



FACULDADE DE  
CIÊNCIAS E TECNOLOGIA  
UNIVERSIDADE DE  
**COIMBRA**

**dei25**

departamento  
de engenharia informática  
1995 – 2020

## Relatório

### **Trabalho 3 - Representação de sinais de voz usando predição linear (LPC)**

Unidade Curricular:  
***Processamento Audiovisual (PA)***

Licenciatura em Engenharia e Ciência de Dados

Realizado por:

Diogo Beltran Dória, 2020246139  
Mariana Lopes Paulino, 2020190448

Ano Letivo 2022/2023

## Introdução

O objetivo deste trabalho prático é a execução de experiências envolvendo predição linear para representar sinais de voz, recorrendo a scripts em MATLAB e respectivas explicações apresentadas em [1] de modo a serem examinadas as várias operações de processamento de sinal que ocorrem durante a predição linear e codificação do erro de predição.

## Experiências Realizadas

Primeiramente é fornecido um script MATLAB de modo a capturarmos um sinal de voz com o nosso computador, neste caso denominamos este ficheiro de “fala.wav” e vamos fazer a análise deste ficheiro em comparação com o que nos foi dado previamente, nomeadamente o ficheiro “speech.wav”.

### Experiência 1 - Examinar um Ficheiro de Áudio

O áudio no qual realizamos esta experiência foi o “speech.wav” uma vez que nos era fornecido e já fazia parte do script MATLAB, em seguida realizámos as mesmas experiências ao áudio gravado com o código fornecido.

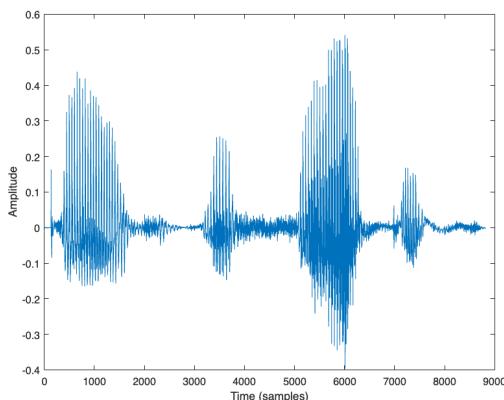


Fig. 1 - Periodograma do áudio “speech”.wav

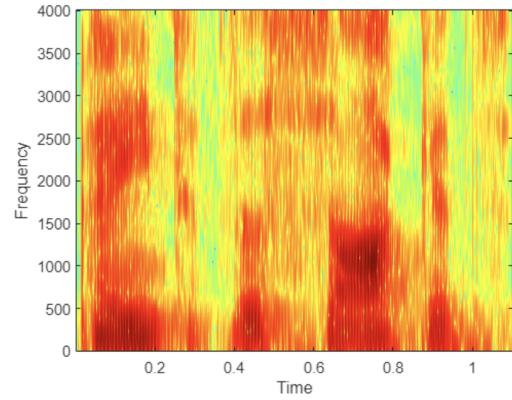


Fig. 2 - Espectrograma do áudio “speech.wav”

O áudio seguinte foi o “fala.wav” que é o áudio gravado através do código fornecido obtendo os resultados representados na figura 3 e na figura 4.

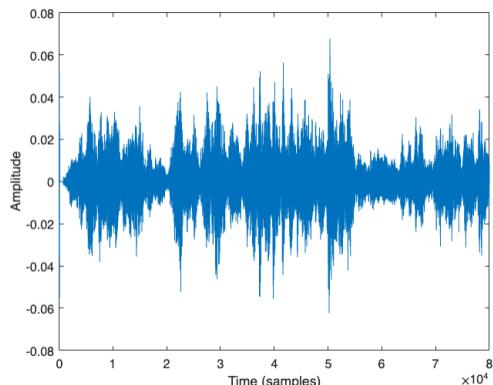


Fig. 3 - Periodograma do áudio “fala.wav”

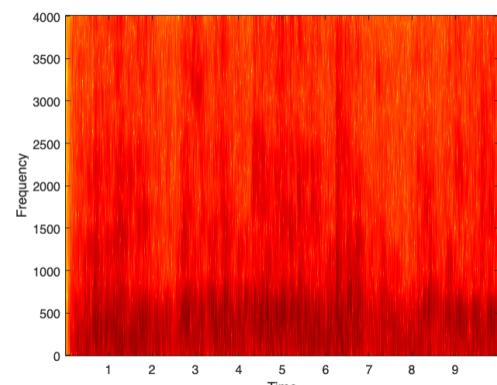


Fig. 4 - Espectrograma do áudio “fala.wav”

Com a análise dos dois áudios conseguimos ver que no primeiro, a voz e o som são mais nítidos, conseguindo perceber o que é dito claramente e no áudio gravado existe uma pior qualidade levando a não conseguirmos perceber bem a informação que é dita nele.

### Experiência 2 - Previsão linear de Síntese de 30ms de áudio

Nesta experiência o objetivo é extrair 30ms de cada um dos áudios onde obtemos como resultado as figuras 4 e 5, respectivamente para os áudios “speech.wav” e “fala.wav”.

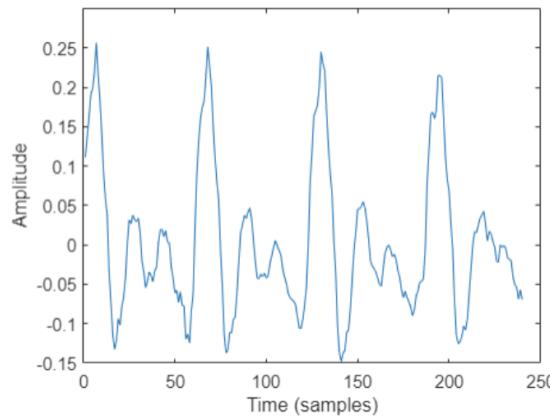


Fig. 4 - Áudio “speech.wav”

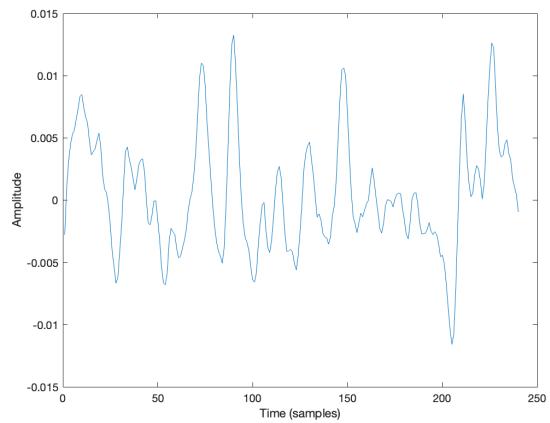


Fig. 5 - Áudio “fala.wav”

Agora vamos ver o conteúdo espectral deste , mostrando o seu periodograma em 512 pontos (usando um eixo de frequência normalizado; sabendo que pi corresponde a Fs/2, ou seja, a 4000 Hz). Nas figuras 6 e 7 apresentam-se os resultados de ambos os áudios.

