

Relatório

Experiência 3 - Como o som é processado em um MP3 player

PSI3531 - Processamento de Sinais Aplicado (2020)

Matheus Bordin Gomes - 9838028

Essa experiência reúne um conjunto de exercícios práticos relativos às temáticas vistas em aula, como *downsampling* e *upsampling*, bancos de filtros multitaxas, tipos de quantização e efeito de mascaramento.

1 - Banco de filtros com dois canais

Esse exercício está implementado no *script* de Matlab "exp3_1.m", entregue em conjunto com esse relatório. Nessa parte da experiência são analisados os efeitos de do processamento com um banco de filtros com dois canais, utilizando como entrada um sinal *chirp* linear.

- a O espectrograma do sinal *chirp* linear gerado e amostrado com $f_a = 8$ KHz pode ser visto na figura 1.

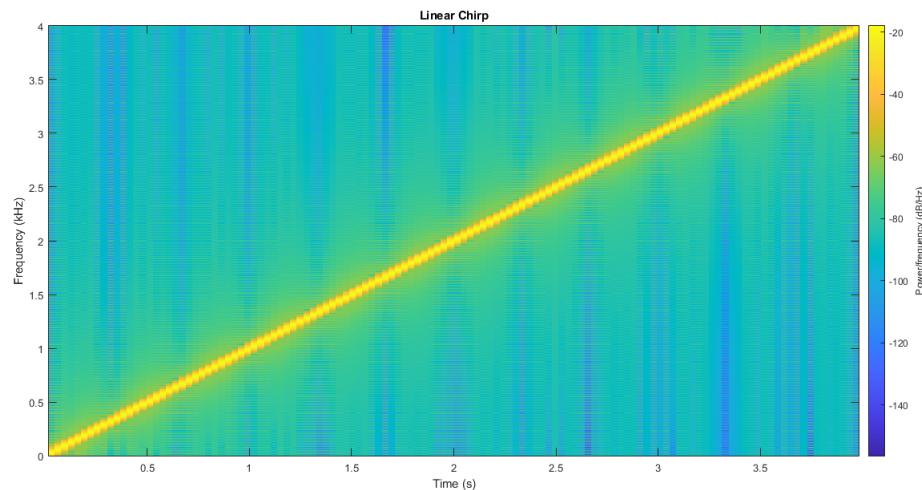


Figura 1. Espectrograma do sinal *chirp* linear gerado e amostrado com a 8 KHz.

- b Ao se realizar *downsampling* em um sinal, é preciso tomar cuidado para evitar *aliasing*. Mesmo que o sinal original tenha sido amostrado respeitando o critério de Nyquist, ele ainda pode ter frequências que sejam maiores do que a frequência de amostragem resultante nesse processo. Ou seja, a condição $F_{max} < \frac{f_s}{2}M$ deve ser satisfeita para garantir uma subamostragem sem *aliasing*.

É por conta desse fato que podemos observar o efeito gerado no espectrograma da figura 2. Temos que a nova taxa de amostragem do sinal é de $f_{s,down} = f_s/M = 4\text{KHZ}$, que coincide com

a frequência máxima do sinal. Logo, há sobreposição de toda a banda entre 2 e 4 kHz com a banda de 0 a 2 kHz.

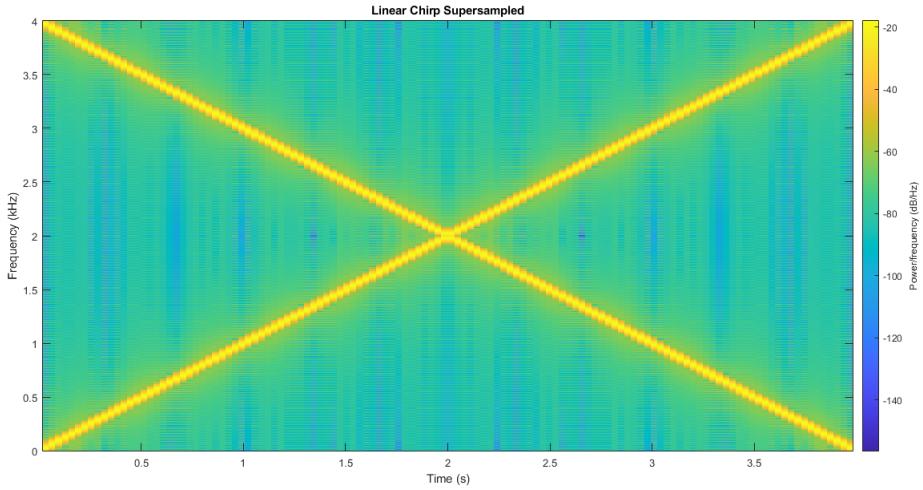


Figura 2. Espectrograma do sinal subamostrado e depois superamostrado com o efeito do *aliasing*.

- c A resposta em frequência do filtro passa-baixas FIR projetado com as funções "firpmord.b" e "firpm.m" do Matlab, que implementam o algoritmo de Parks-McClellan, pode ser vista na [3](#).

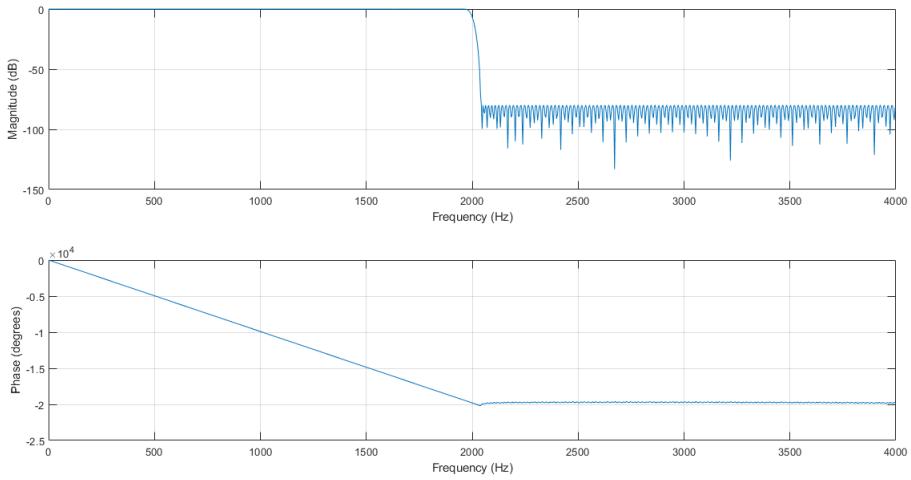


Figura 3. Resposta em frequência (módulo e fase) do filtro passa-baixas FIR projetado.

- d Como o sinal original não foi filtrado antes da subamostragem, o *aliasing* continuou a ocorrer, gerando o rebatimento do espectro. Dessa forma, o efeito da filtragem do sinal superamostrado foi atenuar todas as frequências acima de 2 kHz, porém não conseguiu retirar o rebatimento abaixo de 2 kHz. Assim, a primeira metade do sinal de saída corresponde à primeira metade do sinal original. Já a segunda metade corresponde à segunda metade do sinal original, que possui frequência maior de que 2 kHz, rebatida. O espectrograma pode ser visto na figura [4](#) e o gráfico do sinal filtrado no tempo pode ser visto na figura [5](#).

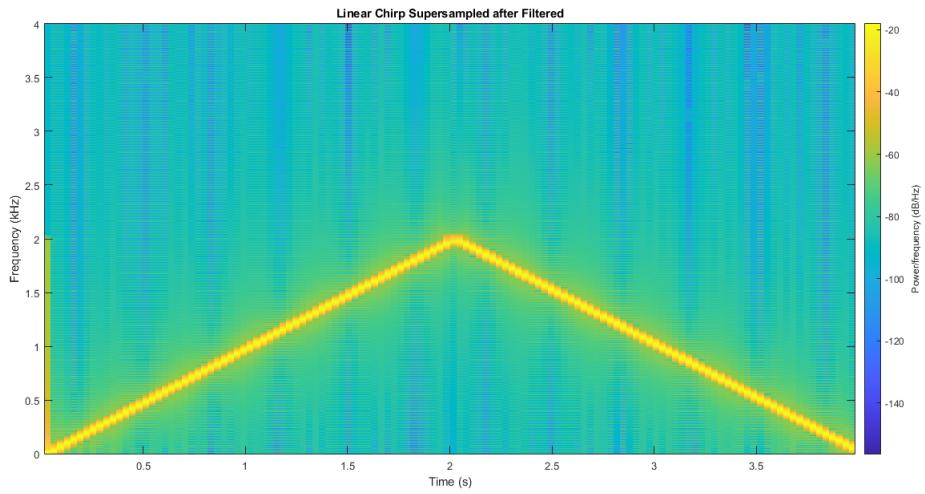


Figura 4. Espectrograma do sinal superamostrado filtrado por um filtro passa-baixas de frequência de corte em 2 kHz.

