ECE53800 Digital Signal Processing I
Matlab Homework 2:
Perfect Reconstruction Filter Banks

Boris Diner

November 20, 2021

0 Deriving the Uniform Filter Bank Equivalent to Tree-Structured Filter Bank

In this Matlab homework, we studied the M-channel uniform perfect reconstruction filter bank that we synthesized from the three stage tree-structured perfect reconstruction filter bank for the system composed of M=8 channels.

Using the Noble's decimation identity, we can derive the uniform filter bank in terms of the given low-pass $H_0^{(2)}(\omega)$ and high-pass $H_1^{(2)}(\omega)$ filters of the analysis side.

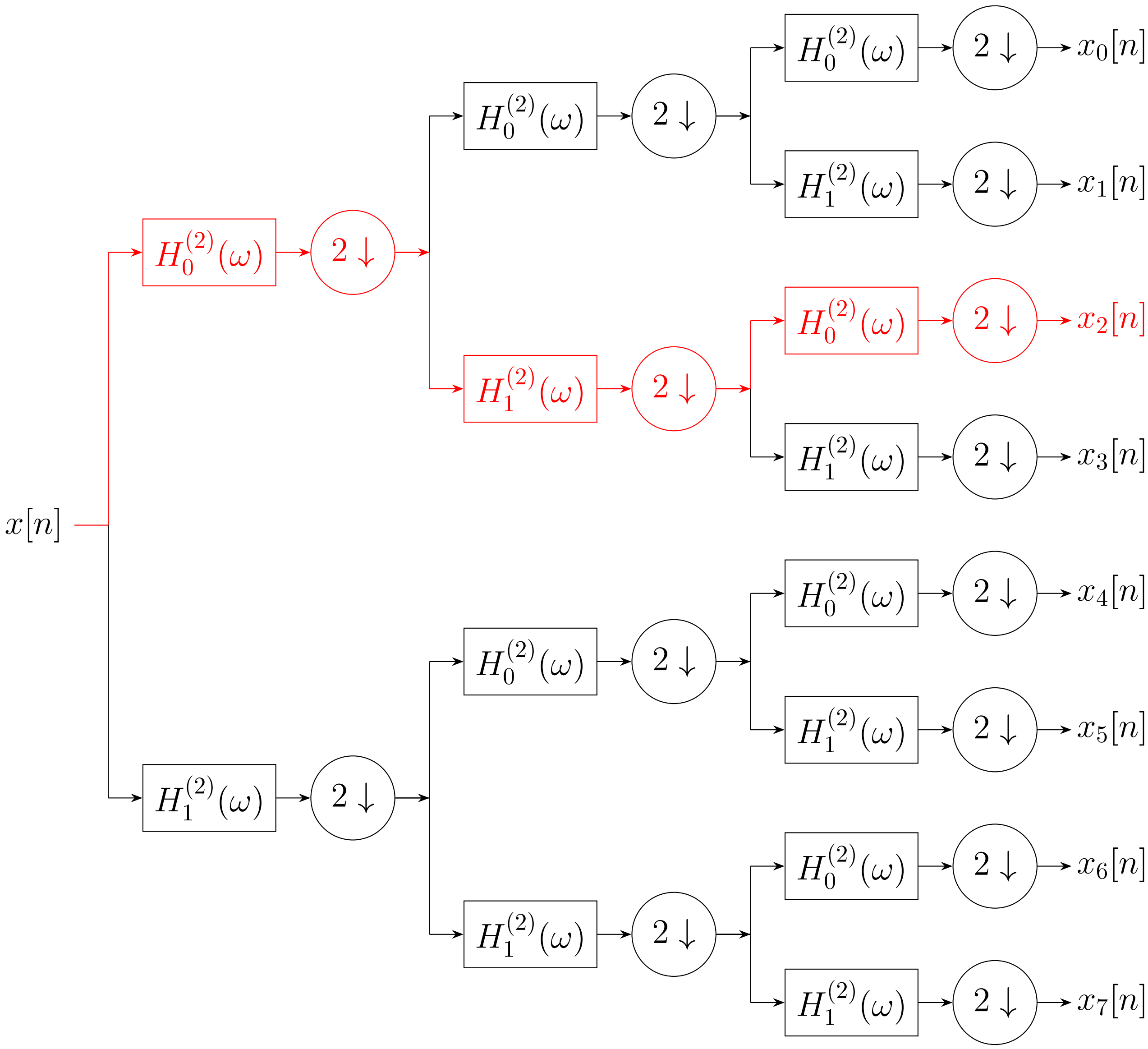

Figure 0(a): Analysis Section of Three-Stage Tree-Structured Filter Bank

Consider the part of the analysis side chain marked in red (see Figure 0(a)).

So we have the following chain (see Figure 0(b)). Using the Noble's decimation identity, we can exchange the first downsampler with the filter $H_1^{(2)}(\omega)$ (marked in blue).

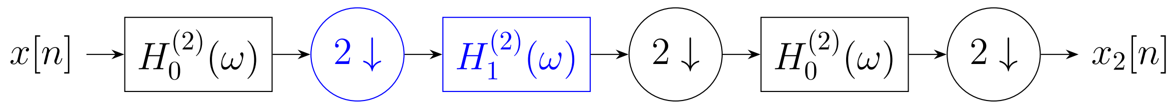

Figure 0(b): One of the chains of the analysis side

After the exchange, we get the following result (see Figure 0(c)).

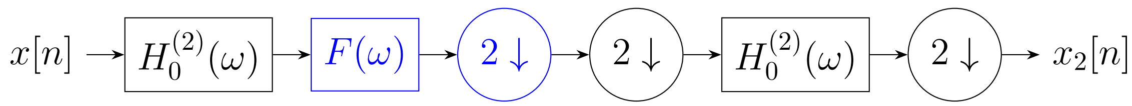

Figure 0(c): Applying the Noble's decimation identity to one chain of the analysis side

From the corresponding Noble's identity, we know that in order for schemes depicted on Figure 0(b) and Figure 0(c) to be equivalent, the following should hold, $F(\omega) = H_1^{(2)}(2\omega)$. Back to the time domain, we get $f[n] = \sum_{k=-\infty}^{+\infty} h_1^{(2)}[k]\delta[n-2k] \triangleq h_{10}^{(2)}[n]$. That is, the filter $f[n]$ is the same as the high-pass filter $h_1^{(2)}[n]$ of the analysis side of the 2-channel quadrature mirror filter (QMF) with 1 zero insert between each

2 successive values of the filter $h_1^{(2)}[n]$. It is true because

$$F(\omega) = \sum_{n=-\infty}^{+\infty} \sum_{k=-\infty}^{+\infty} h_1^{(2)}[k]\delta[n-2k]e^{-j\omega n} =$$
$$= \sum_{k=-\infty}^{+\infty} h_1^{(2)}[k] \sum_{n=-\infty}^{+\infty} \delta[n-2k]e^{-j\omega n} =$$
$$= \sum_{k=-\infty}^{+\infty} h_1^{(2)}[k]e^{-j(2\omega)k} = H_1^{(2)}(2\omega)$$

Finally, we also know that 2 successive downsamplers are equivalent to a single downsampler that decimates a signal by the factor equal to the product of factors of 2 successive downsamplers. For instance, 2 successive downsamplers each of which decimates by the factor of 2 are equivalent to one downsampler that desimates by the factor of $2 \times 2 = 4$). Therefore, we get the following chain.

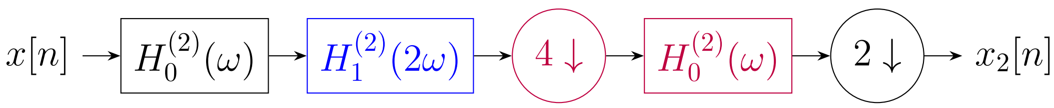

Figure 0(d): Applying the Noble's decimation identity to the next stage of the chain on the analysis side

Now applying the procedure described above to the part marked in purple, we get the following result.

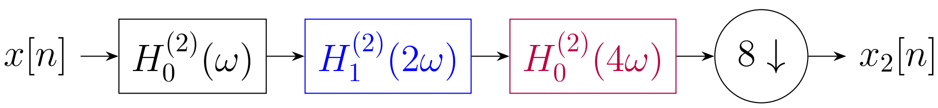

Figure 0(e): The resulting one chain of the analysis side of the uniform filter bank equivalent to the tree-structured filter bank

Here $H_0^{(2)}(4\omega)$ corresponds to $h_{000}[n] \triangleq \sum_{k=-\infty}^{+\infty} h_0^{(2)}[k]\delta[n-4k]$, that is the initial low-pass filter $h_0^{(2)}[n]$ of the 2-channel quadrature mirror filter (QMF) with 3 zero inserts between each 2 successive values of $h_0^{(2)}[n]$. The downsampler that decimates by the factor of 8 is the result of combining 2 successive downsamplers one of which decimates by the factor of 4 and another one that decimates by the factor of 2.

