

Tarefa 3 – Processamento Digital de sinais

Prof: Ricardo Queiroz

Aluno: Lindeberg Pessoa Leite - Matrícula 200051733

A tarefa consiste em realizar a filtragem de um sinal de áudio utilizando um filtro passa-baixa por 3 métodos diferentes:

1. Convolução no domínio do tempo;
2. Overlap and add;
3. Overlap and save.

Para convolução no domínio do tempo, utilizou-se a função conv do Matlab:
`convolution=conv(x,h);`

Para o método Overlap and add, implementaram-se os seguintes passos:

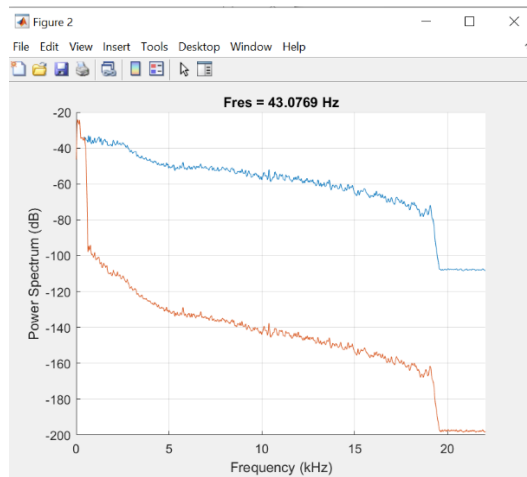
1. Dividiu-se $x[n]$ em blocos de comprimento N ;
2. Preencheram-se $h[n]$ e cada bloco $x_m[n]$ com zeros até o comprimento $(N+K-1)$;
3. Calculou-se a FFT para cada bloco de comprimento $(N+K-1)$;
4. Somaram-se os resultados de acordo com o método Overlap and add.

Para o método Overlap and save, os seguintes passos foram implementados:

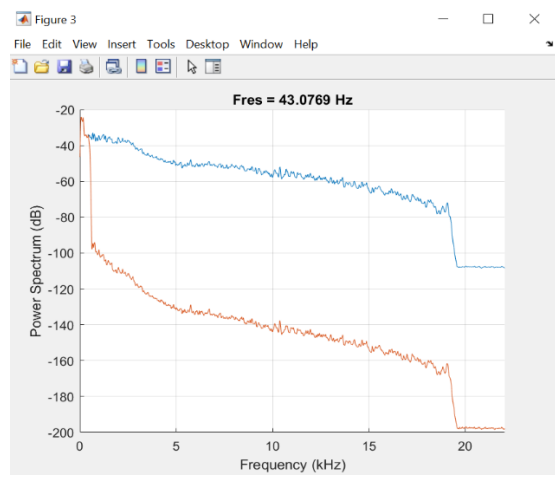
1. $x[n]$ foi dividido em blocos de comprimento N com sobreposição de $K-1$ amostras;
2. $h[n]$ foi preenchido com zeros até o comprimento N ;
3. Foi calculado a FFT para cada bloco de comprimento N ;
4. O sinal foi combinado de acordo com o método Overlap and save.

Passamos a analisar o resultado da implementação

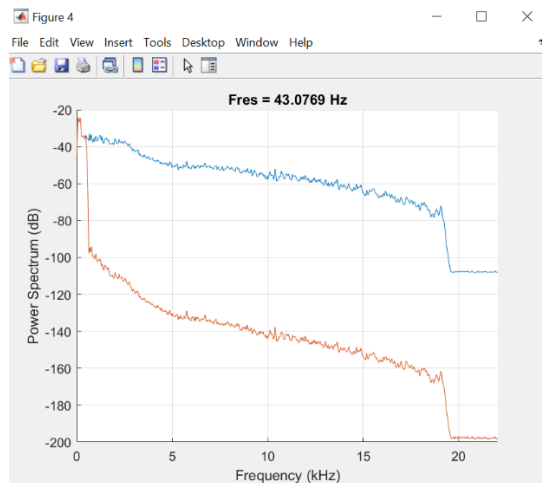
1. Comparação da frequência original do sinal e da frequência do sinal convoluído, ou seja, após passar pelo filtro passa-baixa:



Frequência original e Frequência convoluída/filtrada (função conv)



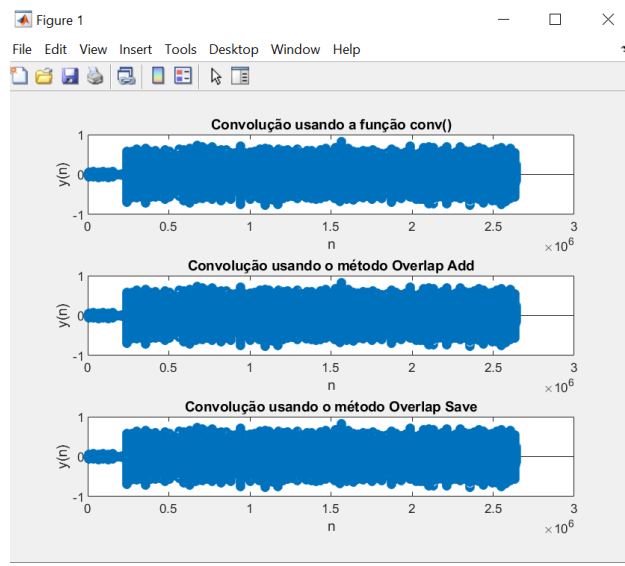
Frequência original e Frequência convoluída/filtrada (overlap_add)



Frequência original e Frequência convoluída/filtrada (overlap_save)

Das figuras apresentadas acima, percebe-se que de fato houve uma filtragem, ao permitir a passagem apenas de frequências mais baixas.

2. Na figura descrita abaixo, é possível notar que o resultado da convolução usando o método de convolução linear conv(), o método Overlap add e Overlap save apresentam o mesmo resultado.



Convolução linear usando três métodos – conv(),
Overlap add e Overlap save

Entretanto, devido à grande quantidade de pontos, apenas por inspeção visual, não temos certeza se as três convoluções produziram de fato o mesmo resultado. Desse modo, também foi implementado no código o cálculo do erro médio entre os sinais. Como cada sinal representa uma sequência de números e está representada em um vetor, basicamente calculou-se a diferença entre os vetores e computou-se o erro médio (foi preciso realizar um preenchimento com zeros para os sinais que não tinha o mesmo tamanho)

```
%Erro médio entre convolução e overlap_add  
erro_media_conv_add = 6.2199e-17
```

```
%Erro médio entre convolução e overlap_save  
erro_media_conv_save = 4.0316e-05
```

```
%Erro médio entre overlap_add e overlap_save  
erro_media_add_save = 4.0316e-05
```

Note que os erros entre os sinais são extremamente pequenos. Desse modo, em termos práticos, esses sinais podem ser considerados iguais.