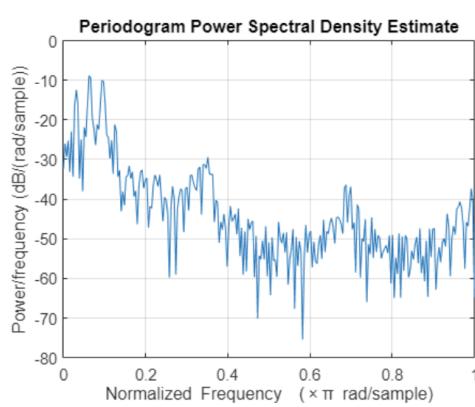


Fig. 6- Periodograma Espectral do Áudio “speech.wav”

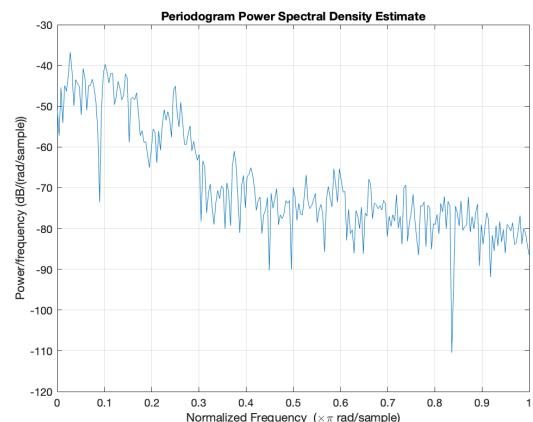


Fig. 7- Periodograma Espectral do Áudio “fala.wav”

Ao fazermos um modelo de predição linear de ordem 10 obtemos os vários coeficientes e a variância do sinal residual. No caso do áudio “speech.wav” estes são:

```
ai = 1x11
1.0000   -1.3928    0.1280    0.4423   -0.0009   ...
sigma_square = 2.2567e-04
```

Fig. 8- Apresentação dos vários coeficientes e da variância do sinal residual do áudio “speech.wav”

Na figura acima conseguimos observar os valores de ai, que correspondem aos vários coeficientes do modelo e a sigma\_square que corresponde à variância do sinal residual do erro do áudio.

Na figura apenas são demonstrados alguns dos coeficientes, de modo a conseguirmos comparar todos os coeficientes, os coeficientes do modelo de ordem 10 para o áudio “speech.wav” são 1, -1.39277929588556, 0.127950079467969, 0.442250618316686, -0.000905465939885216, -0.221586945842148, -0.400588171348072, 0.966660225044421, -0.288350551386893, -0.202190991140667, 0.0548323184732264. A estes coeficientes vem aliado uma variância de 2.2567 e-4.

O algoritmo de estimação utilizado num LPC é chamado de Levinson-Durbin que consiste em escolher os coeficientes de um filtro FIR  $A(z)$  de modo a que aquando da passagem da frame de input em  $A(z)$  o output seja determinado como previsão residual com a minima energia possível. Isto leva a que um filtro tenha anti-resonâncias em qualquer que seja o formato que o input venha. Deste modo este filtro é denominado de filtro inverso. Na próxima parte da experiência é necessário que demonstremos que a resposta em frequência (nos 512 pontos) e que demonstremos também o filtro da síntese, neste caso  $1/A(z)$  para ambos os áudios como demonstrado nas figuras 9 e 10.

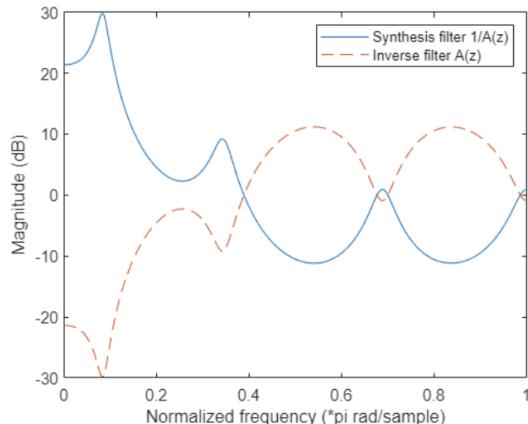


Fig. 9 - Resposta em Frequência do Filtro Síntese e do Filtro Inverso do áudio “speech.wav”

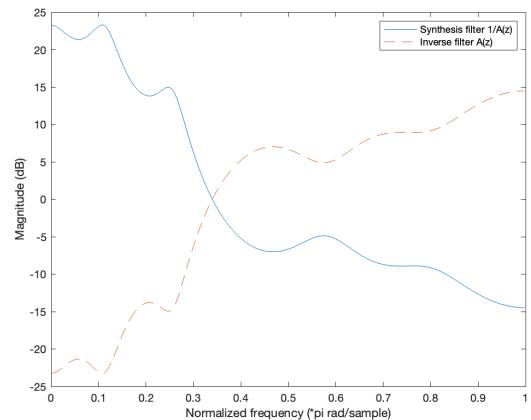


Fig. 10 - Resposta em Frequência do Filtro Síntese e do Filtro Inverso do áudio “fala.wav”

A resposta do filtro  $1/A(z)$  é igual à amplitude espectral da frame nos dois casos. Em seguida utilizámos o periodograma de modo a que consigamos verificar o valor do periodograma de um lado e o de dois lados, onde concluimos que o de um lado tem o dobro do valor do de dois lados como conseguimos verificar nas figuras abaixo.