Thus, we have expressed one chain of the analysis side of the tree-structured filter bank via the equivalent uniform filter bank, and we got the following relationship

$$H_2(\omega) = H_0^{(2)}(\omega) \cdot H_1^{(2)}(2\omega) \cdot H_0^{(2)}(4\omega)$$

In the discrete time domain, we have the following structure.

$$x[n] \longrightarrow \boxed{h_0^{(2)}[n]} \longrightarrow \boxed{h_{10}^{(2)}[n]} \longrightarrow \boxed{h_{000}^{(2)}[n]} \longrightarrow 8\downarrow \longrightarrow x_2[n]$$

Figure 0(f): The resulting one chain of the analysis side of the uniform filter bank equivalent to the tree-structured filter bank in the DT domain

So we can compute the equivalent filter as

$$h_2[n] = h_0^{(2)}[n] * h_{10}^{(2)}[n] * h_{000}^{(2)}[n]$$

Here $*$ denotes the convolution, $h_{10}^{(2)}[n]$ is the high-pass filter of the analysis side of the 2-channel QMF filter bank with 1 zero insert between each 2 successive values of the filter $h_1^{(2)}[n]$, $h_{000}^{(2)}[n]$ is the low-pass filter of the analysis side of the 2-channel QMF filter bank with 3 zero inserts between each 2 successive values of the filter $h_0^{(2)}[n]$.

Therefore, we get the following relationships for all 8 chains.

$$h_0[n] = h_0^{(2)}[n] * h_{00}^{(2)}[n] * h_{000}^{(2)}[n] \overset{DTFT}{\rightarrow} H_2(\omega) = H_0^{(2)}(\omega) \cdot H_0^{(2)}(2\omega) \cdot H_0^{(2)}(4\omega)$$
$$h_1[n] = h_0^{(2)}[n] * h_{00}^{(2)}[n] * h_{100}^{(2)}[n] \overset{DTFT}{\rightarrow} H_2(\omega) = H_0^{(2)}(\omega) \cdot H_0^{(2)}(2\omega) \cdot H_1^{(2)}(4\omega)$$
$$h_2[n] = h_0^{(2)}[n] * h_{10}^{(2)}[n] * h_{000}^{(2)}[n] \overset{DTFT}{\rightarrow} H_2(\omega) = H_0^{(2)}(\omega) \cdot H_1^{(2)}(2\omega) \cdot H_0^{(2)}(4\omega)$$
$$h_3[n] = h_0^{(2)}[n] * h_{10}^{(2)}[n] * h_{100}^{(2)}[n] \overset{DTFT}{\rightarrow} H_2(\omega) = H_0^{(2)}(\omega) \cdot H_1^{(2)}(2\omega) \cdot H_1^{(2)}(4\omega)$$
$$h_4[n] = h_1^{(2)}[n] * h_{00}^{(2)}[n] * h_{000}^{(2)}[n] \overset{DTFT}{\rightarrow} H_2(\omega) = H_1^{(2)}(\omega) \cdot H_0^{(2)}(2\omega) \cdot H_0^{(2)}(4\omega)$$
$$h_5[n] = h_1^{(2)}[n] * h_{00}^{(2)}[n] * h_{100}^{(2)}[n] \overset{DTFT}{\rightarrow} H_2(\omega) = H_1^{(2)}(\omega) \cdot H_0^{(2)}(2\omega) \cdot H_1^{(2)}(4\omega)$$
$$h_6[n] = h_1^{(2)}[n] * h_{10}^{(2)}[n] * h_{000}^{(2)}[n] \overset{DTFT}{\rightarrow} H_2(\omega) = H_1^{(2)}(\omega) \cdot H_1^{(2)}(2\omega) \cdot H_0^{(2)}(4\omega)$$
$$h_7[n] = h_1^{(2)}[n] * h_{10}^{(2)}[n] * h_{100}^{(2)}[n] \overset{DTFT}{\rightarrow} H_2(\omega) = H_1^{(2)}(\omega) \cdot H_1^{(2)}(2\omega) \cdot H_1^{(2)}(4\omega)$$

Now we are going to consider the chain of the synthesis side marked in red (see Figure 0(g)).

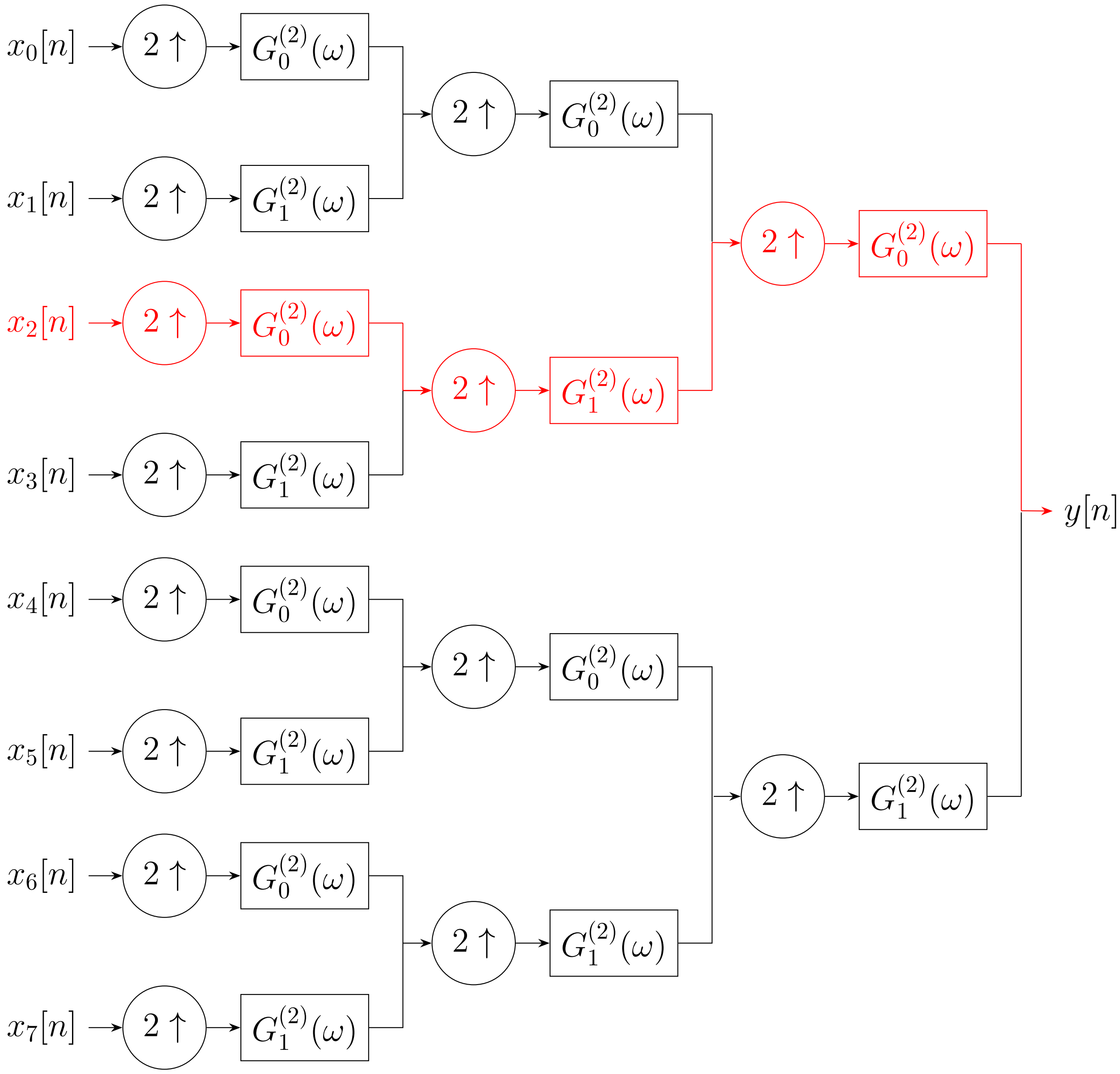

Figure 0(g): Synthesis Section of Three-Stage Tree-Structured Filter Bank

So we have the following chain (see Figure 0(h)). Using the Noble's upsampling identity, we can exchange the last upsampler with the preceding filter $G_1^{(2)}(\omega)$ (marked in blue).

