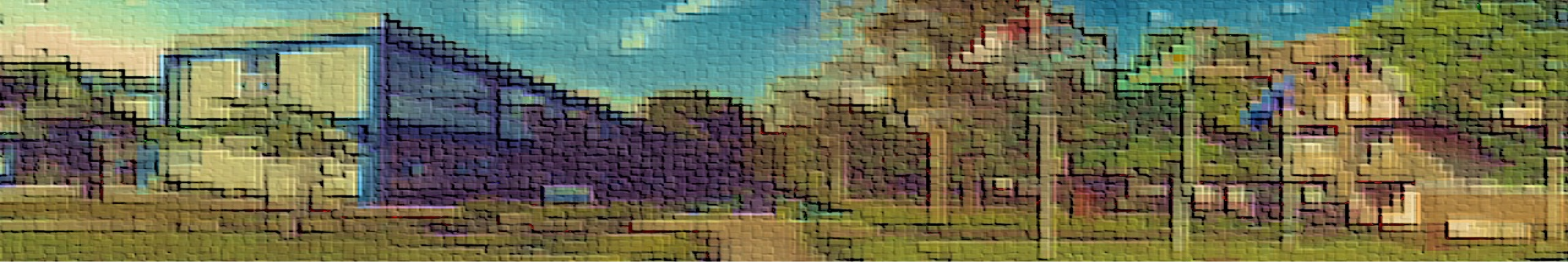


MAC317
Introdução ao Processamento de Sinais Digitais

Prof. Marcel P. Jackowski
mjack@ime.usp.br

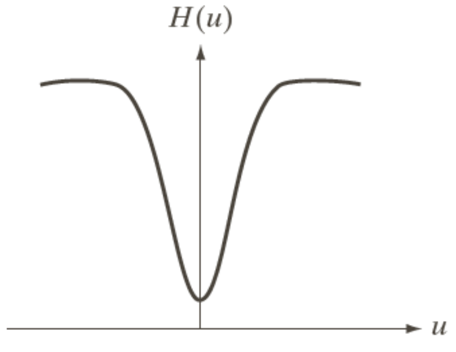
Aula #16: Janelamento e localização

Diferença de Gaussianas (DoG)

Let $H(u)$ denote the difference of Gaussian filter

$$H(u) = Ae^{-u^2/2\sigma_1^2} - Be^{-u^2/2\sigma_2^2}$$

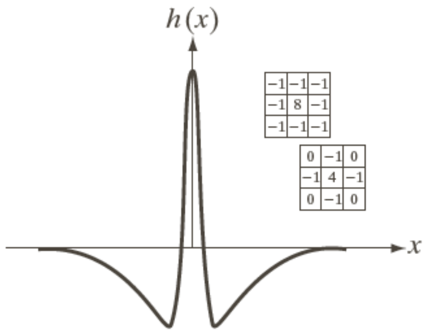
with $A \geq B$ and $\sigma_1 \geq \sigma_2$



The corresponding filter in the spatial domain

$$h(x) = \sqrt{2\pi}\sigma_1 Ae^{-2\pi^2\sigma_1^2x^2} - \sqrt{2\pi}\sigma_2 Ae^{-2\pi^2\sigma_2^2x^2}$$

Approximates the Laplacian



Filtragem homomórfica

- Can be used to remove shading effects in an image (i.e., due to uneven illumination)
  - Enhance high frequencies
  - Attenuate low frequencies but preserve fine detail.