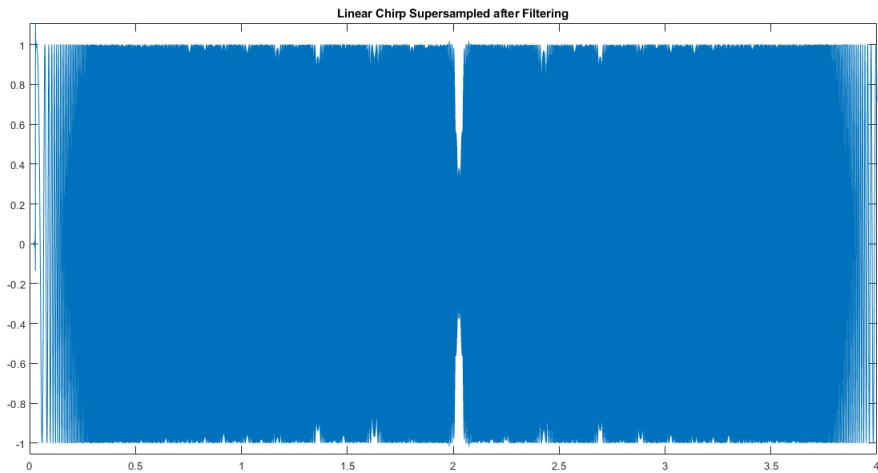


Figura 5. Gráfico no tempo do sinal superamostrado filtrado por um filtro passa-baixas de frequência de corte em 2 kHz.

- e Para evitar o *aliasing* é necessário passar o sinal original por um filtro passa-baixas com frequência de corte igual a $F_c = \frac{F_s}{2M}$. Assim, não haverá *aliasing* no processo de *downsampling*. Além disso, é necessário adicionar um filtro após o processo de *upsampling*, para realizar a interpolação das amostras. Esse filtro será tal que $G_0(e^{j\omega}) = 2 * H_0(e^{j\omega})$. Vale notar que essa solução serve apenas para as frequências menores do que 2 kHz, nesse caso. O espectrograma do sinal de saída obtido com a solução proposta pode ser visto na figura 6.

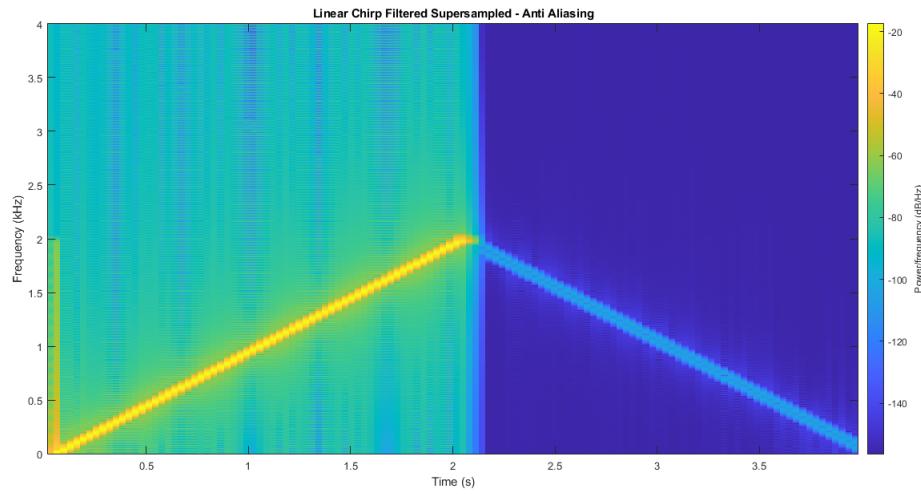


Figura 6. Espectrograma do sinal de saída da solução para restaurar as baixas frequências do sinal sem aliasing.

f Para obter uma solução para a banda de alta frequência, em primeiro lugar é necessário filtrar o sinal por um filtro passa-altas, o qual pode ser obtido do filtro passa-baixas anterior com a relação $H_1(e^{j\omega}) = H_0(e^{j(\omega-\pi)})$. Após o processo de subamostragem e superamostragem, também será necessário passar o sinal obtido por um filtro interpolador tal que $G_1(e^{j\omega}) = -2H_0(e^{j(\omega-\pi)})$. A resposta em frequência do filtro passa-altas obtido do filtro passa-baixas anterior pode ser vista na figura 7. Já o espectrograma do sinal resultante dessa solução pode ser visto na figura 8.

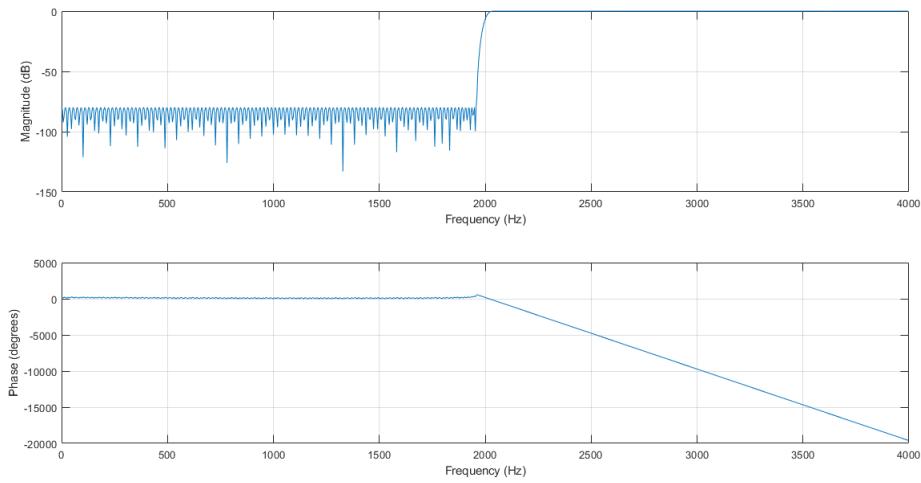


Figura 7. Resposta em frequência do filtro passa-altas obtido do filtro passa-baixa utilizado na solução anterior.

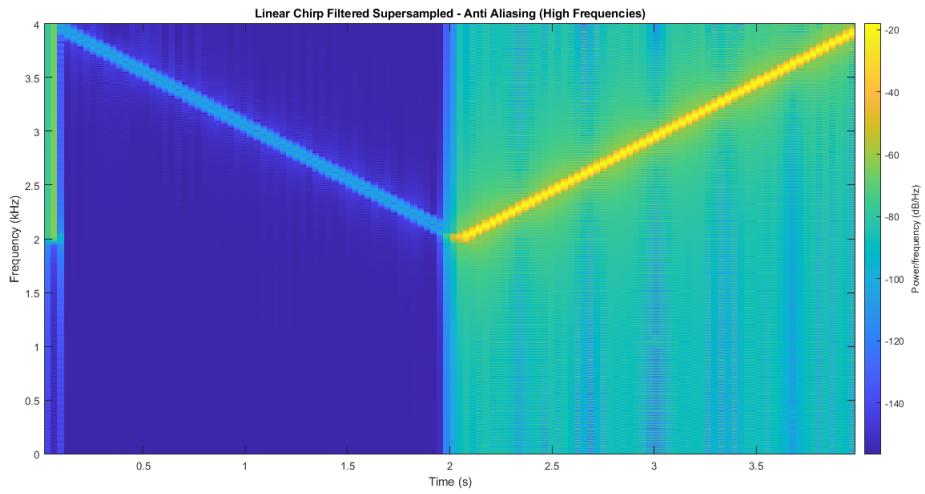


Figura 8. Espectrograma do sinal de saída da solução para a banda de alta frequência de forma a evitar *aliasing*.

g) O espectrograma da soma dos sinais gerados nos itens e) e f) pode ser visto na figura 9, o qual se assemelha muito ao espectrograma do sinal original. Para avaliar melhor o erro na reconstrução do sinal, foi calculada a diferença entre o sinal original e o sinal restaurado, que pode ser vista na figura 10. Observa-se um pico significativo por volta de 2 segundos, provavelmente devido a banda de rejeição dos dois filtros estarem em torno de 2 kHz, frequência nesse instante. Ademais, o erro obtido foi da ordem de 10^{-3} .