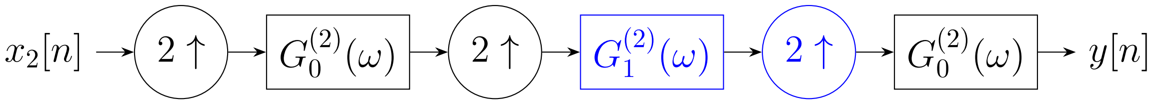

Figure 0(h): One of the chains of the synthesis side

In this case, we get

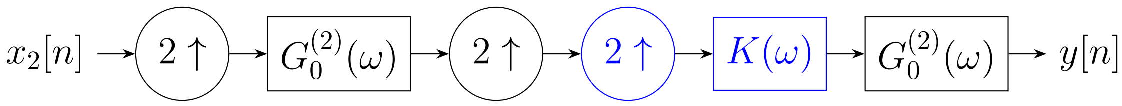

Figure 0(i): Applying the Noble's upsampling identity to the one chain on the synthesis side

From the corresponding Noble's identity, we know that in order for schemes depicted on Figure 0(h) and Figure 0(i) to be equivalent, the following should hold, $K(\omega) = G_1^{(2)}(2\omega)$. And this relationship is equivalent to inserting 1 zero between each 2 successive values of the filter $g_1^{(2)[n]}$ on the synthesis side of the 2-channel QMF filter bank, that is $k[n] = \sum_{k=-\infty}^{+\infty} g_1^{(2)}[k]\delta[n-2k] \triangleq g_{10}^{(2)}[n]$. We can also replace 2 successive upsamplers with one taking into account the fact that the new upsampler that upsamples by the factor of $2 \times 2 = 4$, and we obtain the following scheme.

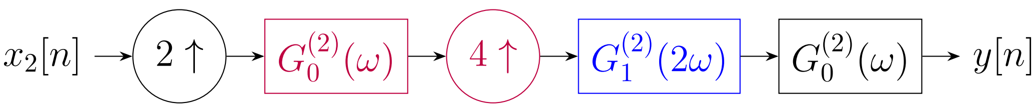

Figure 0(j): Applying the Noble's upsampling identity to the next stage of the chain on the synthesis side

Now we can use the Noble's upsampling identity for the part marked in purple, then we end up with the result depicted on Figure 0(k).

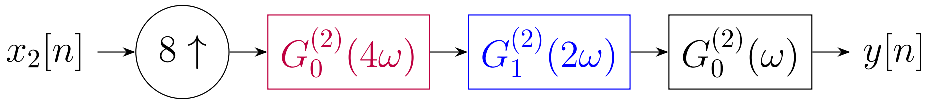

Figure 0(k): The resulting one chain of the synthesis side of the uniform filter bank equivalent to the tree-structured filter bank

Here $G_0^{(2)}(4\omega)$ corresponds to $g_{000}[n] \triangleq \sum_{k=-\infty}^{+\infty} g_0^{(2)}[k]\delta[n-4k]$, that is the filter $g_0^{(2)}[n]$ of the synthesis side of the 2-channel quadrature mirror filter (QMF) with 3 zero inserts between each 2 successive values of $g_0^{(2)}[n]$. The upsampler that upsamples by the factor of 8 is the result of combining 2 successive upsamplers one of which upsamples by the factor of 4 and another one upsamples by the factor of 2.

Thus, we have expressed one chain of the synthesis side of the tree-structured filter bank via the equivalent uniform filter bank. Taking the commutative property of multiplication, we get

$$G_2(\omega) = G_0^{(2)}(\omega) \cdot G_1^{(2)}(2\omega) \cdot G_0^{(2)}(4\omega)$$

In the discrete time domain, we have the following structure.

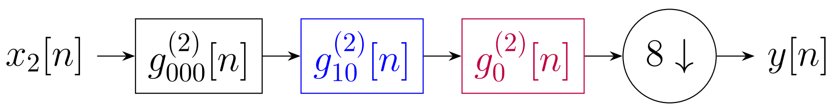

Figure 0(l): The resulting one chain of the synthesis side of the uniform filter bank equivalent to the tree-structured filter bank in the DT domain

Finally, using the fact that for the synthesis side of the 2-channel QMF filter bank $g_0^{(2)}[n] = h_0^{(2)}[n]$ and $g_1^{(2)}[n] = -h_1^{(2)}[n]$, we get

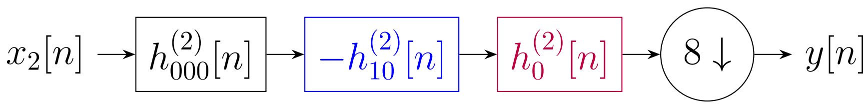

Figure 0(m): The resulting one chain of the synthesis side of the uniform filter bank in the DT domain expressed via $h_0^{(2)}[n]$ and $h_1^{(2)}[n]$

Thus, we get the following relationships for the synthesis side for all 8 chains.

$$
\begin{aligned}
g_0[n] = g_0^{(2)}[n] * g_{00}^{(2)}[n] * g_{000}^{(2)}[n] &\overset{DTFT}{\rightarrow} G_0(\omega) = G_0^{(2)}(\omega) \cdot G_0^{(2)}(2\omega) \cdot G_0^{(2)}(4\omega) \\
g_1[n] = g_0^{(2)}[n] * g_{00}^{(2)}[n] * g_{100}^{(2)}[n] &\overset{DTFT}{\rightarrow} G_1(\omega) = G_0^{(2)}(\omega) \cdot G_0^{(2)}(2\omega) \cdot G_1^{(2)}(4\omega) \\
g_2[n] = g_0^{(2)}[n] * g_{10}^{(2)}[n] * g_{000}^{(2)}[n] &\overset{DTFT}{\rightarrow} G_2(\omega) = G_0^{(2)}(\omega) \cdot G_1^{(2)}(2\omega) \cdot G_0^{(2)}(4\omega) \\
g_3[n] = g_0^{(2)}[n] * g_{10}^{(2)}[n] * g_{100}^{(2)}[n] &\overset{DTFT}{\rightarrow} G_3(\omega) = G_0^{(2)}(\omega) \cdot G_1^{(2)}(2\omega) \cdot G_1^{(2)}(4\omega) \\
g_4[n] = g_1^{(2)}[n] * g_{00}^{(2)}[n] * g_{000}^{(2)}[n] &\overset{DTFT}{\rightarrow} G_4(\omega) = G_1^{(2)}(\omega) \cdot G_0^{(2)}(2\omega) \cdot G_0^{(2)}(4\omega) \\
g_5[n] = g_1^{(2)}[n] * g_{00}^{(2)}[n] * g_{100}^{(2)}[n] &\overset{DTFT}{\rightarrow} G_5(\omega) = G_1^{(2)}(\omega) \cdot G_0^{(2)}(2\omega) \cdot G_1^{(2)}(4\omega) \\
g_6[n] = g_1^{(2)}[n] * g_{10}^{(2)}[n] * g_{000}^{(2)}[n] &\overset{DTFT}{\rightarrow} G_6(\omega) = G_1^{(2)}(\omega) \cdot G_1^{(2)}(2\omega) \cdot G_0^{(2)}(4\omega) \\
g_7[n] = g_1^{(2)}[n] * g_{10}^{(2)}[n] * g_{100}^{(2)}[n] &\overset{DTFT}{\rightarrow} G_7(\omega) = G_1^{(2)}(\omega) \cdot G_1^{(2)}(2\omega) \cdot G_1^{(2)}(4\omega)
\end{aligned}
$$

Also, if $g_0^{(2)}[n] = h_0^{(2)}[n]$ and $g_1^{(2)}[n] = -h_1^{(2)}[n]$, then $G_0^{(2)}(\omega) = H_0^{(2)}(\omega)$ and $G_1^{(2)}(\omega) = -H_1^{(2)}(\omega)$. So we obtain the following relationship between the filters on both sides.

$$
\begin{aligned}
G_0(\omega) &= H_0(\omega) \\
G_1(\omega) &= -H_1(\omega) \\
G_2(\omega) &= -H_2(\omega) \\
G_3(\omega) &= H_3(\omega) \\
G_4(\omega) &= -H_4(\omega) \\
G_5(\omega) &= H_5(\omega) \\
G_6(\omega) &= H_6(\omega) \\
G_7(\omega) &= -H_7(\omega)
\end{aligned}
$$

1 Matlab Calculations

1.1 Part (A)

In part (A), we considered the system with the following impulse responses $h_0^{(2)}[n] = \frac{1}{\sqrt{2}}\{1, 1\}$ (the low-pass part) and $h_1^{(2)}[n] = \frac{1}{\sqrt{2}}\{1, -1\}$ (the high-pass part). The results are presented below (see Figure 1(a), Table 1, Figure 1(b), Figure 1(c)).