a b

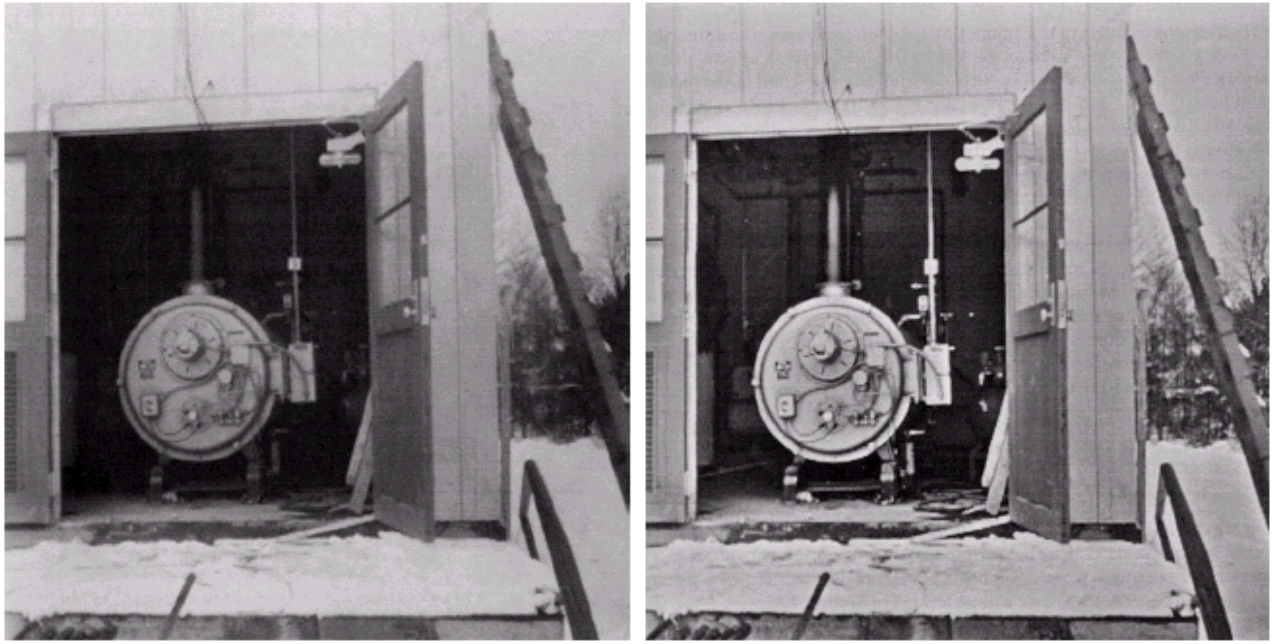


FIGURE 4.33 (a) Original image. (b) Image processed by homomorphic filtering (note details inside shelter). (Stockham.)

Filtragem homomórfica

- Consider the following model of image formation:

$$f(x, y) = i(x, y)\, r(x, y)$$

- Illumination $i(x,y)$: varies slowly and affects low frequencies mostly
- Reflection $r(x,y)$: varies faster and affects high frequencies mostly

Filtragem homomófica

Uma imagem pode ser representada através dos componentes de reflectância e luminância:

$$f(x, y) = i(x, y)r(x, y)$$

A equação acima não pode ser trabalhada diretamente no domínio da frequência uma vez que:

$$\Im\{f(x, y)\} \neq \Im(i(x, y))\Im\{r(x, y)\}$$

Filtragem homomórfica

Mas supomos que:

$$z(x,y) = \ln(f(x,y)) = \ln(i(x,y)) + \ln(r(x,y))$$

Então:

$$\Im(z(x,y)) = \Im(\ln(f(x,y))) = \Im(\ln(i(x,y))) + \Im(\ln(r(x,y)))$$

$$Z(u,v) = I(u,v) + R(u,v)$$

Filtragem homomórfica

Se processarmos $Z(u,v)$ com um filtro $H(u,v)$:

$$Z(u,v) = I(u,v) + R(u,v)$$

$$S(u,v) = H(u,v)Z(u,v)$$

$$S(u,v) = H(u,v)I(u,v) + H(u,v)R(u,v)$$

onde $S(u,v)$ é a transformada de Fourier do resultado

Filtragem homomórfica

No domínio espacial:

$$\Im^{-1}\{S(u,v)\} = s(x,y)$$

$$s(x,y) = \Im^{-1}\{H(u,v)I(u,v)\} + \Im^{-1}\{H(u,v)R(u,v)\}$$

supondo

$$i'(x,y) = \Im^{-1}\{H(u,v)I(u,v)\}$$

e

$$r'(x,y) = \Im^{-1}\{H(u,v)R(u,v)\}$$

$$s(x,y) = i'(x,y) + r'(x,y)$$

Filtragem homomórfica

Finalmente, uma vez que *z(x,y)* foi construía como o logaritmo de *f(x,y)*, a inversa de *s(x,y)* leva ao resultado desejado:

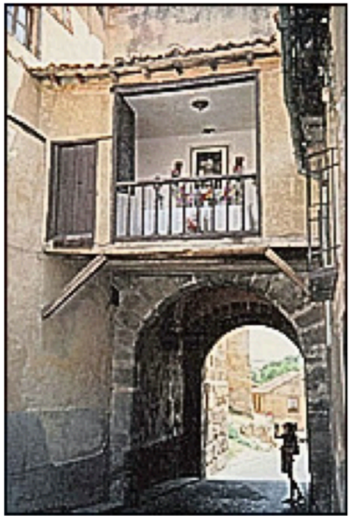
$$
\begin{aligned}
g(x,y) &= e^{\left[s(x,y)\right]} \\
&= e^{\left[i^{'}(x,y)+r^{'}(x,y)\right]} \\
&= e^{i^{'}(x,y)}e^{r^{'}(x,y)}
\end{aligned}
$$

Exemplo

Original Picture

Homomorphic Filter

Exemplos

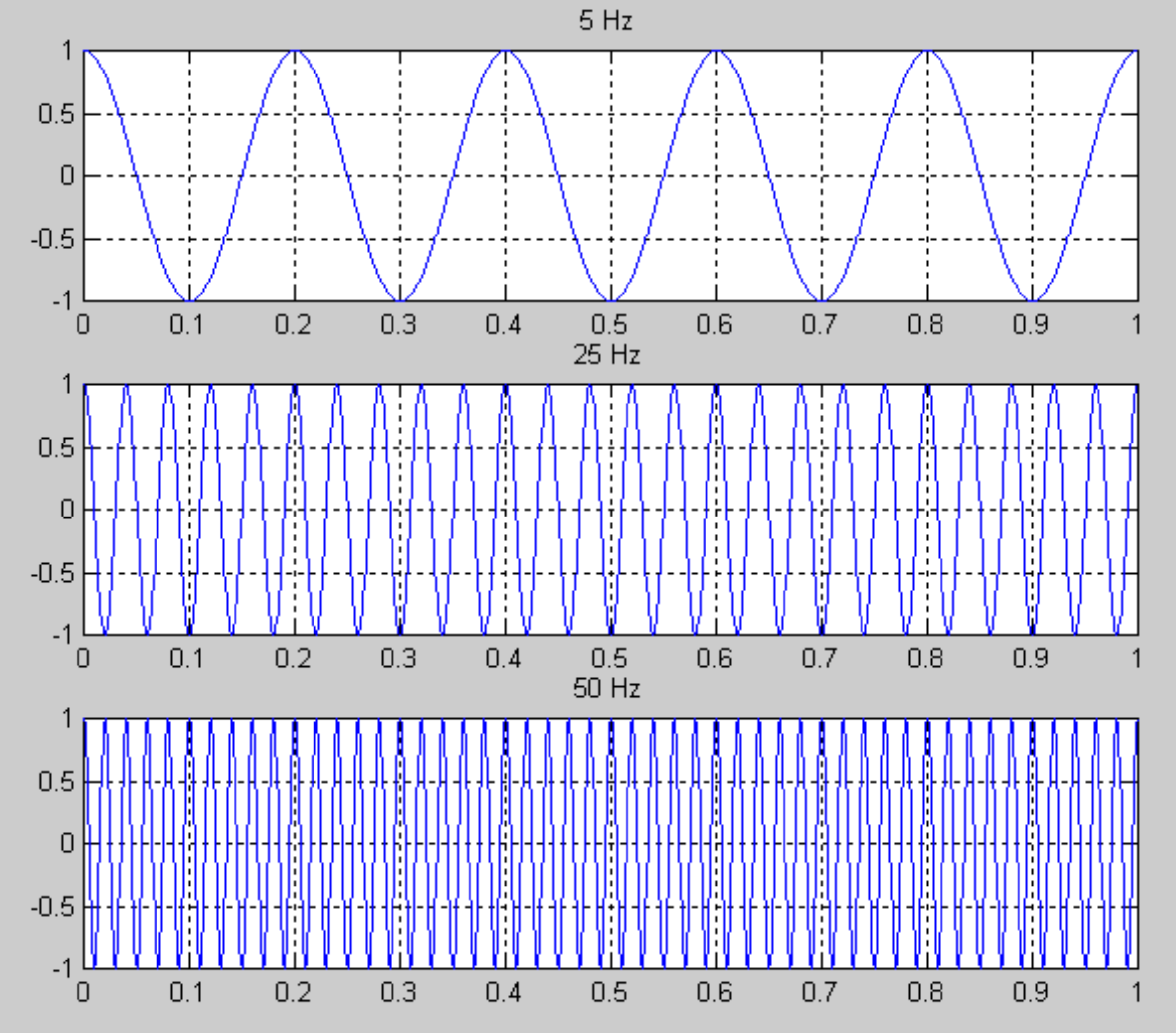


Exemplos de sinais simples

$$f_1(t) = \cos(2\pi \cdot 5 \cdot t)$$

$$f_2(t) = \cos(2\pi \cdot 25 \cdot t)$$

$$f_3(t) = \cos(2\pi \cdot 50 \cdot t)$$