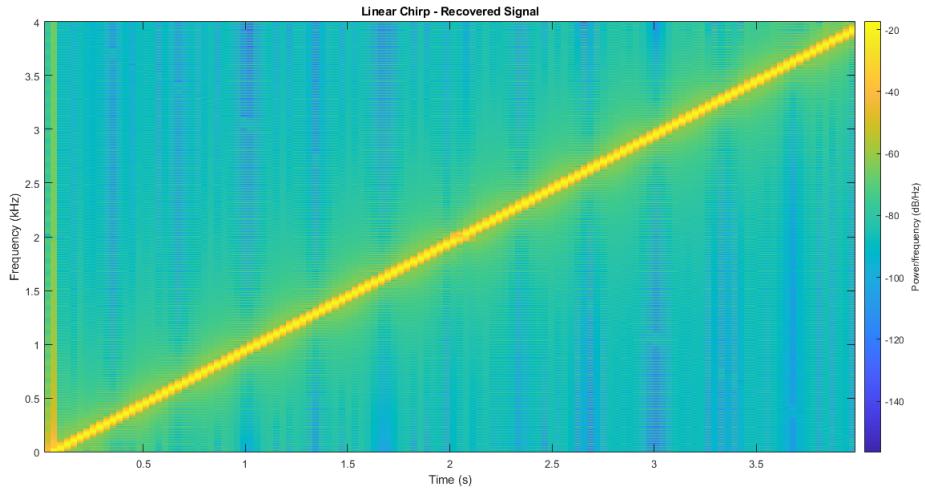


Figura 9. Espectrograma do sinal restaurado.

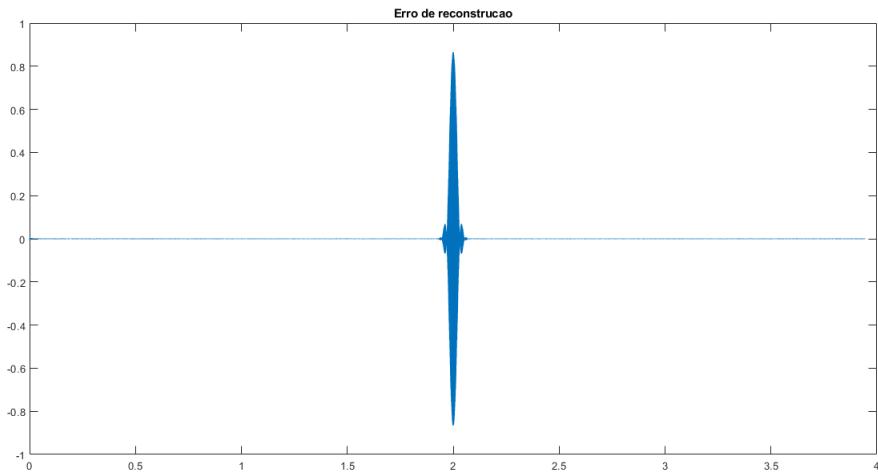


Figura 10. Erro de restauração.

2 - Banco de filtros QMF com dois canais

Esse exercício está implementado no *script* de Matlab "exp3_2.m", entregue em conjunto com esse relatório. Nessa parte da experiência são analisados os efeitos de do processamento com um banco de filtros com dois canais entregue junto com o enunciado, utilizando como entrada o mesmo sinal *chirp* linear do exercício anterior.

- a As repostas em frequência dos filtros do banco "h0_QMF" podem ser vistas nas figuras 11, 12, 13 e 14. Pode-se notar que esses filtros possuem uma rejeição pior do que a dos filtros projetados anteriormente. Além disso, a faixa de transição desses filtros é menos brusca do que nos casos anteriores. Dessa forma, a intersecção entre as bandas dos filtros será maior.

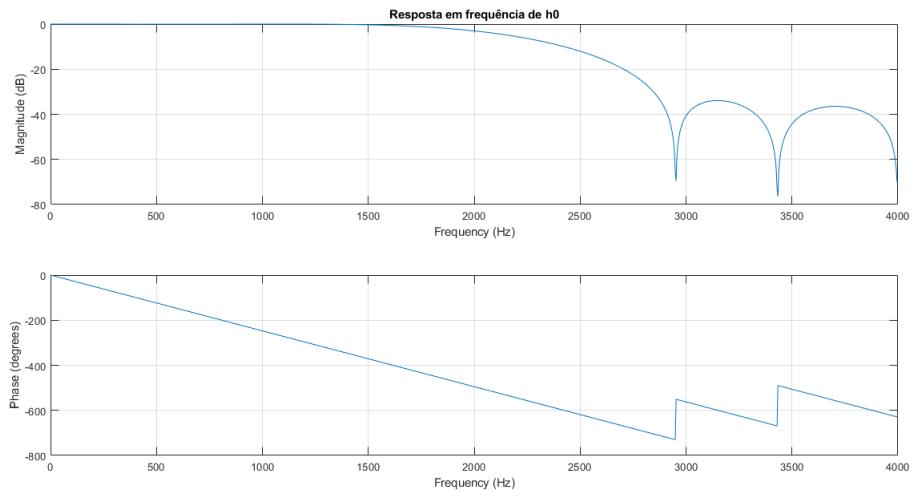


Figura 11. Resposta em frequência do filtro h_0 do banco "h0_QMF".

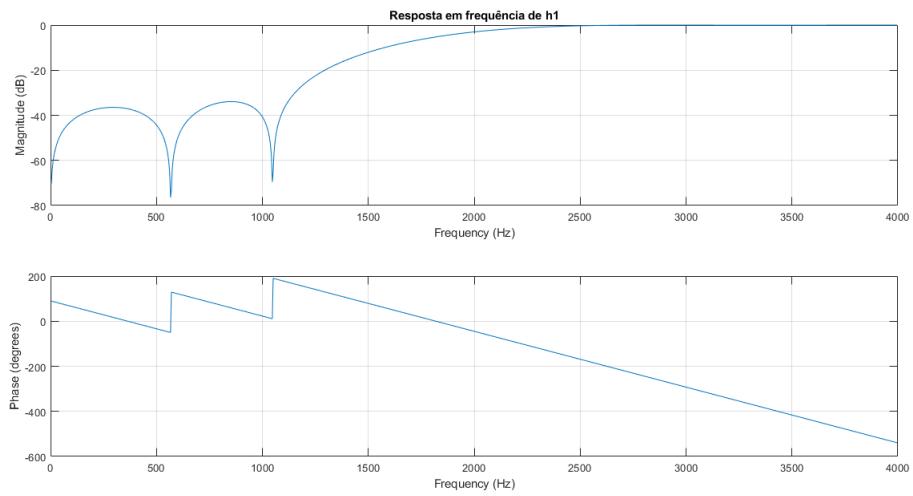


Figura 12. Resposta em frequênciade filtro h_1 do banco "h0_QMF".