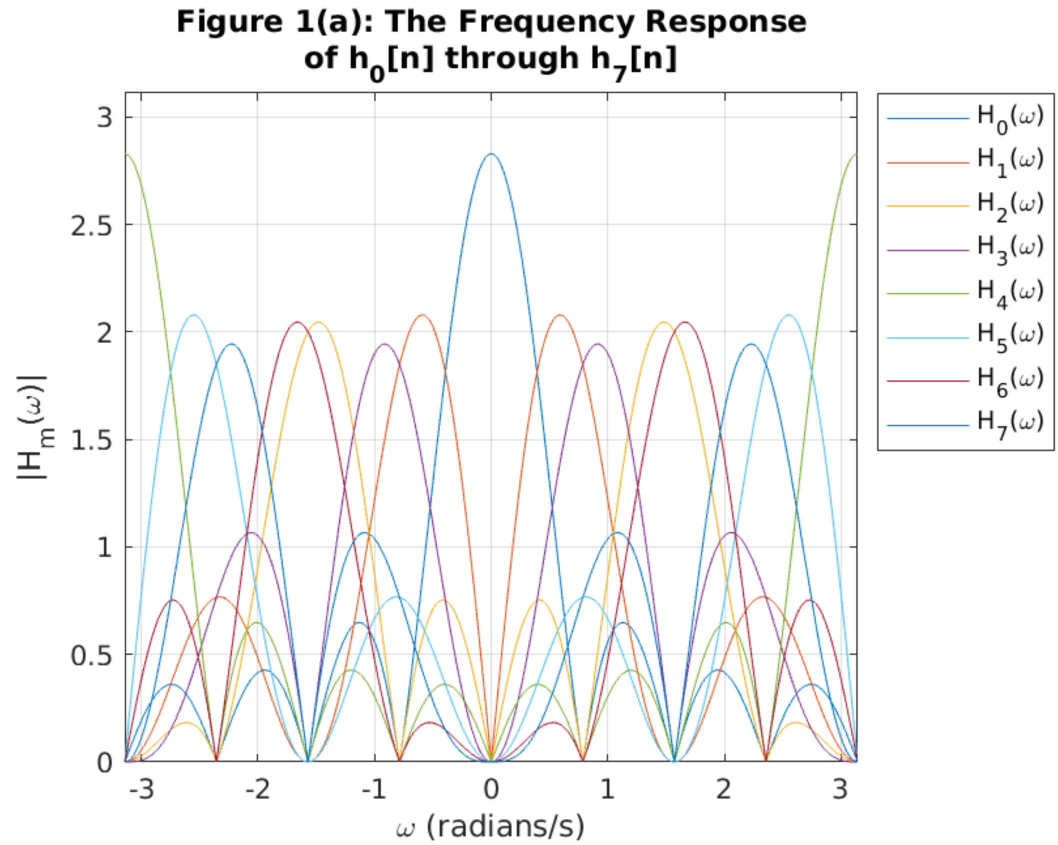

Figure 1(a): The plot of the corresponding DTFTs $H_m(\omega)$, $m = 0, \ldots, 7$ for $h_0^{(2)}[n] = \frac{1}{\sqrt{2}}\{1, 1\}$ and $h_1^{(2)}[n] = \frac{1}{\sqrt{2}}\{1, -1\}$

	$H_0(\omega)$	$H_1(\omega)$	$H_2(\omega)$	$H_3(\omega)$	$H_4(\omega)$	$H_5(\omega)$	$H_6(\omega)$	$H_7(\omega)$
$H_0(\omega)$	1.0000	0	0	0	0	0	0	0
$H_1(\omega)$	0	1.0000	0	0	0	0	0	0
$H_2(\omega)$	0	0	1.0000	0	0	0	0	0
$H_3(\omega)$	0	0	0	1.0000	0	0	0	0
$H_4(\omega)$	0	0	0	0	1.0000	0	0	0
$H_5(\omega)$	0	0	0	0	0	1.0000	0	0
$H_6(\omega)$	0	0	0	0	0	0	1.0000	0
$H_7(\omega)$	0	0	0	0	0	0	0	1.0000

Table 1: The values of the 8×8 matrix $\mathbf{HH}^H$ for $h_0^{(2)}[n] = \frac{1}{\sqrt{2}}\{1, 1\}$ and $h_1^{(2)}[n] = \frac{1}{\sqrt{2}}\{1, -1\}$

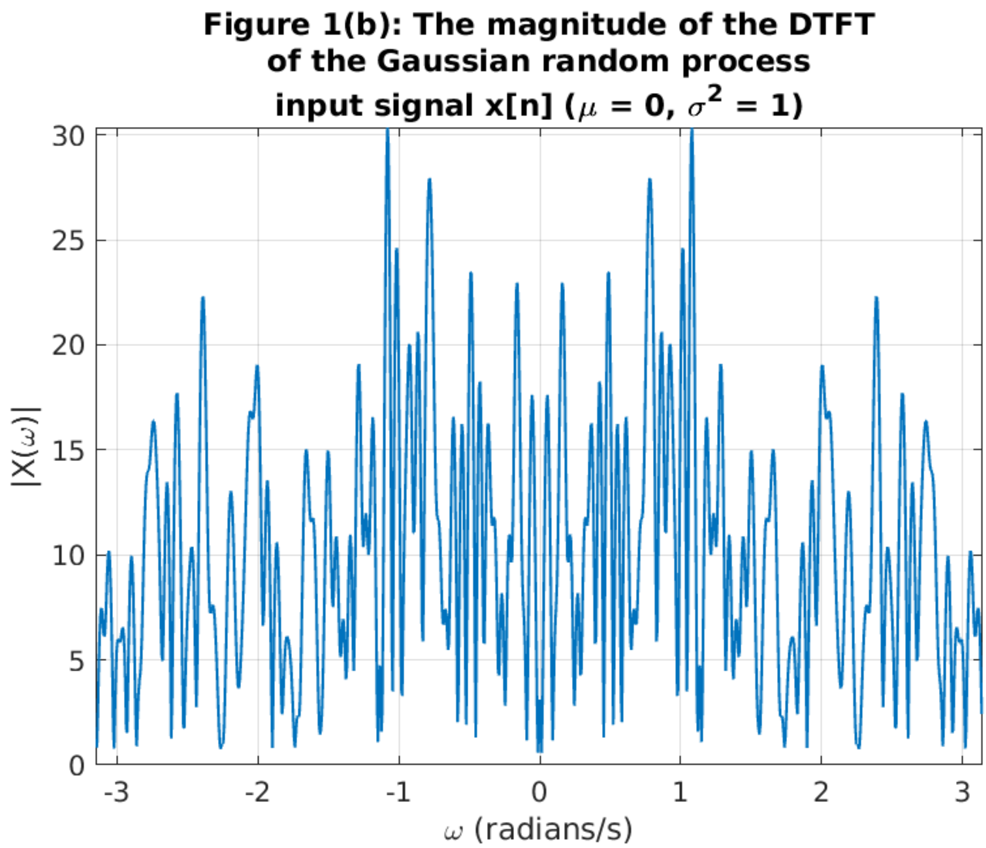

Figure 1(b): The plot of the magnitude of the DTFT of the Gaussian random process input signal ($h_0^{(2)}[n] = \frac{1}{\sqrt{2}}\{1, 1\}$ and $h_1^{(2)}[n] = \frac{1}{\sqrt{2}}\{1, -1\}$)