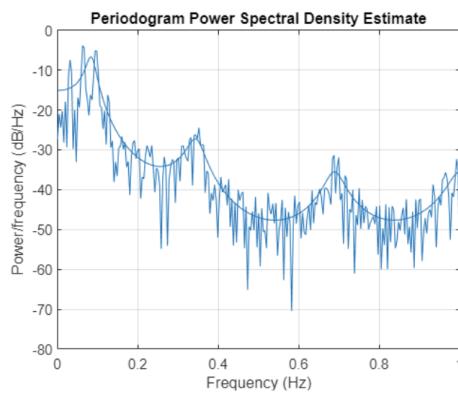


Fig. 11 - Periodograma da Energia Espectral do Áudio  
“speech.wav”

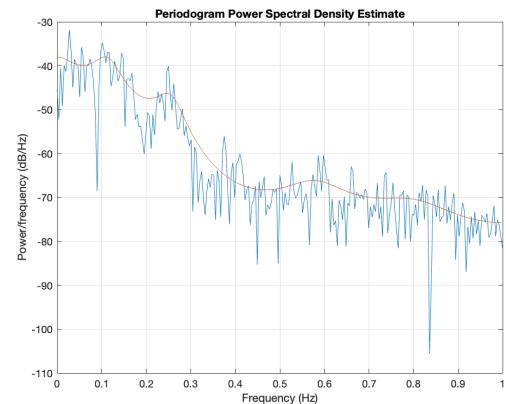


Fig. 12- Periodograma da Energia Espectral do Áudio  
“fala.wav”

Na parte seguinte da experiência o modelo sabemos que o modelo faz um fit automático e ajusta os polos do filtro sintetizado perto de um círculo num ângulo escolhido de modo a imitar o formato das ressonâncias como demonstrado na figura 13 e 14.

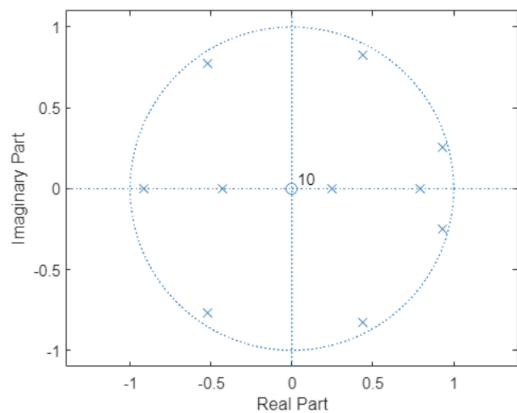


Fig. 13 - Imitação do formato das ressonâncias no áudio  
“speech.wav”

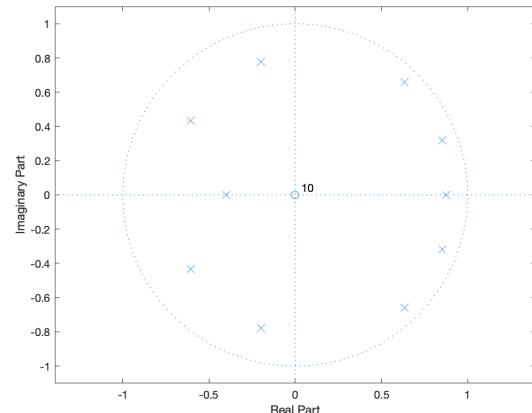
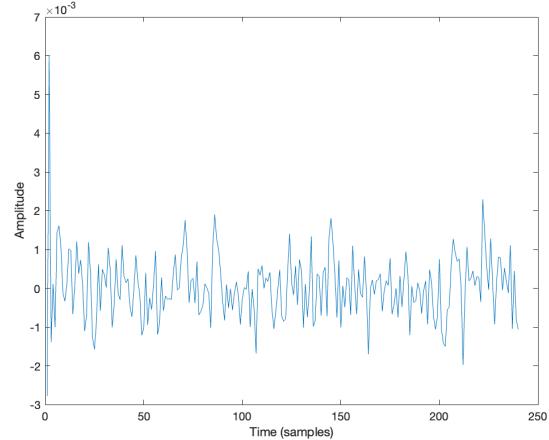
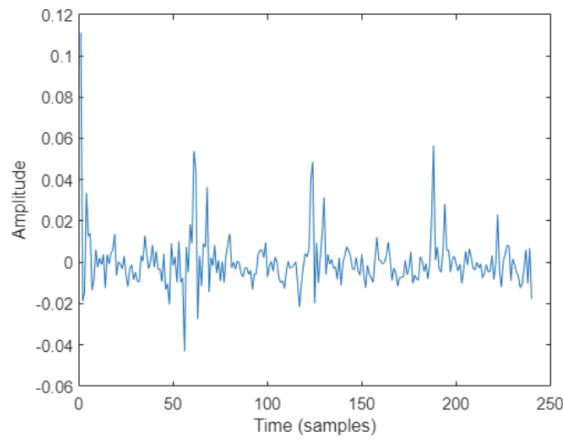


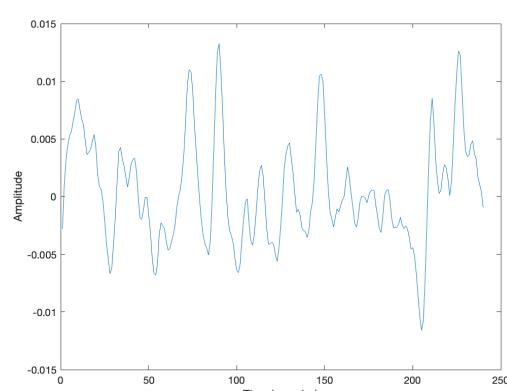
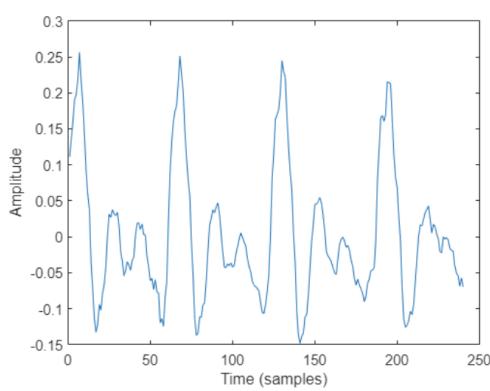
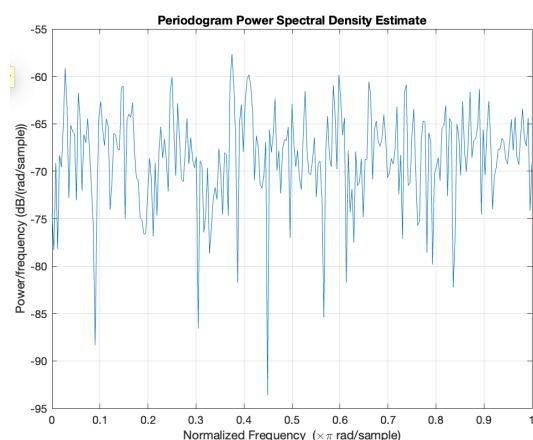
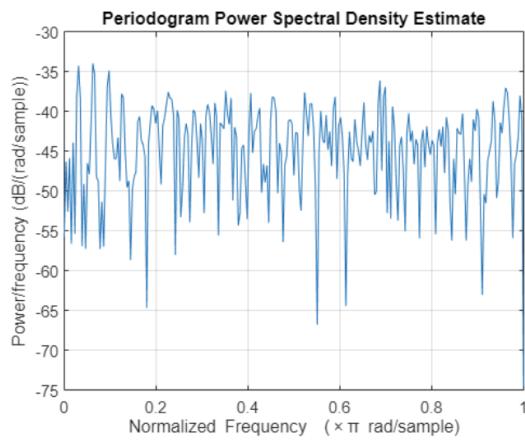
Fig. 14 - Imitação do formato das ressonâncias no áudio  
“fala.wav”