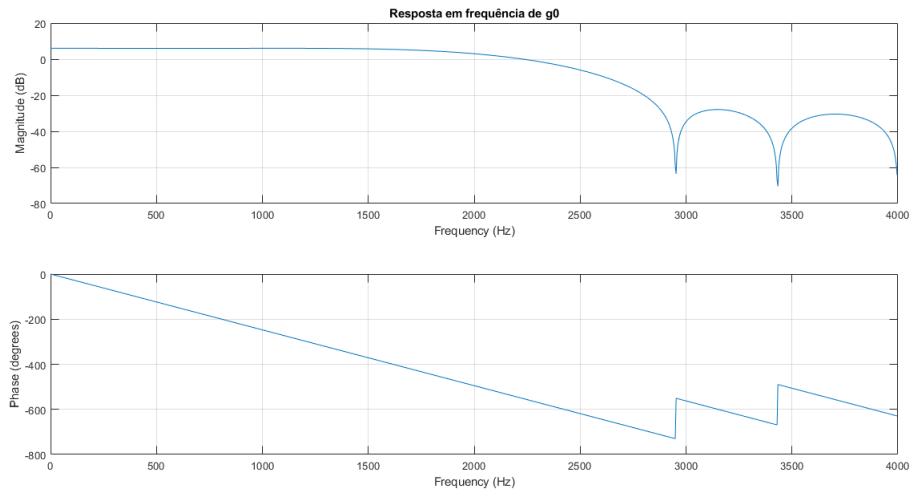


Figura 13. Resposta em frequênciade filtro g_0 do banco "h0_QMF".

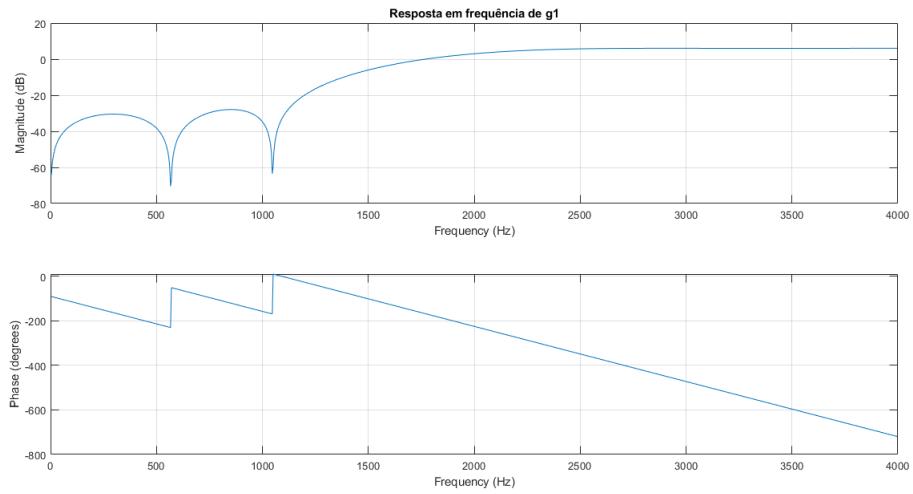


Figura 14. Resposta em frequênciade g_1 do banco "h0_QMF".

b O espectrograma do sinal recuperado desse banco de filtros pode ser visto na figura 15 e o erro de reconstrução pode ser visto na figura 16. Pode-se notar que o erro foi reduzido em toda a banda do sinal e possui uma amplitude máxima menor que $5 * 10^{-3}$. Assim, o erro na faixa de transição entre os dois filtros, que era significativo no item anterior, foi significativamente reduzido.

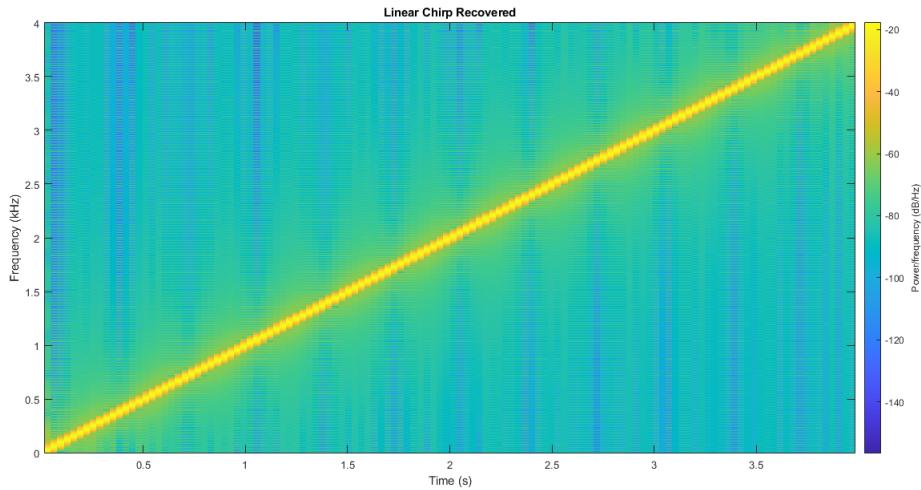


Figura 15. Espectrograma do sinal recuperado do do banco de filtros "h0_QMF".

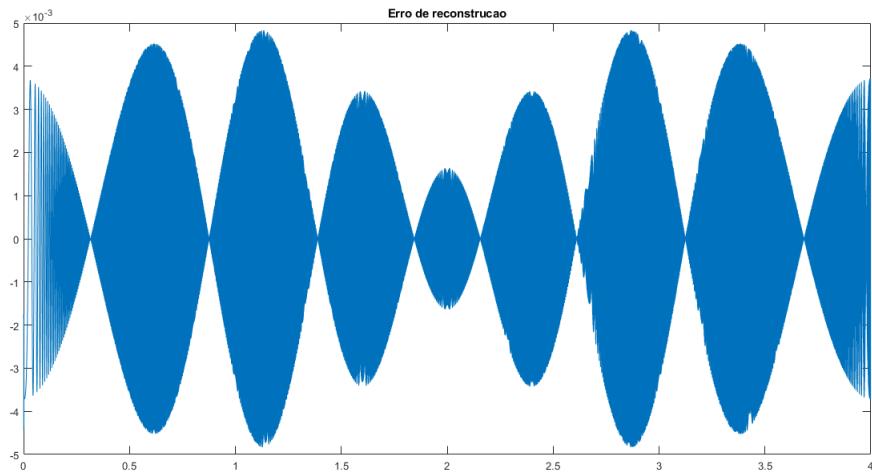


Figura 16. Erro de recuperação do banco de filtros "h0_QMF".

3 - Banco de filtros pseudo-QMF com 32 canais

Esse exercício está implementado nos *scripts* de Matlab "exp3_3.m" e "exp3_3d.m", entregues em conjunto com esse relatório. Nessa parte da experiência são analisados os efeitos de do processamento com um banco de filtros pseudo-QMF de 32 canais, utilizando como entrada um sinal *chirp* linear.

- a A resposta em frequência dos filtros H do banco de filtros "Analise_Sintese32" pode ser vista na figura 17. É notável a intersecção entre as bandas de rejeição dos filtros, além do ganho para compensar a perda de potência na subamostragem. Já o espectrograma do novo sinal *chirp* linear pode ser visto na figura 18.

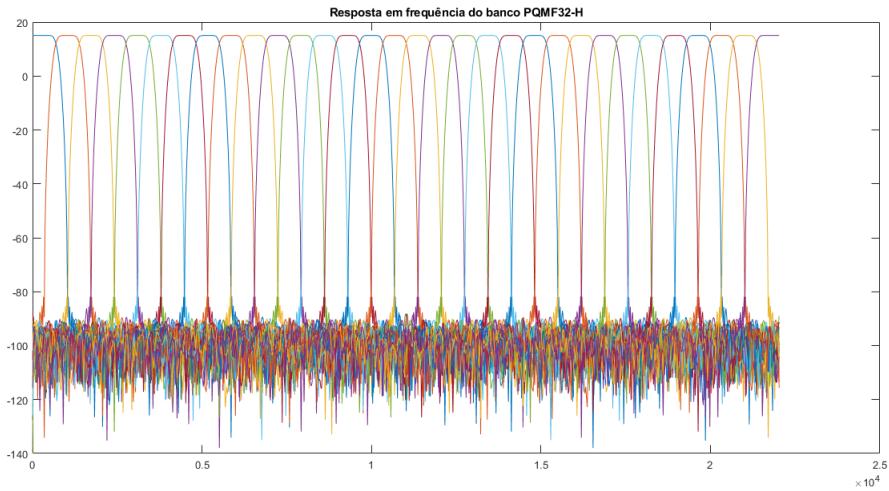


Figura 17. Resposta em frequência dos filtros H do banco de filtros "Analise_Sintese32".

