the length of the padded record is a multiple of \( N \). Don’t implement the DCT until everything else works.)

Plot the output \( u(n) \) and the reconstruction error \( e(n) = u(n) - x(n - M + 1) \) for comparison with an \( M \)-delayed version of the input \( x(n) \). See Fig. 5 for an example.

(b) Calculate the mean-squared reconstruction error (MSRE):

\[
\text{MSRE} = \frac{1}{L} \sum_{n=M-1}^{M+L-2} |u(n) - x(n - M + 1)|^2
\]

where \( L \) is the length of the input record. How does it compare to \( \sigma_s^2 \)?

Figure 5: Input, output, and reconstruction error for MPEG filterbank.