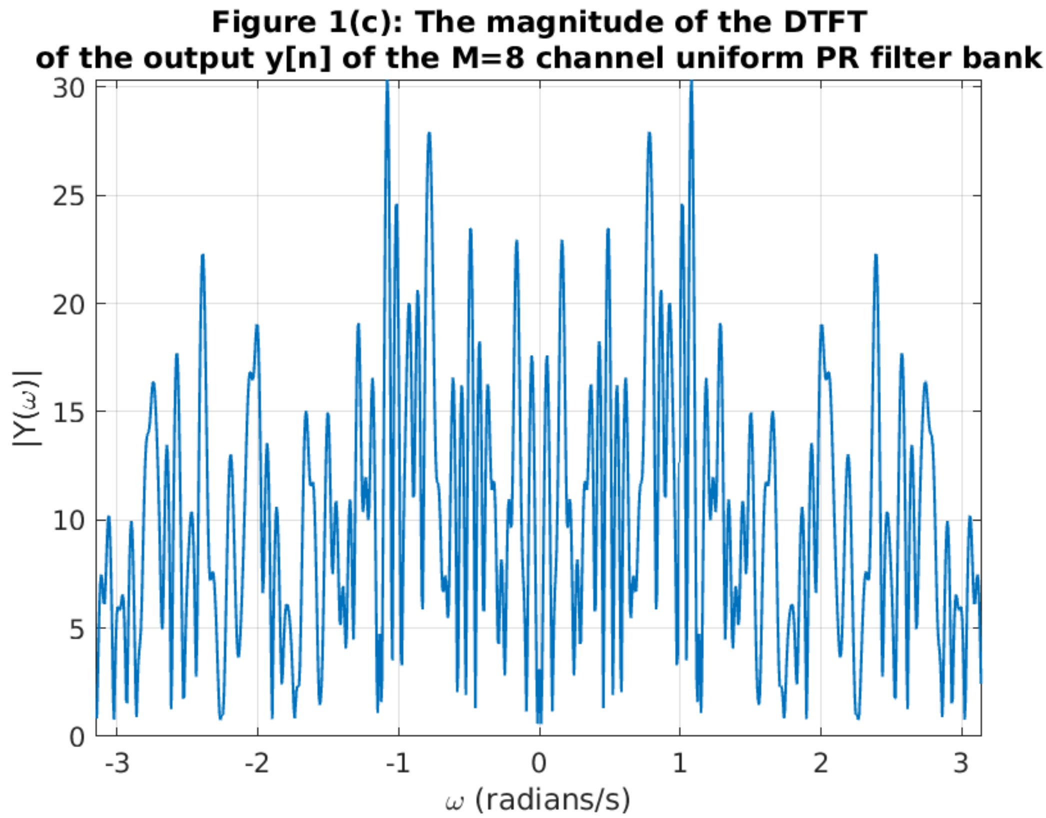

Figure 1(c): The plot of the magnitude of the DTFT of the output of the $M = 8$ channel uniform PR filter bank ($h_0^{(2)}[n] = \frac{1}{\sqrt{2}}\{1, 1\}$ and $h_1^{(2)}[n] = \frac{1}{\sqrt{2}}\{1, -1\}$)

1.2 Part (B)

In part (B), calculations for the following impulse response

$$h_{sr}[n] = \sqrt{2}\left\{\frac{2\beta\cos[(1+\beta)\pi(n+.5)/2]}{\pi[1-4\beta^2(n+.5)^2]} + \frac{\beta\sin[(1-\beta)\pi(n+.5)/2]}{\pi[(n+.5)-4\beta^2(n+.5)^3]}\right\}$$

were performed for $\beta = 0.35$, and $h_0^{(2)}[n] = h_{sr}[n-16]$, $h_1^{(2)}[n] = (-1)^n h_0^{(2)}[n]$, $n = -16, \ldots, 1, \ldots, 15$.

The results are presented below (see Figure 2(a), Table 2, Figure 2(b), Figure 2(c)).

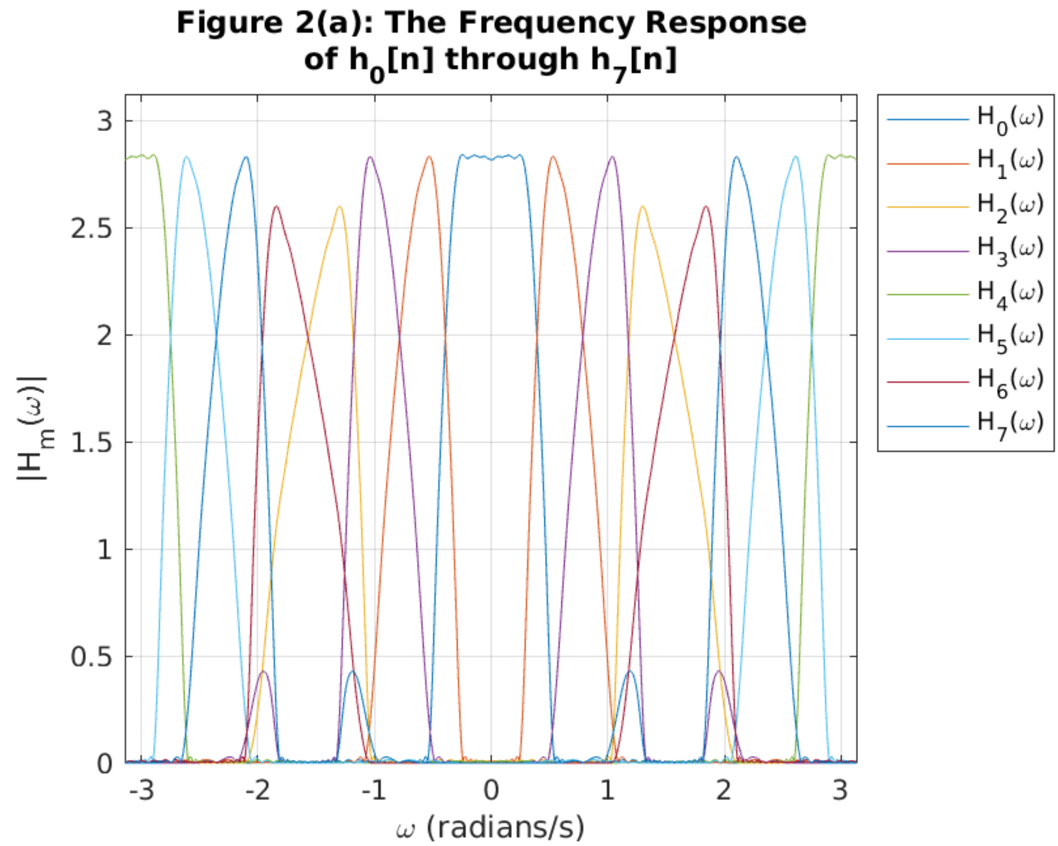

Figure 2(a): The plot of the corresponding DTFTs $H_m(\omega)$, $m = 0, \ldots, 7$ for $h_{sr}[n] = \sqrt{2}\left\{\frac{2\beta\cos[(1+\beta)\pi(n+.5)/2]}{\pi[1-4\beta^2(n+.5)^2]} + \frac{\beta\sin[(1-\beta)\pi(n+.5)/2]}{\pi[(n+.5)-4\beta^2(n+.5)^3]}\right\}$, $\beta = 0.35$ $h_0^{(2)}[n] = h_{sr}[n-16]$ and $h_1^{(2)}[n] = (-1)^n h_0^{(2)}[n]$, $n = -16, \ldots, 1, \ldots, 15$

	$H_0(\omega)$	$H_1(\omega)$	$H_2(\omega)$	$H_3(\omega)$	$H_4(\omega)$	$H_5(\omega)$	$H_6(\omega)$	$H_7(\omega)$
$H_0(\omega)$	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	-0.0000	-0.0000
$H_1(\omega)$	0.0000	0.9999	-0.0000	0.0000	0.0000	0.0000	-0.0000	-0.0000
$H_2(\omega)$	0.0000	-0.0000	1.0002	0.0000	-0.0000	-0.0000	0.0000	-0.0000
$H_3(\omega)$	0.0000	0.0000	0.0000	0.9995	-0.0000	-0.0000	-0.0000	-0.0000
$H_4(\omega)$	0.0000	0.0000	-0.0000	-0.0000	1.0000	0.0000	0.0000	0.0000
$H_5(\omega)$	0.0000	0.0000	-0.0000	-0.0000	0.0000	0.9999	-0.0000	0.0000
$H_6(\omega)$	-0.0000	-0.0000	0.0000	-0.0000	0.0000	-0.0000	1.0002	0.0000
$H_7(\omega)$	-0.0000	-0.0000	-0.0000	-0.0000	0.0000	0.0000	0.0000	0.9995