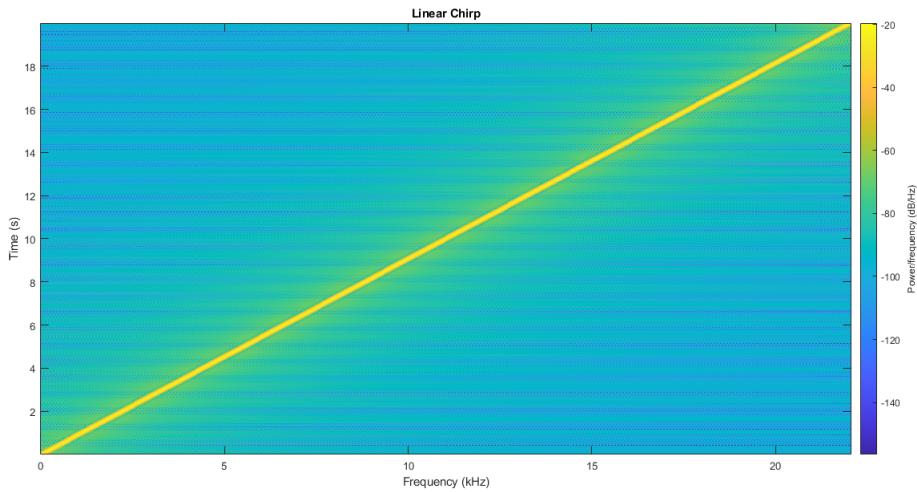


Figura 18. Espectrograma do novo sinal *chirp* linear.

b Foram feitas modificações nos *scripts* anteriores para realizar o processamento com o banco de filtros de 32 canais do novo sinal *chirp* linear, que agora possui 20 segundos de duração e varia de 0 a 22050 Hz. Pode-se ver o espectrograma do sinal de saída desse banco na figura 19, já os sinais de saída dos filtros G podem ser vistos na figura 20 e, por fim, o erro de reconstrução pode ser visto na figura 21.

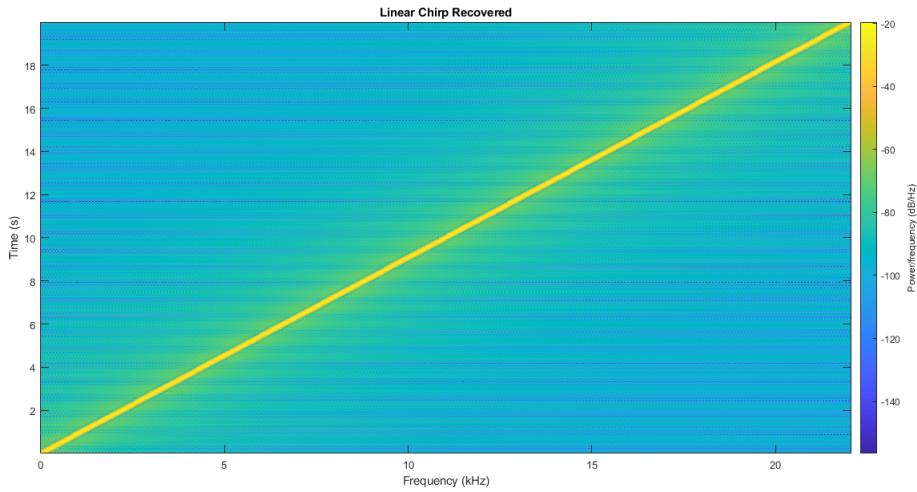


Figura 19. Espectrograma do sinal recuperado do banco de filtros "Analise_Sintese32".

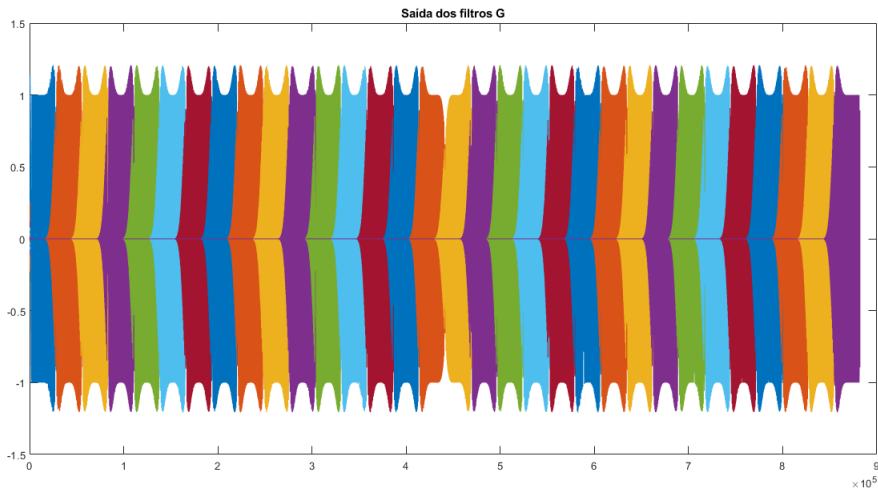


Figura 20. Sinais de saída de cada um dos filtros G.

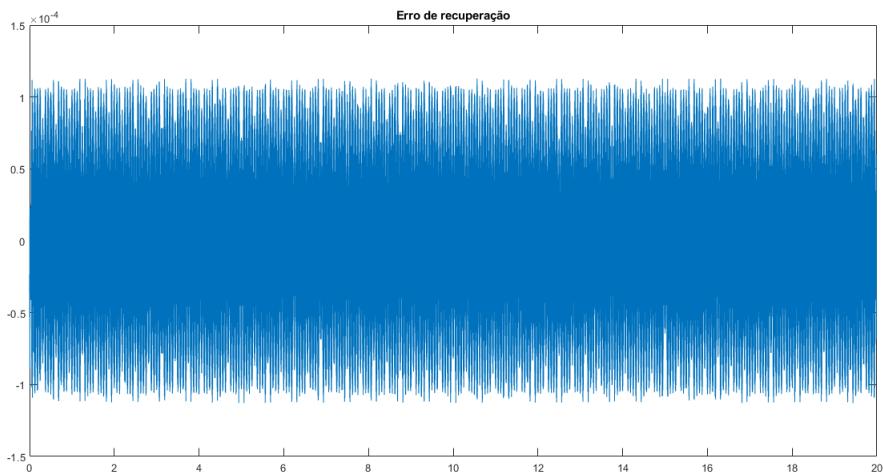


Figura 21. Erro de reconstrução no caso do banco de filtros pseudo-QMF com 32 canais.

- c O cálculo da SNR do sinal reconstruído, levando em consideração o atraso dos filtros, foi feito com a função "snr.m" disponibilizada com o enunciado e o valor obtido foi de $SNR = 84.3506dB$.
- d Esse item foi implementado no script "exp3_3d.m". O espectrograma do sinal "violino.wav" pode ser visto na figura 22. Já a saída dos filtros de síntese dos canais 0, 5 e 10 podem ser vistas na figura 23. O espectrograma do sinal de saída pode ser visto na figura 24 e o erro de recuperação está disponível na figura 25. Por fim, o cálculo da relação sinal ruído resultou em $SNR = 84.2443dB$.

É possível analisar que a amplitude dos sinais de saída de cada canal varia muito. Isso pode gerar diversos tipos de possibilidades, por exemplo utilizar quantificações diferentes para cada canal ou para cada trecho de cada cana, como será feito futuramente. Além disso, caso haja conhecimento prévio do sinal a ser processado, seria possível poupar o esforço computacional de processar canais que irão operar em bandas com potência desprezível em relação ao sinal total.

