



Universidade Federal de Campina Grande
Centro de Engenharia Elétrica e Informática
Departamento de Engenharia Elétrica
Processamento Digitais de Sinais
Prof. Dr. Edmar Candeia Gurjão
Aluno: Antonio Felipe de Melo Neto | Matrícula: 121111174

RELATÓRIO:

Implementação da FFT por decimação no tempo

1. INTRODUÇÃO

O presente trabalho, que será melhor descrito abaixo, encarrega-se de implementar a FFT (Fast Fourier Transform), ou simplesmente transformada rápida de Fourier, por decimação no tempo. Sua centralização idealizadora, rotaciona em torno da otimização do código de programação da FFT, de modo a ser implementada uma forma mais rápida e consistente de resolução dos problemas.

2. DESCRIÇÃO DA IMPLEMENTAÇÃO

Utilizando-se da linguagem de programação python, em execução no google colab, no desenvolvimento do código, se fez presente a seguinte formatação, indicada abaixo, para a resolução da FFT.

Assim como indicado pela literatura, para realizar-se o cálculo da transformada rápida, separamos o sinal em duas componentes, pares e ímpares, uma vez que conseguimos realizar a soma dos somatórios dessas duas componentes, e entregar ao usuário o cálculo da FFT completa. De maneira a garantir que o cálculo fosse realizado sem danos ao resultado, foi testado o tamanho do N indicado, uma vez que só iria retornar o valor da FFT, caso esse fosse maior que 1, lembrando-se que o N, deve ser $2^l = N$.

Quando testados os pontos que poderiam falhar, recebendo-se a resposta positivamente, deu-se continuidade ao procedimento, desta vez, realizando o cálculo da sequência de números complexos que representam os fatores de rotação, seguindo posteriormente a impressão dos valores obtidos, como também, plotagem desses, em um gráfico que relaciona a amplitude com a frequência do sinal.

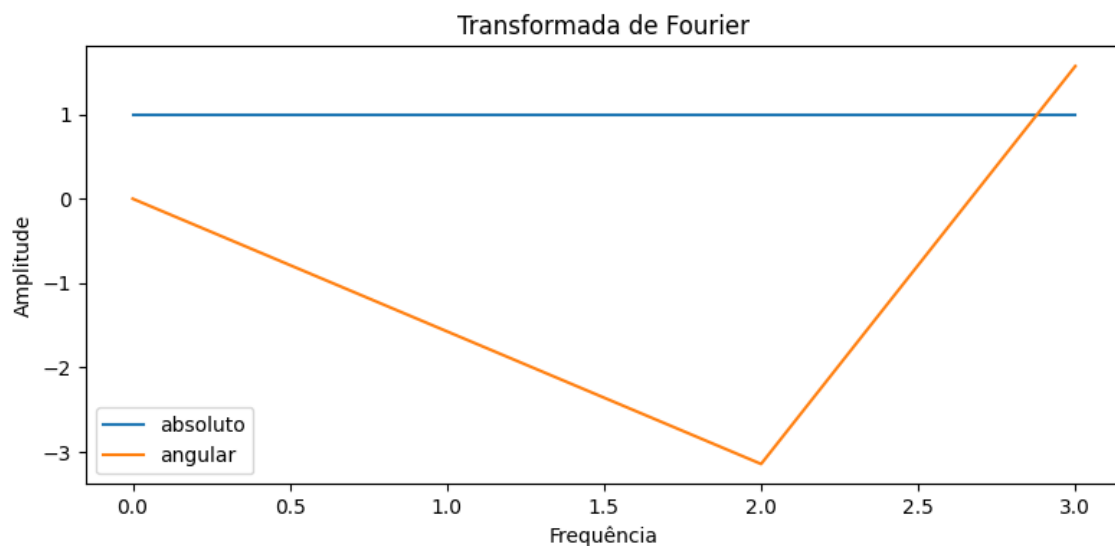
Para melhor acompanhamento e análise do trabalho realizado e resultados obtidos, segue o link de acesso ao programa que realiza o cálculo da FFT, anexado junto ao github.

3. APRESENTAÇÃO DE RESULTADOS

- Tomando como exemplo um vetor do tipo $[0, 1, 0, 0]$
- Matematicamente, após os cálculos, foram obtidos os seguintes resultados:

```
Transformada de Fourier dos dados de entrada:  
(1+0j)  
(6.123233995736766e-17-1j)  
(-1-1.2246467991473532e-16j)  
(-1.8369701987210297e-16+1j)
```

- Tomando os resultados obtidos, matematicamente, pode-se realizar a plotagem do gráfico de frequência versus Amplitude



- O código para os resultados acima discutidos, pode ser encontrado no link:
[https://github.com/AFNetto/PDS/blob/6060e4f7f03c0232679bcc838319c8de844bbbc7/FFT%20\(Tansformada%20r%C3%A1pida%20de%20Fourier\)/fft.py](https://github.com/AFNetto/PDS/blob/6060e4f7f03c0232679bcc838319c8de844bbbc7/FFT%20(Tansformada%20r%C3%A1pida%20de%20Fourier)/fft.py)