Table 2: The values of the 8×8 matrix $\mathbf{HH}^H$ ($h_0^{(2)}[n] = h_{sr}[n-16]$ and $h_1^{(2)}[n] = (-1)^n h_0^{(2)}[n]$, $n = -16, \ldots, 1, \ldots, 15$)

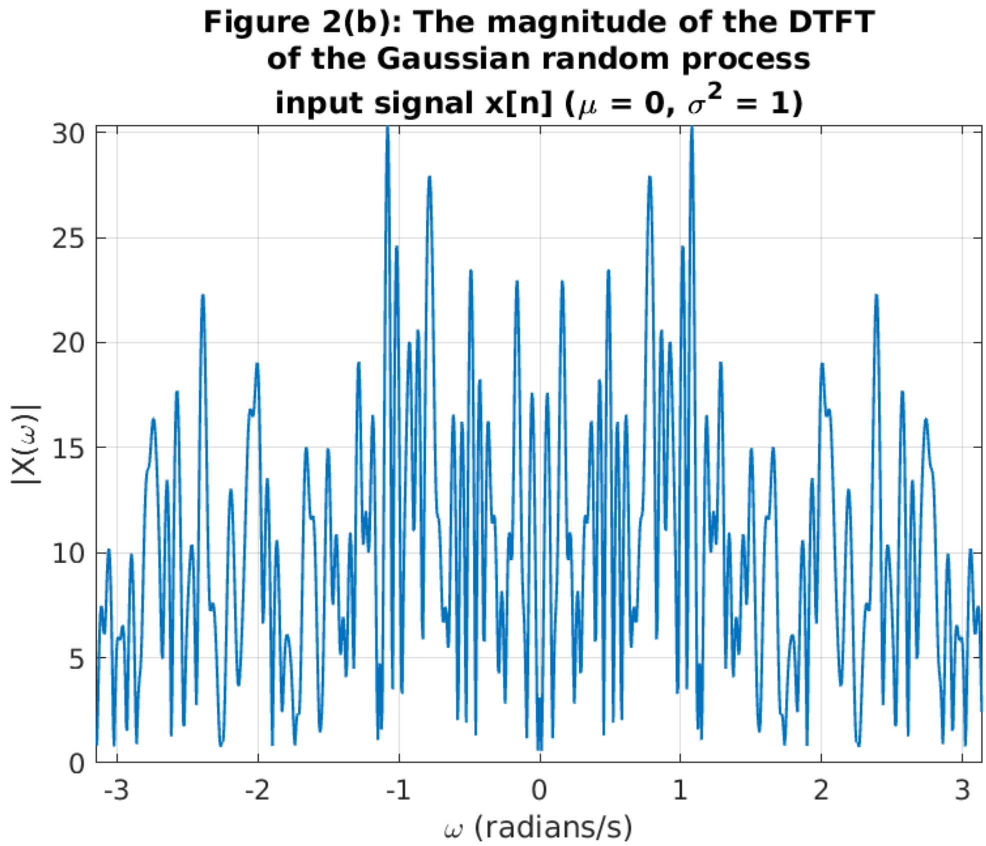

Figure 2(b): The plot of the magnitude of the DTFT of the Gaussian random process input signal ($h_0^{(2)}[n] = h_{sr}[n-16]$ and $h_1^{(2)}[n] = (-1)^n h_0^{(2)}[n]$, $n = -16, \ldots, 1, \ldots, 15$)

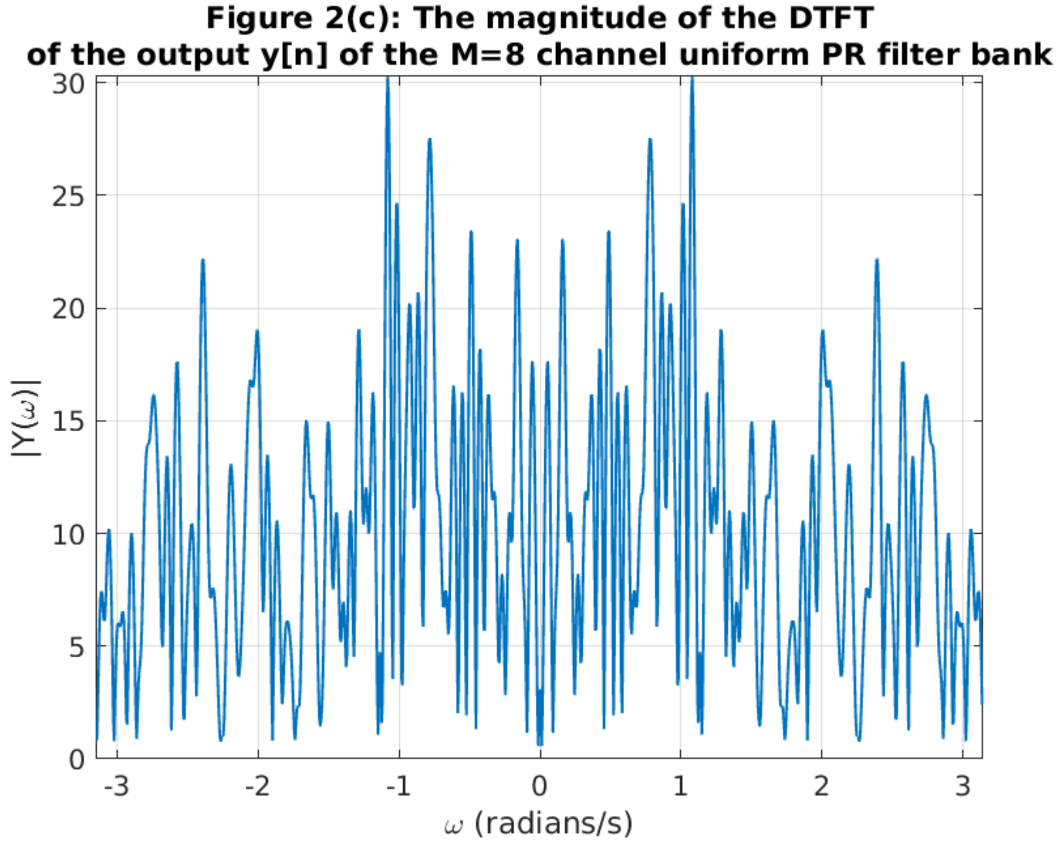

Figure 2(c): The plot of the magnitude of the DTFT of the output of the $M = 8$ channel uniform PR filter bank ($h_0^{(2)}[n] = h_{sr}[n-16]$ and $h_1^{(2)}[n] = (-1)^n h_0^{(2)}[n]$, $n = -16, \ldots, 1, \ldots, 15$)

1.3 Part (C)

In part (C), calculations for the following impulse response were performed

$$h_{sr}[n] = \sqrt{2}\left\{\frac{2\beta\cos[(1+\beta)\pi(n+.5)/2]}{\pi[1-4\beta^2(n+.5)^2]} + \frac{\beta\sin[(1-\beta)\pi(n+.5)/2]}{\pi[(n+.5)-4\beta^2(n+.5)^3]}\right\}$$

for $\beta = 0.1$, and $h_0^{(2)}[n] = h_{sr}[n-24]$, $h_1^{(2)}[n] = (-1)^n h_0^{(2)}[n]$, $n = -24, \ldots, 1, \ldots, 23$.

The results are presented below (see Figure 3(a), Table 3, Figure 3(b), Figure 3(c))