Após esta experiência ao aplicar o inverso do filtro à frame obtemos a predição residual demonstradas nas figuras 15 e 16.

Comparamos agora ambos os periódigramas do espetro original com o espetro residual de ambos os áudios. Obtendo assim os seguintes periódigramas dos sinais residuais:



Conseguimos observar os detalhes do espectro e também os seus harmônicos que são preservados. Ao aplicar o filtro de síntese a esta predição residual resulta na frame de análise seguinte:



O modelo LPC faz uma previsão residual da fala como treino com vários ajustes que podem ser feitos à amplitude e período, por exemplo. Para a frame considerada o LPC é uma sequência de 64 zeros,

isto é importo uma vez que temos um sinal com 65 amostras. Tendo como resultado os seguintes gráficos:

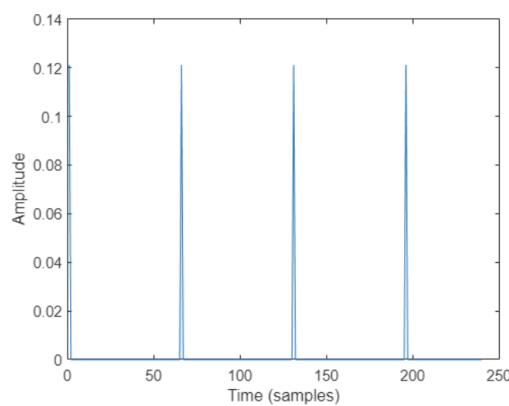


Fig. 21 - Gráfico correspondente ao áudio “speech.wav”

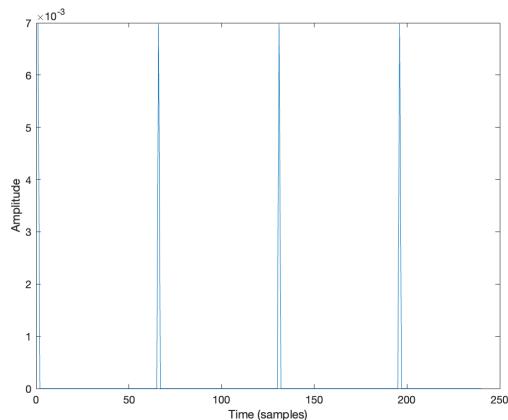


Fig. 22- Gráfico correspondente ao áudio “fala.wav”

Em seguida conseguimos verificar que a excitação do LPC é muito diferente da predição residual. O seu espetro, por outro lado, tem as mesmas características que o seu resíduo, são estas o conteúdo harmónico e o envelope. Em comparação as suas diferenças assentam no facto de na excitação o espetro ser harmônico demais comparado ao resíduo, demonstrado abaixo nas figuras 23 e 24.

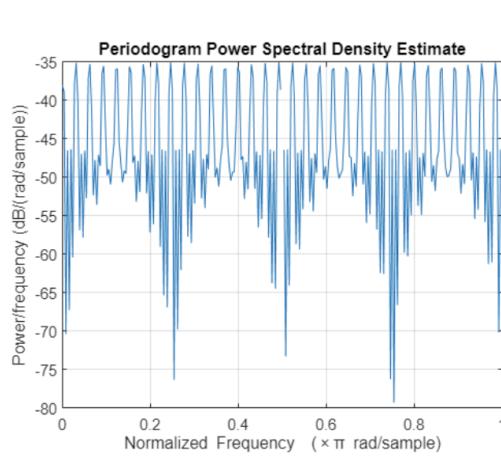


Fig. 23 - Periodograma correspondente ao áudio “speech.wav”

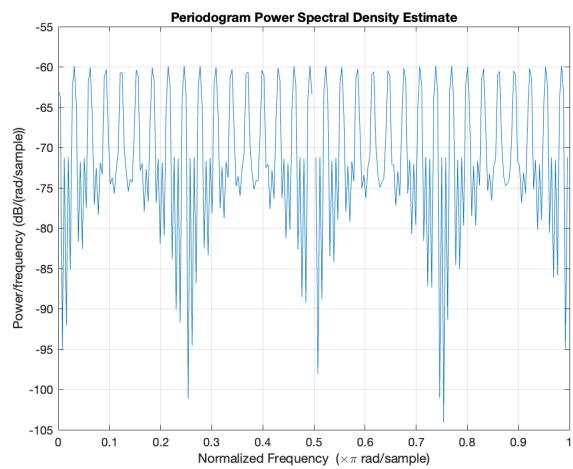


Fig. 24- Periodograma correspondente ao áudio “fala.wav”

Nos gráficos seguintes é utilizado um filtro de síntese de modo a produzir uma variável.

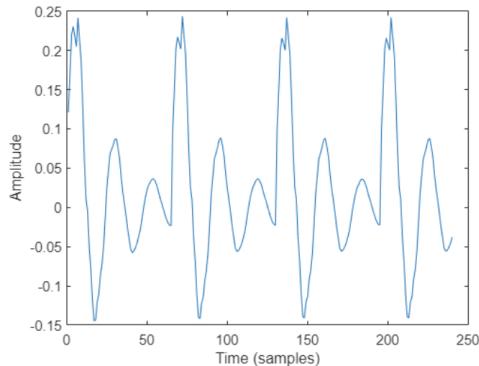


Fig. 25 - Gráfico correspondente ao áudio “speech.wav”

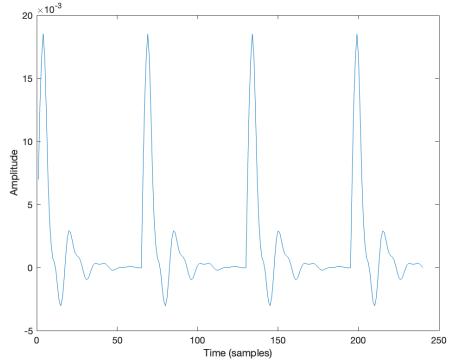


Fig. 26- Gráfico correspondente ao áudio “fala.wav”

De modo a concluir esta experiência nas figuras 27 e 28 notamos que a sua forma difere das originais por o modelo LP não considerar a fase espectral do sinal original, os seus detalhes harmónicos também alteram, mas diferem simultaneamente, sendo demasiado harmónicos.