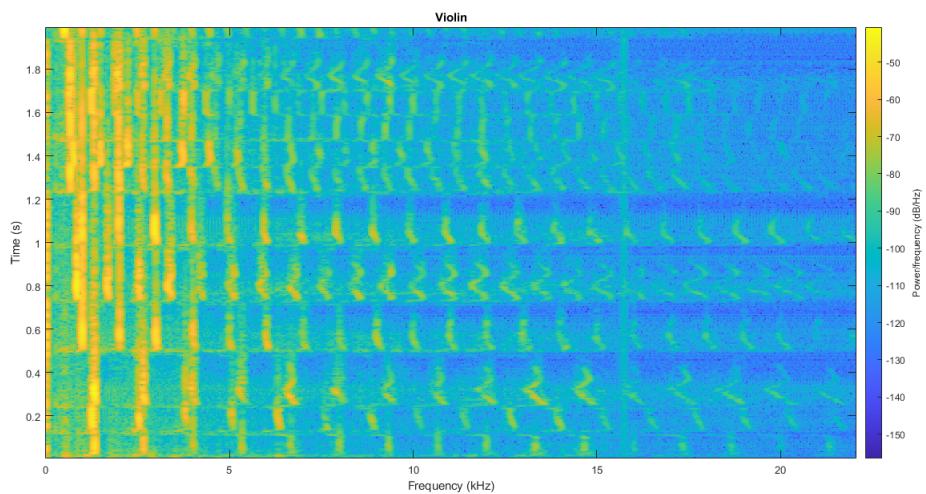


Figura 22. Espectrograma do sinal "violino.wav".

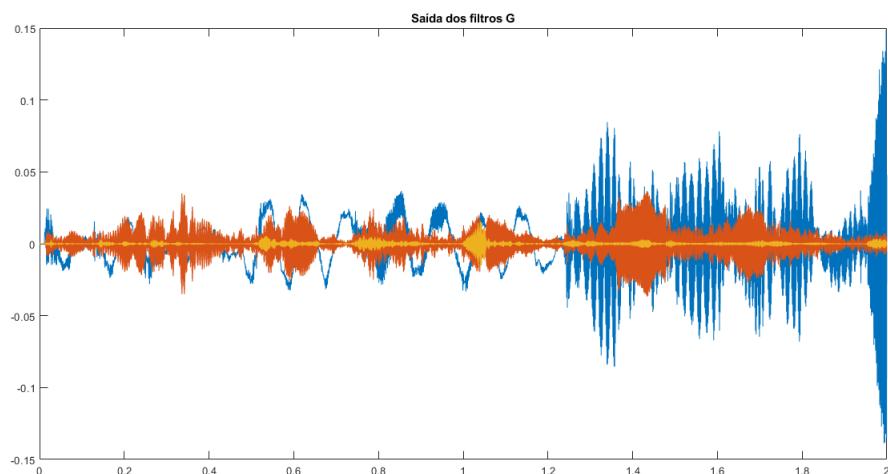


Figura 23. Saída dos filtros de síntese dos canais 0, 5 e 10.

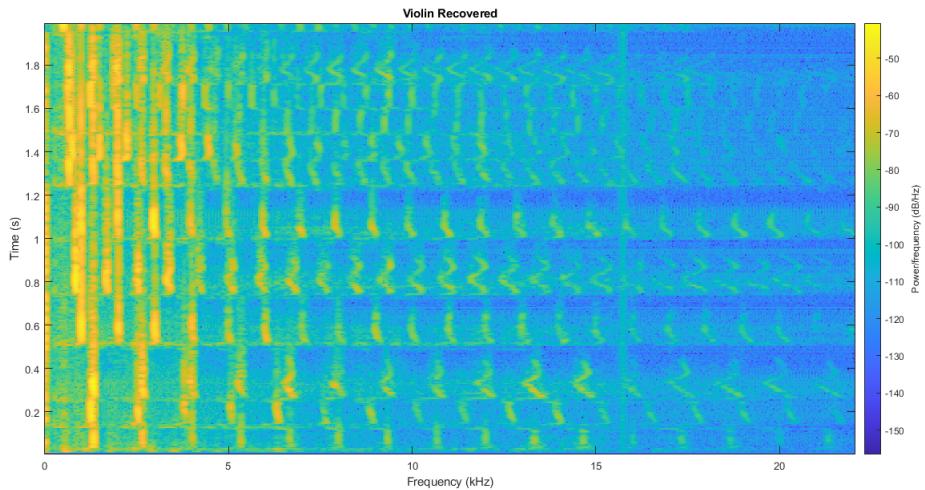


Figura 24. Espectrogrma do sinal recuperado do banco de filtros pseudo-QMF de 32 canais, tendo como entrada o sinal "violino.wav".

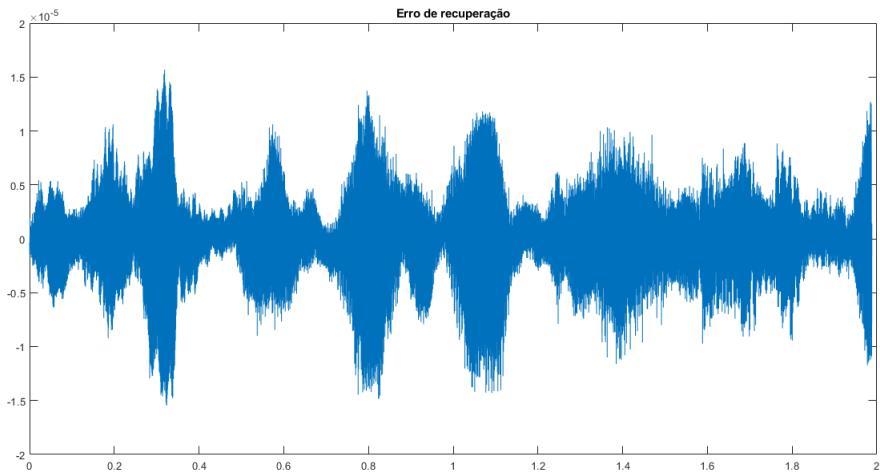


Figura 25. Erro de recuperação do sinal "violino.wav".

4- Codificação de áudio perceptual

Esse exercício está implementado nos *scripts* de Matlab "exp3_4a.m", "exp3_4b.m", "exp3_4c.m" e "exp3_4d.m" entregues em conjunto com esse relatório. Nessa parte da experiência são testados diferentes métodos de quantização, como mid-tread e perceptual, e a implementação dos filtros com a utilização da transformada *Lapped*.

- a Esse exercício está implementado no *script* "exp3_4a.m". Pode-se notar que a quantização do tipo *mid-tread* com o fator igual a 1, aplicada a todos os canais, distorceu consideravelmente o sinal de saída do banco de filtros. Como prova disso, a SNR caiu de $SNR = 84.2443dB$, no caso anterior, para $SNR = 4.1030dB$ nesse caso. O espectrogrma do sinal de saída e o erro de reconstrução podem ser vistos nas figuras 26 e 27.

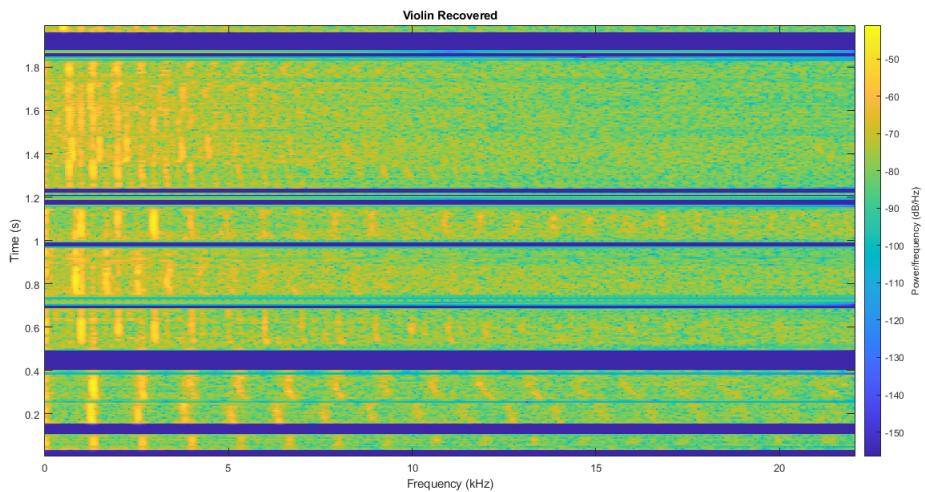


Figura 26. Espectrograma do sinal reconstruído, utilizando quantização *mid-tread* com fator fixo.