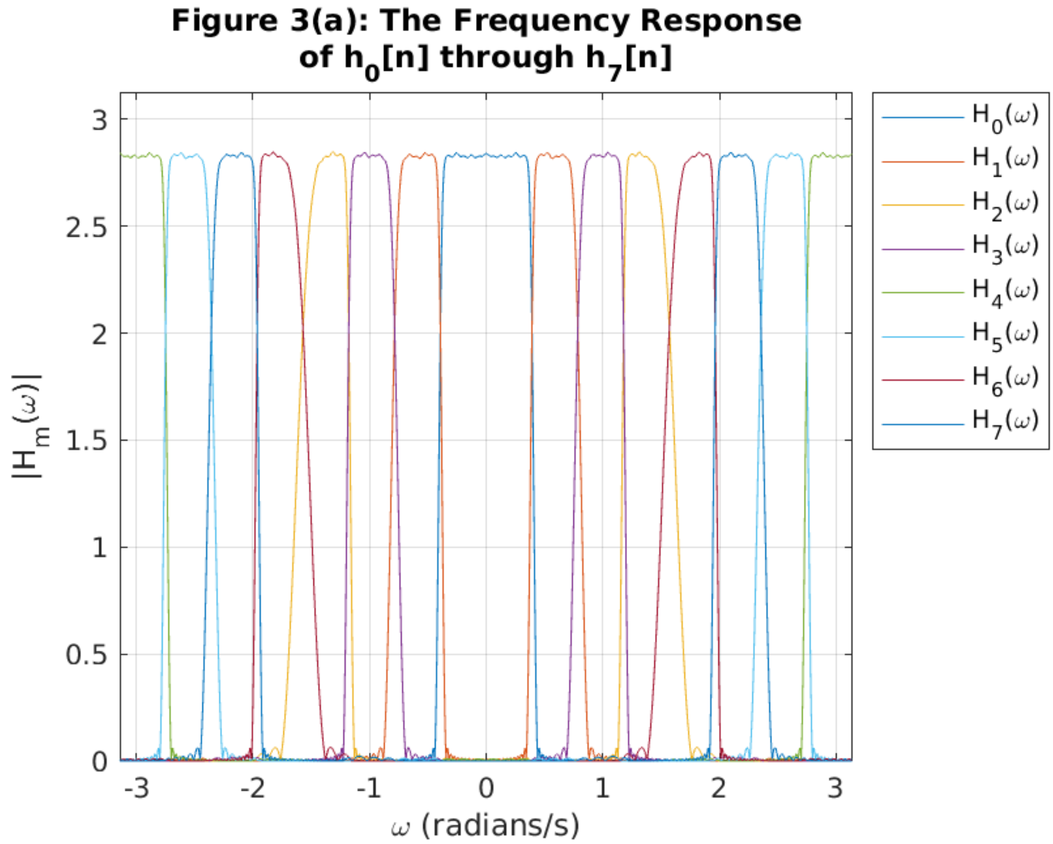

Figure 3(a): The plot of the corresponding DTFTs $H_m(\omega)$, $m = 0, \ldots, 7$ for $h_{sr}[n] = \sqrt{2}\left\{\frac{2\beta\cos[(1+\beta)\pi(n+.5)/2]}{\pi[1-4\beta^2(n+.5)^2]} + \frac{\beta\sin[(1-\beta)\pi(n+.5)/2]}{\pi[(n+.5)-4\beta^2(n+.5)^3]}\right\}$, $\beta = 0.1$, $h_0^{(2)}[n] = h_{sr}[n-24]$ and $h_1^{(2)}[n] = (-1)^n h_0^{(2)}[n]$, $n = -24, \ldots, 1, \ldots, 23$

	$H_0(\omega)$	$H_1(\omega)$	$H_2(\omega)$	$H_3(\omega)$	$H_4(\omega)$	$H_5(\omega)$	$H_6(\omega)$	$H_7(\omega)$
$H_0(\omega)$	0.9999	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	-0.0000
$H_1(\omega)$	0.0000	0.9998	-0.0000	-0.0000	0.0000	-0.0000	-0.0000	0.0000
$H_2(\omega)$	0.0000	-0.0000	0.9998	0.0000	0.0000	-0.0000	-0.0000	-0.0000
$H_3(\omega)$	0.0000	-0.0000	0.0000	0.9991	-0.0000	0.0000	-0.0000	0.0000
$H_4(\omega)$	0.0000	0.0000	0.0000	-0.0000	0.9999	0.0000	0.0000	0.0000
$H_5(\omega)$	0.0000	-0.0000	-0.0000	0.0000	0.0000	0.9998	-0.0000	-0.0000
$H_6(\omega)$	0.0000	-0.0000	-0.0000	-0.0000	0.0000	-0.0000	0.9998	0.0000
$H_7(\omega)$	-0.0000	0.0000	-0.0000	0.0000	0.0000	-0.0000	0.0000	0.9991

Table 3: The values of the 8×8 matrix $\mathbf{HH}^H$ ($h_0^{(2)}[n] = h_{sr}[n-23]$ and $h_1^{(2)}[n] = (-1)^n h_0^{(2)}[n]$, $n = -24, \ldots, 1, \ldots, 23$)

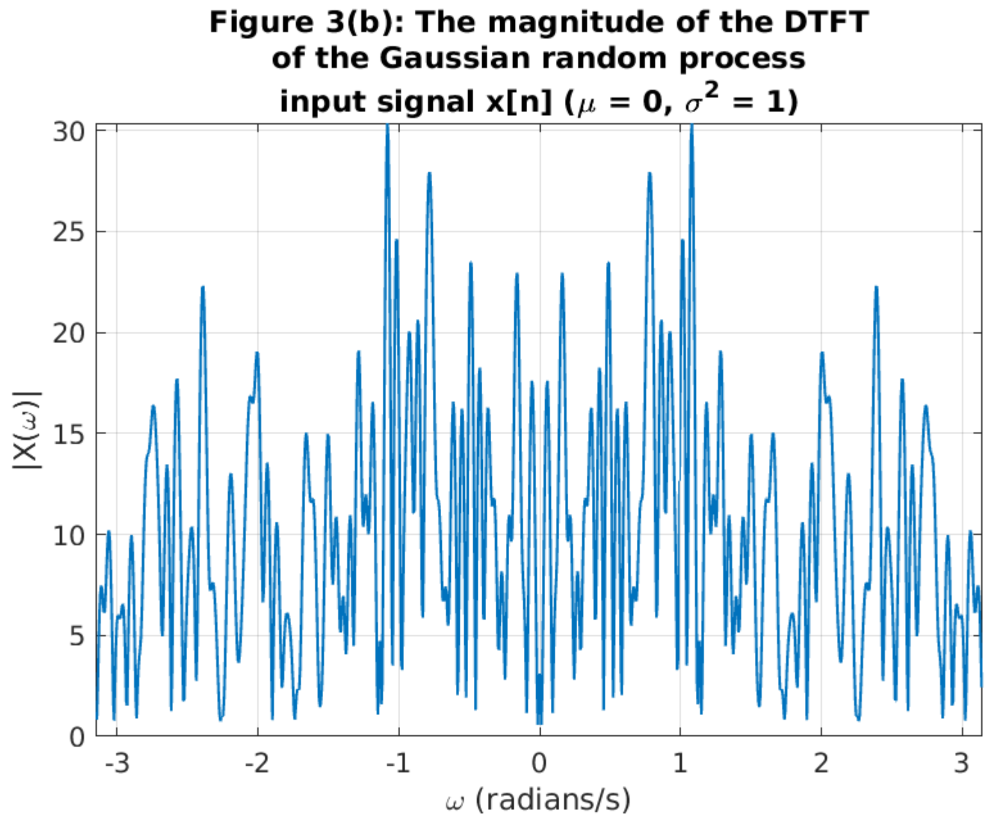

Figure 3(b): The plot of the magnitude of the DTFT of the Gaussian random process input signal ($h_0^{(2)}[n] = h_{sr}[n-24]$ and $h_1^{(2)}[n] = (-1)^n h_0^{(2)}[n]$, $n = -24, \ldots, 1, \ldots, 23$)