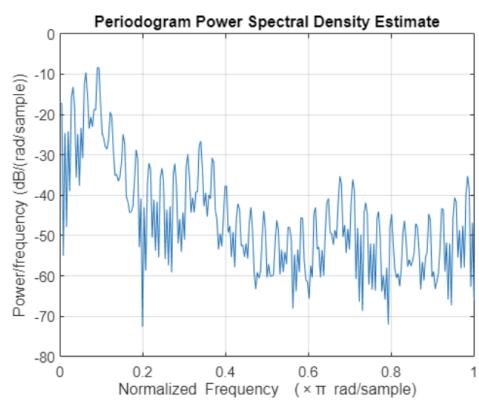


Fig. 27 - Gráfico correspondente ao áudio “speech.wav”

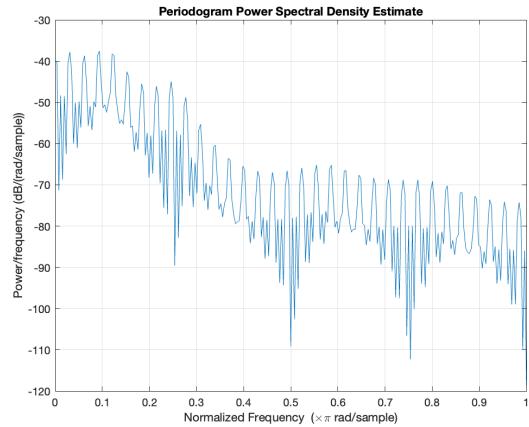


Fig. 28- Gráfico correspondente ao áudio “fala.wav”

### **Experiência 3 - Predição Linear de 30ms de Speech Unvoiced**

O objetivo desta parte da experiência é aplicar os mesmos processos novamente a uma frame unvoiced e comparar o espetro final. Primeiro extraímos uma frame do sinal que não tenha voz. Obtendo os seguintes resultados tanto para o áudio “speech.wav” e “fala.wav” figuras 29 e 30 respectivamente.

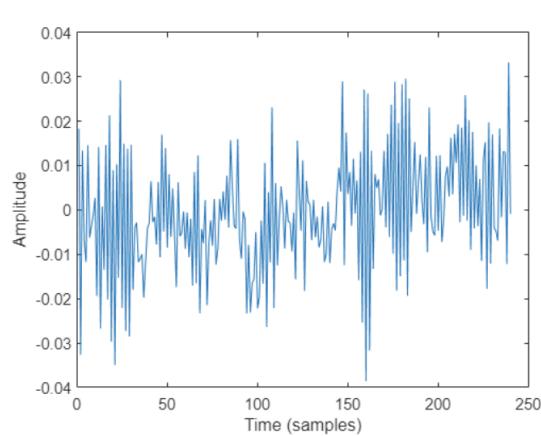


Fig. 29 - Gráfico correspondente ao áudio “speech.wav”

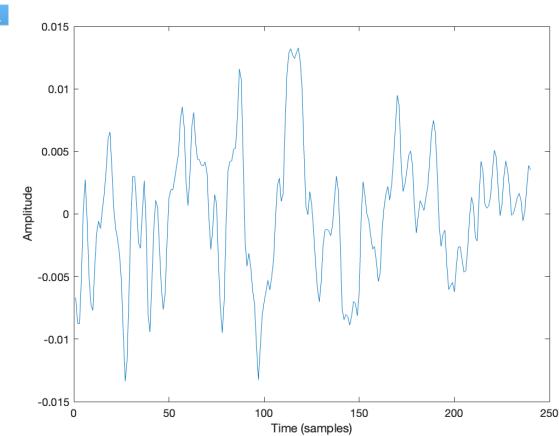


Fig. 30- Gráfico correspondente ao áudio “fala.wav”

Como esperado não conseguimos notar nenhuma peridiocidade nos gráficos acima, o próximo passo é a verificação do conteúdo espectral da frame. Para isso utilizamos a função do MATLAB pwlech que utiliza o periograma médio e, simultaneamente, estima melhor a densidade do poder espectral com menos resolução de frequência que ao utilizar um simples periograma. Isto é feito com 8 sub frames por omissão de especificação e 50% de overlap das frames obtendo os resultados das figuras 31 e 32.

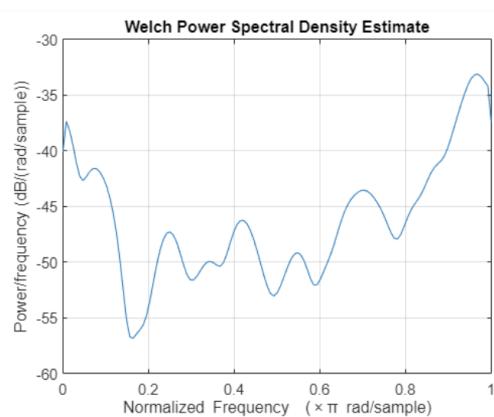


Fig. 31 - Gráfico correspondente ao áudio “speech.wav”

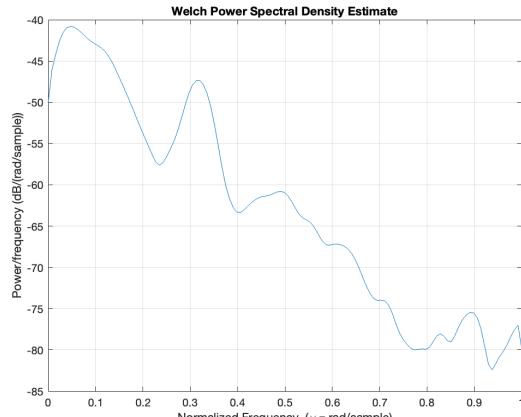


Fig. 32- Gráfico correspondente ao áudio “fala.wav”

Ao gráficos anteriores aplicamos novamente um modelo LPC de ordem 10 e sintetizamos uma nova frame, ficando com os gráficos presentes abaixo nas figuras 33 e 34.