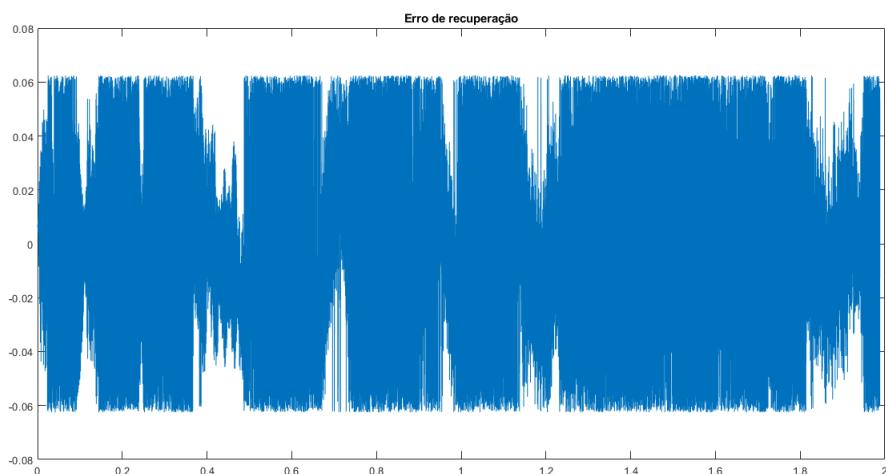


Figura 27. Erro de reconstrução do sinal "violino.wav", utilizando quantização *mid-tread* com fator fixo.

- b Esse exercício está implementado no script "exp3_4b.m". Ao se utilizar a função "lappedQ.m", que implementa tanto a transformada *Lapped* quanto a quantização *mid-tread* com fator fixo, houve uma melhoria significativa em relação ao caso anterior, porém ainda com distorção significativa e audível do sinal reconstruído. Pode-se observar o espectro do sinal reconstruído e o erro de reconstrução nas figuras 28 e 29, respectivamente. Já o cálculo da relação sinal-ruído resultou em $SNR = 10.3338dB$, maior do que a obtida anteriormente.

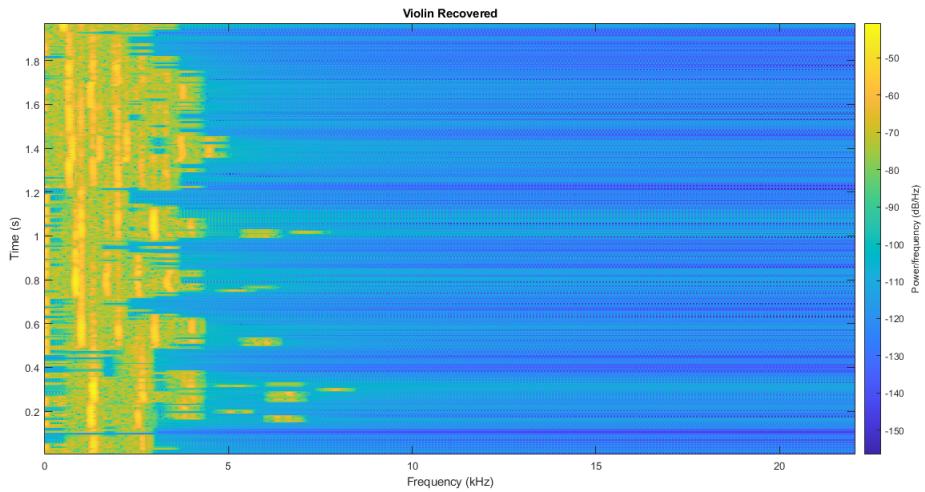


Figura 28. Espectrograma do sinal reconstruído utilizando a função "lappedQ.m".

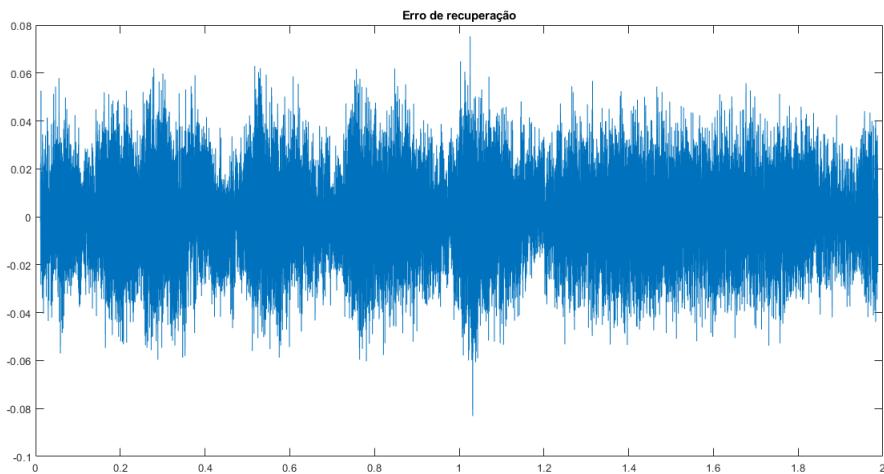


Figura 29. Erro de reconstrução do sinal "violino.wav" utilizando a função "lappedQ.m".

- c Esse exercício está implementado no script "exp3_4c.m". Nesse caso, o fator da quantização do tipo *mid-tread* é calculado adaptativamente, de acordo com cada bloco de amostras, de forma que $\rho_K = \max_{bloco} \{|x[n]| \}$. Essa pequena mudança no algoritmo foi capaz de melhorar significativamente o sinal recuperado. O espectrograma do sinal de saída e o erro de recuperação podem ser vistos nas figuras 30 e 31, respectivamente. Nota-se um aumento significativo também na relação sinal-ruído, que nesse caso foi de $SNR = 25.0629dB$. Vale salientar também que, nesse caso, a distorção sonora já não é mais facilmente perceptível.

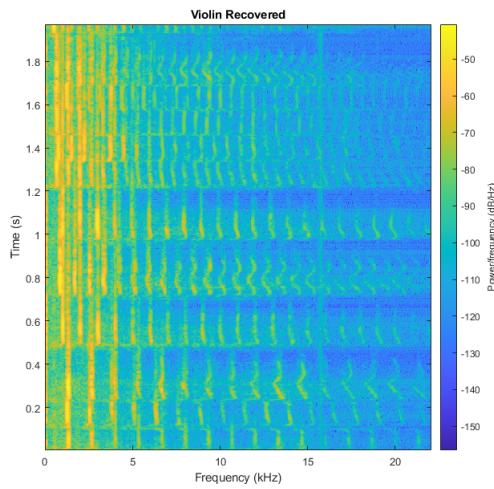


Figura 30. Espectrograma do sinal reconstruído utilizando a função "lappedQadap.m".

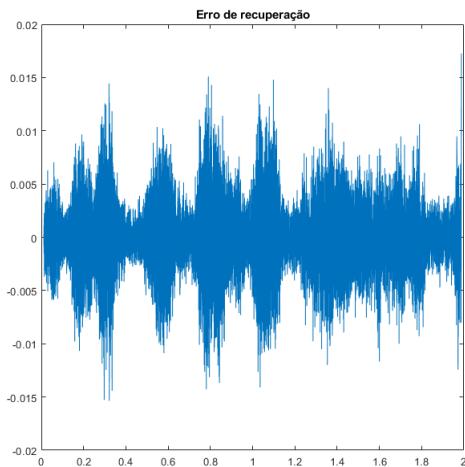


Figura 31. Erro de reconstrução do sinal "violino.wav" utilizando a função "lappedQadap.m".

- d Esse exercício está implementado no script "exp3_4d.m". Nesse caso é implementado o modelo psicoacústico, com auxílio da função "lappedPsico.m", que considera o limiar de percepção humana para cada frequência sonora. O espectrograma do sinal de saída e o erro de recuperação podem ser vistos nas figuras 32 e 33, respectivamente. Essa implementação também gerou um aumento da relação sinal-ruído em relação ao caso anterior, igual a $SNR = 38.6457dB$ nesse caso.

