Fourier Analysis Function: furi.m

This documentation explains the custom spectral analysis script written in MATLAB and provides a Python equivalent for reference.

Function Overview

- Manual amplitude and phase spectrum calculations
- Frequency vector generation
- Power spectrum estimation

MATLAB Code Breakdown

```
[w, N] = hanning(length(x)), length(x)
ffx = fft(x)


for i = 1:N/2
    An(i) = -1/(2*N) * imag(ffx(i))
    Bn(i) =  1/(2*N) * real(ffx(i))
    phase(i) = atan(Bn(i)/An(i))
end
Cn = sqrt(An.^2 + Bn.^2)


freq = (0:Fs/N:Fs/2 - Fs/N)'


p = 2 * abs(ffx)/N
p = p(1:(N-1)/2).^2
```

Python Equivalent

```
import numpy as np


def furi(x, Fs):
    N = len(x)
```

```
    ffx = np.fft.fft(x)


    An = -1/(2*N) * np.imag(ffx[:N//2])
    Bn =  1/(2*N) * np.real(ffx[:N//2])
    phase = np.arctan2(Bn, An)


    Cn = np.sqrt(An**2 + Bn**2)
    freq = np.linspace(0, Fs/2, N//2, endpoint=False)


    p = (2 * np.abs(ffx) / N)**2
    p = p[:(N-1)//2]


    return freq, Cn, p, phase
```

Notes

Originally developed during a research internship in 2018. Converted into documented form for demonstration purposes.