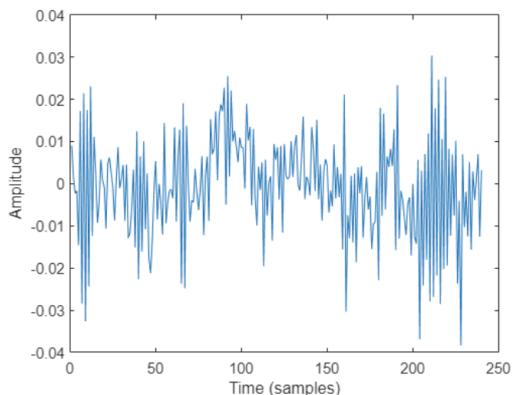


Fig. 33 - Gráfico correspondente ao áudio “speech.wav”

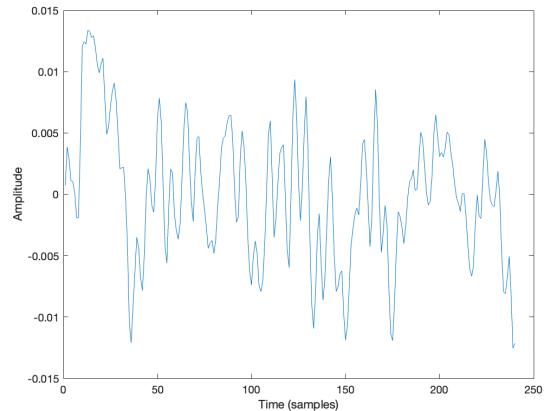


Fig. 34- Gráfico correspondente ao áudio “fala.wav”

Ao comparar a sintetizada com a normal concluimos que não existe nada em comum. No entanto os envelopes das frames representados nas figuras 35 e 36 são similares aos originais.

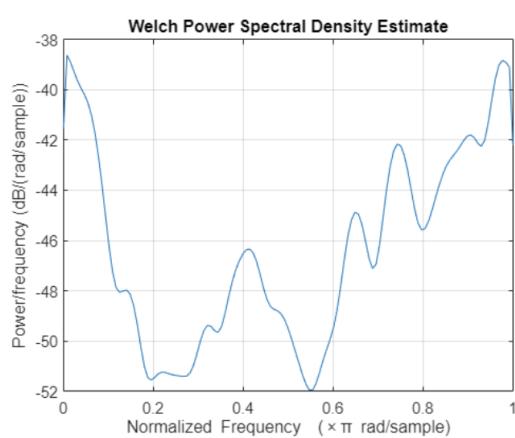


Fig. 35 - Gráfico correspondente ao áudio “speech.wav”

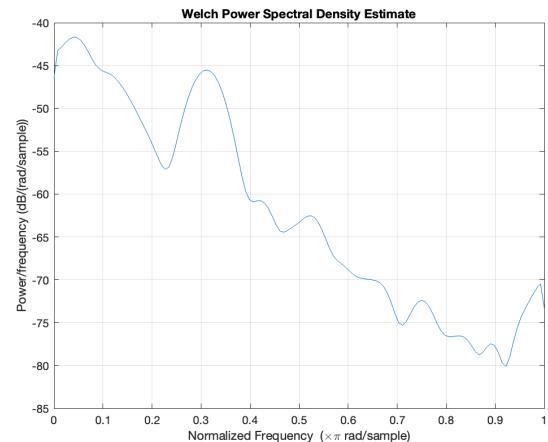


Fig. 36- Gráfico correspondente ao áudio “fala.wav”

#### Experiência 4 - Predição Linear Sintetizada de um speech file

Para esta nova experiência utilizaremos o que já foi feito previamente utilizando o ficheiro completo onde as frames analisadas vão ser de 30ms com um overlap de 20ms. Estas são agora pesadas. Aquando da síntese apenas sintetizaremos 10ms de fala, e concatenar o resultado das frames sintetizadas para obter o ficheiro de áudio de output. Escolhemos por conveniência 200 Hz para sintetizar  $F_0$ . Desta forma, 10ms de excitação têm exatamente 2 pulsos.

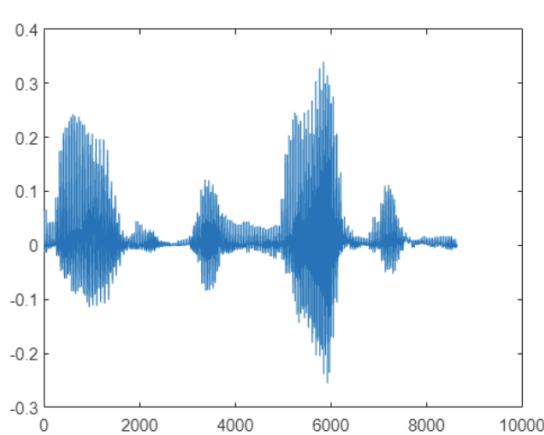


Fig. 37 - Gráfico correspondente ao áudio “speech.wav”

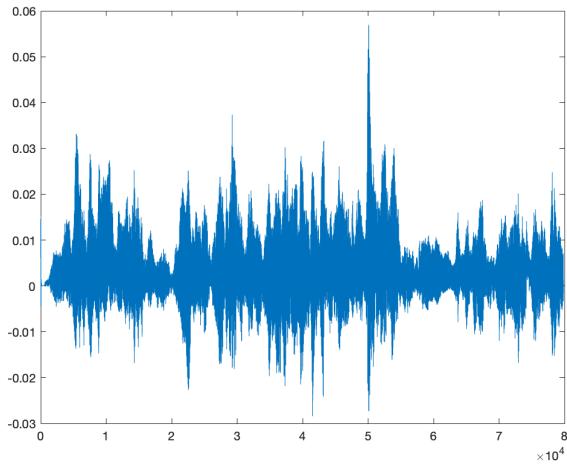


Fig. 38- Gráfico correspondente ao áudio “fala.wav”

A onda de output contém uma sequência de impulsos de resposta como demonstrado nas figuras abaixo representadas. Na figura 39 encontramos o gráfico correspondente ao áudio “speech.wav” e na figura 40 o gráfico do áudio “fala.wav”.