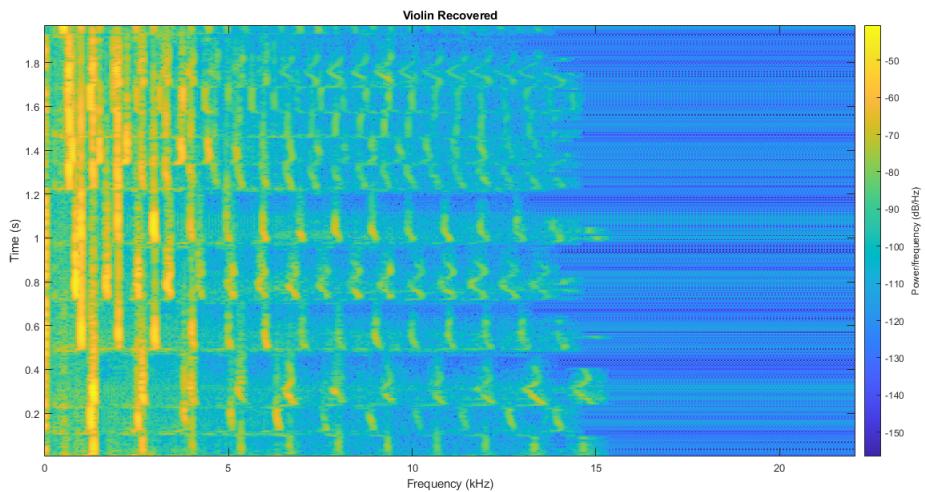


Figura 32. Espectrogrma do sinal reconstruído utilizando a função "lappedPsico.m".

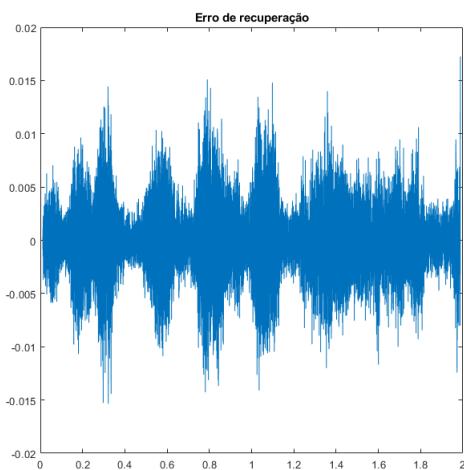


Figura 33. Erro de reconstrução do sinal "violino.wav" utilizando a função "lappedPsico.m".

- e Esse exercício está implementado no script "exp3_4d.m". O gráfico da SMR em relação aos canais e aos blocos pode ser visto na figura 34. É possível notar que os canais mais altos, ou seja, as faixas de frequêcia mais altas, estão com SMR abaixo de zero. Isso quer dizer que essas faixas de frequêcia possuem potêcia abaixo do limiar audível. A consequêcia desse fato no algoritmo pode ser vista na figura 35, que mostra o númer de bits alocados em relaçao aos canais e aos blocos de amostras. Esses canais com SMR menor do que zero, não tiveram bits alocados. Porém, mesmo que isso gere erro na reconstruçao do sinal, não será perceptível ao ouvido humano.

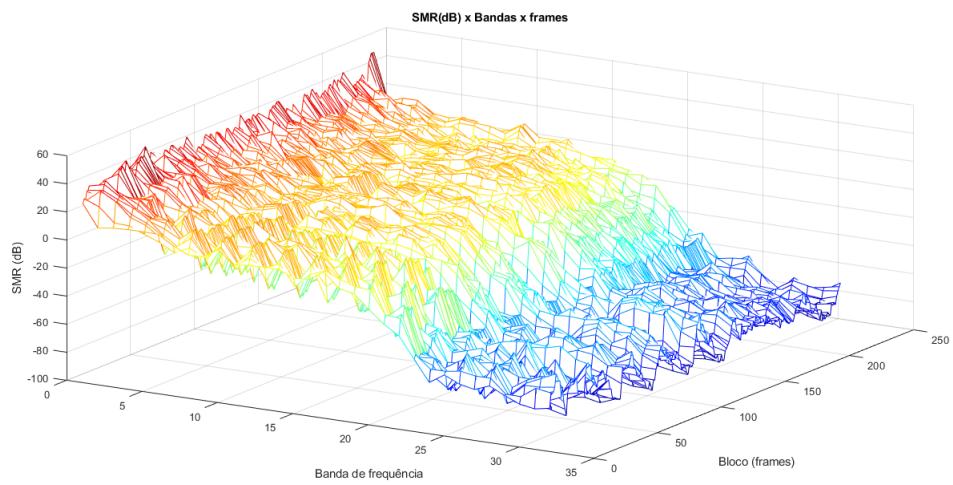


Figura 34. Gráfico da SMR em relação aos canais e aos blocos de amostras.

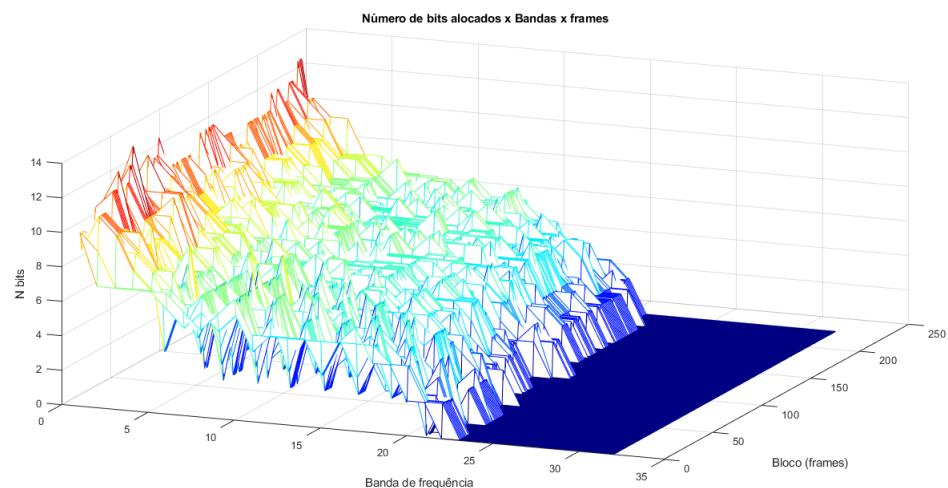


Figura 35. Gráfico do número de bits alocados em relação aos canais e aos blocos de amostras.

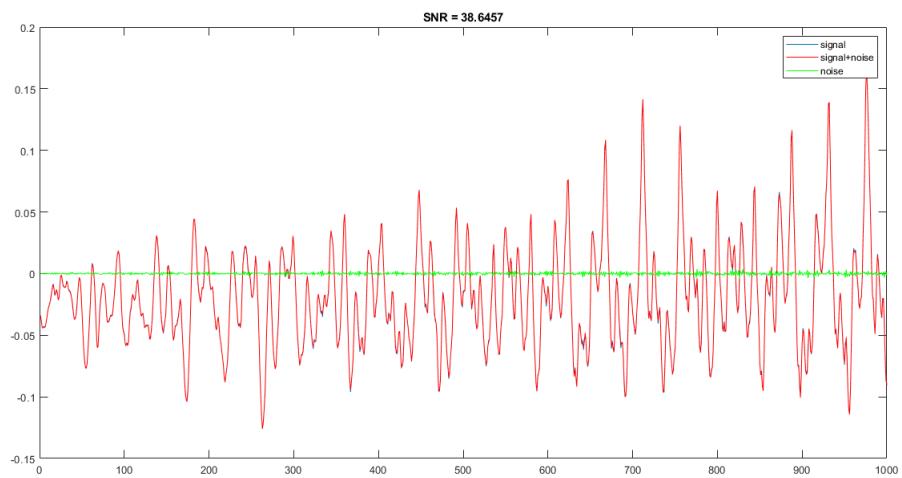


Figura 36. Gráfico do erro, do sinal reconstruído e do sinal original.