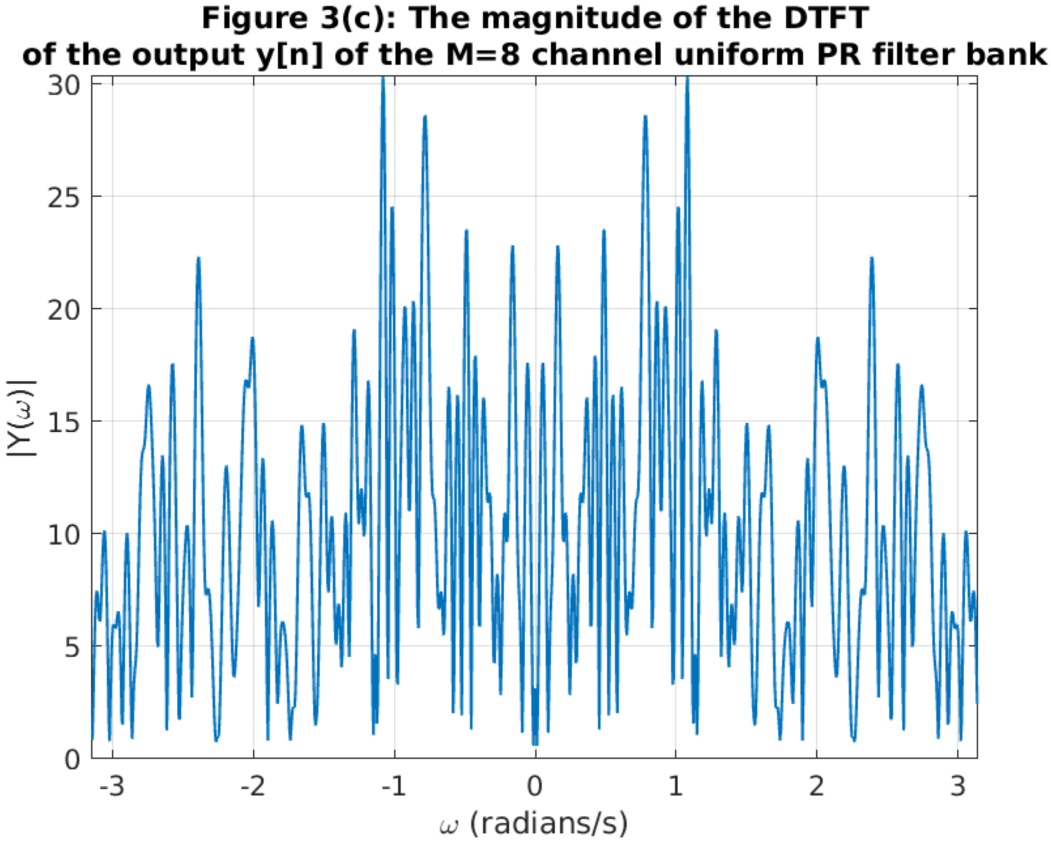

Figure 3(c): The plot of the magnitude of the DTFT of the output of the $M = 8$ channel uniform PR filter bank ($h_0^{(2)}[n] = h_{sr}[n-24]$ and $h_1^{(2)}[n] = (-1)^n h_0^{(2)}[n]$, $n = -24, \ldots, 1, \ldots, 23$)

2 Conclusion

From the obtained results, we can see that the perfect reconstruction filter banks works for any low-pass filter, even if the filter $h_0^{(2)}[n]$, which defines the structure of the filter bank, does not have a flat part in its DTFT spectrum (see the results for Part (A)).

Also, we can see that in all the considered cases filters of the analysis side $H_m(\omega)$ of the uniform filter bank are mutually orthogonal and contain the equal amount of energy.

3 The Matlab code

```
clf
clear all

set(groot,'defaultAxesXGrid','on')
set(groot,'defaultAxesYGrid','on')

M=8; % the number of channels

% Part A
h_mx_a = calc_filters(1);
[H_a, G_a] = uniform_filter_bank(h_mx_a);
disp(H_a*H_a');

% Part B
h_mx_b = calc_filters(16, 0.35);
[H_b, G_b] = uniform_filter_bank(h_mx_b);
disp(H_b*H_b');

% Part C
h_mx_c = calc_filters(24, 0.1);
[H_c, G_c] = uniform_filter_bank(h_mx_c);
disp(H_c*H_c')
```

```
% Generate a random Gaussian process input signal
x=randn(1,128);

% Number of DFT points
Nfft=1024;

% Collect all the filters
H_filters = {H_a, H_b, H_c};
G_filters = {G_a, G_b, G_c};

for k=1:numel(H_filters)

    % Get the filters for parts A, B, and C
    H = H_filters{k};
    G = G_filters{k};

    % Calculate the output of the M=8 channel
       uniform PR filter bank
    y = calc_output(H,G,M,x);

    domega=2*pi/Nfft;
    omega=-pi:domega:pi-domega;

    % Plot the DTFTs of the analysis side
    fig1=figure(1+3*(k-1));

    for ind=1:M
        hf(ind,:)=abs(fftshift(fft(H(ind,:), Nfft)))
           ;
        hold on
        plot(omega,hf(ind,:));
    end

    box on;
```

```

    axis([-pi pi 0 1.1*max(hf(1,:))])
    legend("H_{0}(\omega)","H_{1}(\omega)", ...
        "H_{2}(\omega)","H_{3}(\omega)", ...
        "H_{4}(\omega)","H_{5}(\omega)", ...
        "H_{6}(\omega)","H_{7}(\omega)", 'Location', 'northeastoutside');
    title({"Figure "+k+"(a): The Frequency Response ", "of h_{0}[n] through h_{7}[n]"});
    xlabel("\omega (radians/s)");
    ylabel("|H_{m}(\omega)|");
    saveas(fig1, sprintf('fig%d.png', 1+3*(k-1)));

    % Compute the DTFTs of the Gaussian RP input signal and the output of
    % the uniform PR filter bank
    yf1=abs(fftshift(fft(x,Nfft)));
    yf2=abs(fftshift(fft(y,Nfft)));

    % Plot the DTFT of the input signal
    fig2=figure(2+3*(k-1));
    plot(omega,yf1,'Linewidth',1)
    axis([-pi pi 0 max(yf1)])
    title({"Figure "+k+"(b): The magnitude of the DTFT", "of the Gaussian random process", "input signal x[n] (\mu = 0, \sigma^2 = 1)"});
    xlabel("\omega (radians/s)");
    ylabel("|X(\omega)|");
    saveas(fig2, sprintf('fig%d.png', 2+3*(k-1)));

    % Plot the DTFT of the output of the uniform PR filter bank
    fig3=figure(3+3*(k-1));
    plot(omega,yf2,'Linewidth',1)
    axis([-pi pi 0 max(yf2)])
```

```
    title({"Figure "+k+"(c): The magnitude of the DTFT", "of the output y[n] of the M=8 channel uniform PR filter bank"});
    xlabel("\omega (radians/s)");
    ylabel("|Y(\omega)|");
    saveas(fig3, sprintf('fig%d.png', 3+3*(k-1)));

end

function h_mx = calc_filters(N, beta)
    n=-N:(N-1);

    if exist('beta','var')
        n=n+0.5;
        h=2*beta*cos((1+beta)*pi*n/2)./(pi*(1-4*beta^2*n.^2));
        h=h+sin((1-beta)*pi*n/2)./(pi*(n-4*beta^2*n.^3));
        h=h*sqrt(2);
    else
        h=[1 1]/sqrt(2);
    end

    % Get all the necessary filters for building the system
    h0=h;
    h1=(-1).^(0:(length(n)-1)).*h;

    h00=zeros(1,2*length(h));
    h10=h00;

    h00(1,1:2:length(h00))=h0;
    h10(1,1:2:length(h10))=h1;

    h000=zeros(1,4*length(h));
```

```
    h100=h000;

    h000(1,1:4:length(h000))=h0;
    h100(1,1:4:length(h100))=h1;

    % A cell array containing all the necessary
      filters
    h_mx{1}= h0;
    h_mx{2}= h1;
    h_mx{3}= h00;
    h_mx{4}= h10;
    h_mx{5}= h000;
    h_mx{6}= h100;

end

function [H, G] = uniform_filter_bank(h_mx)

    h0=h_mx{1};
    h1=h_mx{2};
    h00=h_mx{3};
    h10=h_mx{4};
    h000=h_mx{5};
    h100=h_mx{6};

    % Analysis side
    H(1,:)=conv(conv(h0,h00), h000);
    H(2,:)=conv(conv(h0,h00), h100);
    H(3,:)=conv(conv(h0,h10), h000);
    H(4,:)=conv(conv(h0,h10), h100);
    H(5,:)=conv(conv(h1,h00), h000);
    H(6,:)=conv(conv(h1,h00), h100);
    H(7,:)=conv(conv(h1,h10), h000);
    H(8,:)=conv(conv(h1,h10), h100);

```

```
    % Synthesis side
    G(1,:) = H(1,:);
    G(2,:) = -H(2,:);
    G(3,:) = -H(3,:);
    G(4,:) = H(4,:);
    G(5,:) = -H(5,:);
    G(6,:) = H(6,:);
    G(7,:) = H(7,:);
    G(8,:) = -H(8,:);

end

function y = calc_output(H,G,M,x)

    for m=1:M
        % Pass x[n] through M filter
        W(m,:)=conv(x,H(m,:));
        % Decimation by the factor of M
        X(m,:)=W(m,1:M:length(W(m,:)));
    end

    for m=1:M
        % Zero inserts
        Z(m,:)=zeros(1,M*length(X(m,:)));
        Z(m,1:M:length(Z(m,:)))=X(m,:);
        % Convolution with synthesis filters
        Y(m,:)=conv(Z(m,:),G(m,:));
    end

    % Get the output y[n]
    y=zeros(1,length(Y(1,:)));

    for m=1:M
        y=y+Y(m,:);
    end
```

```
end
```