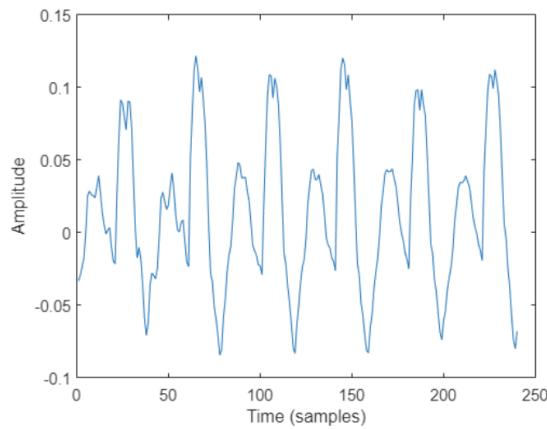


Fig. 39 - Gráfico correspondente ao áudio “speech.wav”

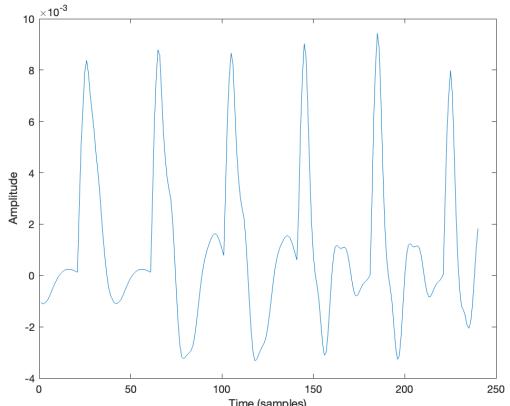


Fig. 40- Gráfico correspondente ao áudio “fala.wav”

Parece que em muitos casos as respostas de impulso foram cortadas. Na verdade, uma vez que cada janela de síntese era composta de dois impulsos idênticos, deve-se esperar que nossa LPC exiba pares de períodos de pitch idênticos. Este não é o caso, devido ao facto de que para produzir cada nova janela sintética as variáveis internas do filtro de síntese são redefinidas implicitamente para zero. Podemos evitar esse problema mantendo as variáveis internas do filtro desde o final de cada janela até o início da próxima.

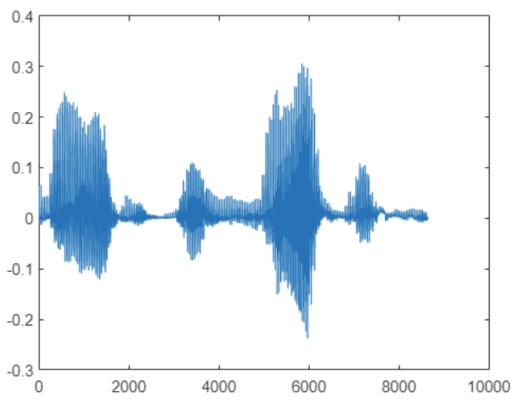


Fig. 41 - Gráfico correspondente ao áudio “speech.wav”

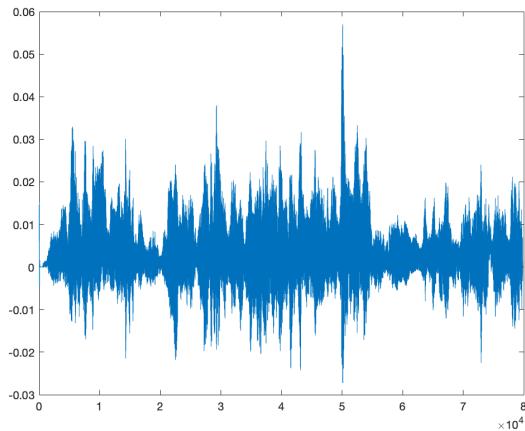


Fig. 42- Gráfico correspondente ao áudio “fala.wav”

Como no fim de cada resposta de impulso é propriamente adicionada ao início da seguinte, os resultados são facilmente verificados ao resultar em períodos de evolução mais suaves como os das figuras 43 e 44.

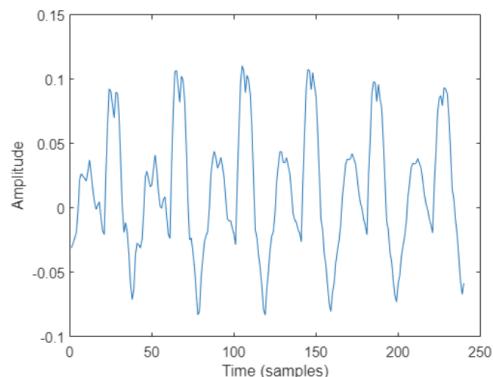


Fig. 43 - Gráfico correspondente ao áudio “speech.wav”

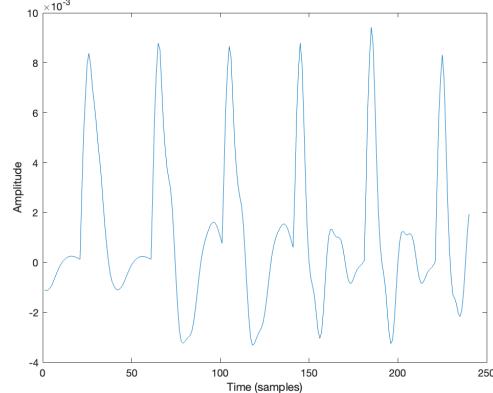


Fig. 44- Gráfico correspondente ao áudio “fala.wav”

Repetimos as operações anteriores, mas desta vez para o áudio completo, usando janelas de análise de 30 ms sobrepostas por 20 ms. As janelas agora são ponderadas por uma janela Hamming. No momento da síntese, simplesmente sintetizamos 10 ms de áudio e concatenamos as janelas sintéticas resultantes para obter o input de saída. Vamos escolher 200 Hz como síntese F0, por conveniência: desta forma cada janela de excitação de 10ms contém exatamente dois pulsos.

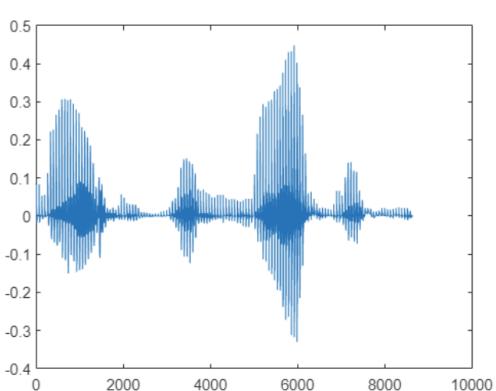


Fig. 45 - Gráfico correspondente ao áudio “speech.wav”

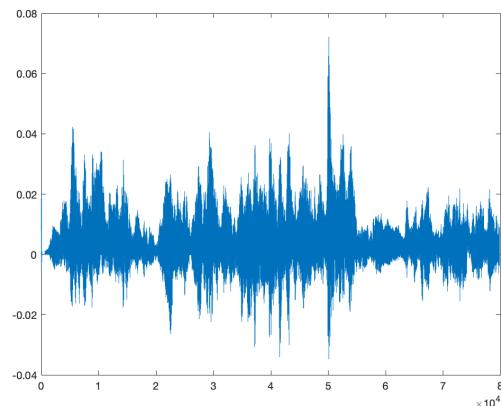


Fig. 46- Gráfico correspondente ao áudio “fala.wav”

### **Experiência 5 - Predição Linear de uma síntese de um speech file**

Os resultados da sintetização do ficheiro de speech inteiro unvoiced são apresentados nas figuras 47 e 48.

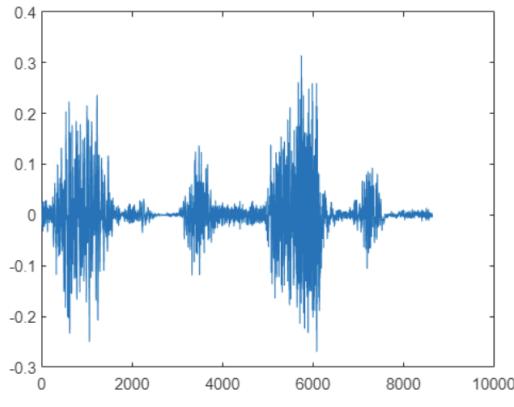


Fig. 47 - Gráfico correspondente ao áudio “speech.wav”

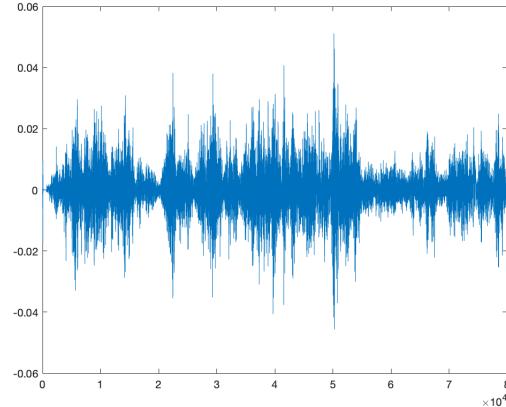


Fig. 48- Gráfico correspondente ao áudio “fala.wav”

### **Experiência 6 - Predição Linear da Síntese de um Speech File com F0 original**

Vamos agora sintetizar o mesmo áudio, usando o F0 original. Teremos, portanto, que lidar com os problemas adicionais de estimativa de tom, incluindo a decisão vozeada ou não vozeada. Essa abordagem é semelhante à do codificador LPC10 (exceto que neste não quantizamos os coeficientes).

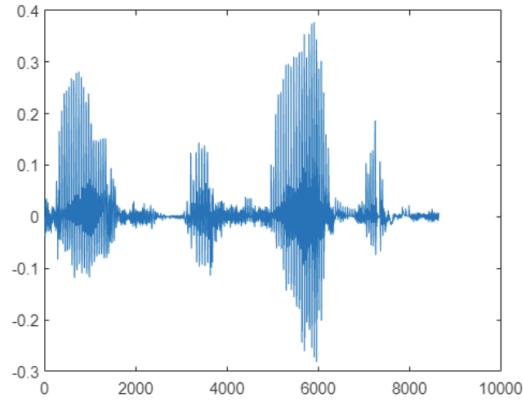


Fig. 49 - Gráfico correspondente ao áudio “speech.wav”

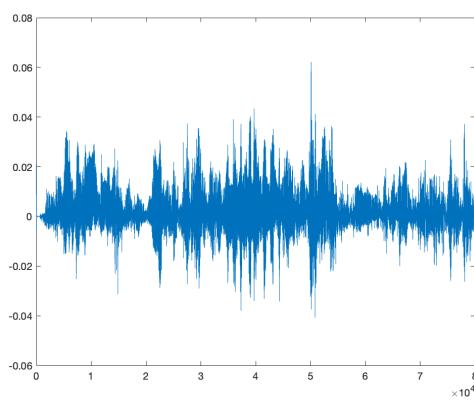


Fig. 50- Gráfico correspondente ao áudio “fala.wav”

O áudio sintético resultante é inteligível. Apresenta as mesmas características da original. É, portanto, acusticamente semelhante ao original, exceto pelo buzzyness adicional que foi adicionado pelo modelo LP nas figuras 51 e 52.

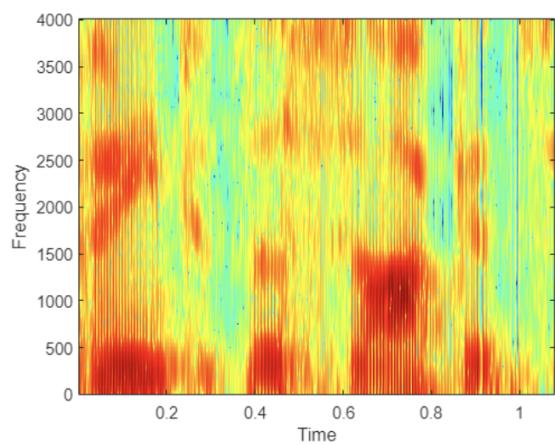


Fig. 51 - Gráfico correspondente ao áudio “speech.wav”

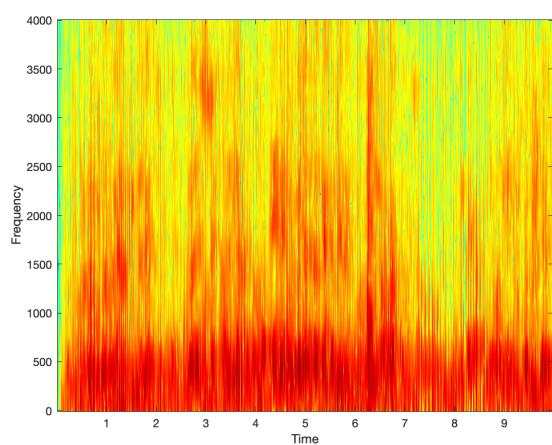


Fig. 52- Gráfico correspondente ao áudio “fala.wav”