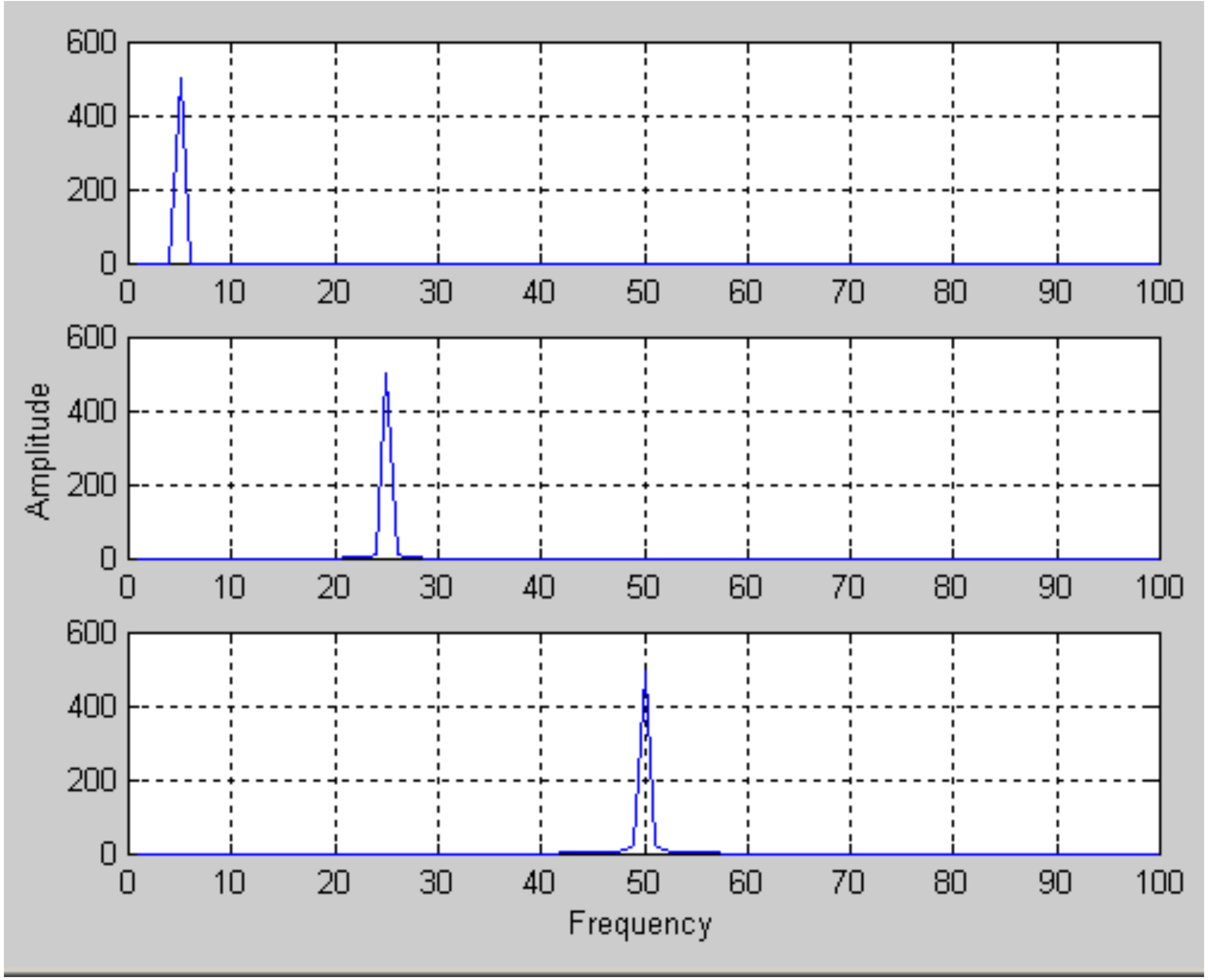


Exemplos de mapas de frequências

$F_1(u)$

$F_2(u)$

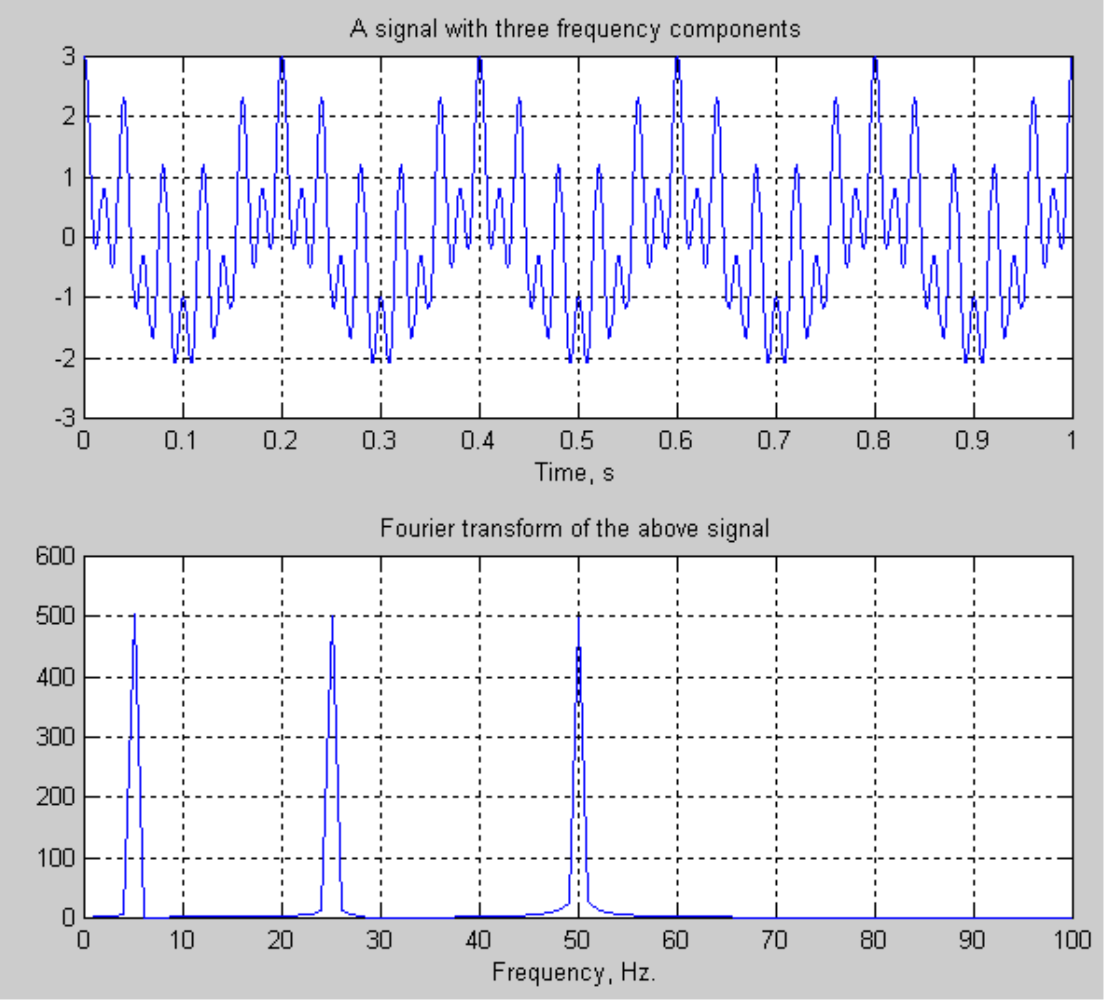
$F_3(u)$



Análise Fourier

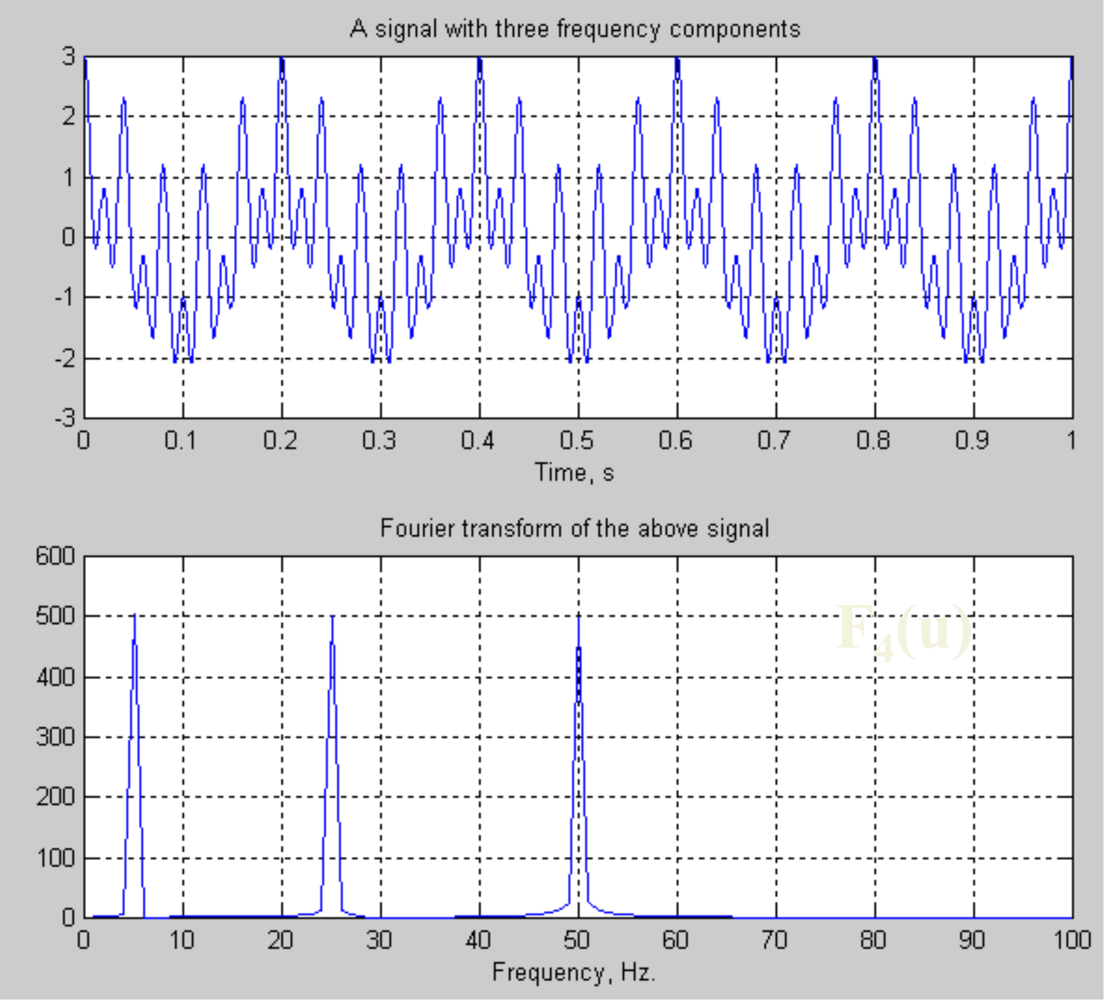
$$f_4(t) = \cos(2\pi \cdot 5 \cdot t) + \cos(2\pi \cdot 25 \cdot t) + \cos(2\pi \cdot 50 \cdot t)$$

$F_4(u)$



Limitações

$$f_4(t) = \cos(2\pi \cdot 5 \cdot t) + \cos(2\pi \cdot 25 \cdot t) + \cos(2\pi \cdot 50 \cdot t)$$

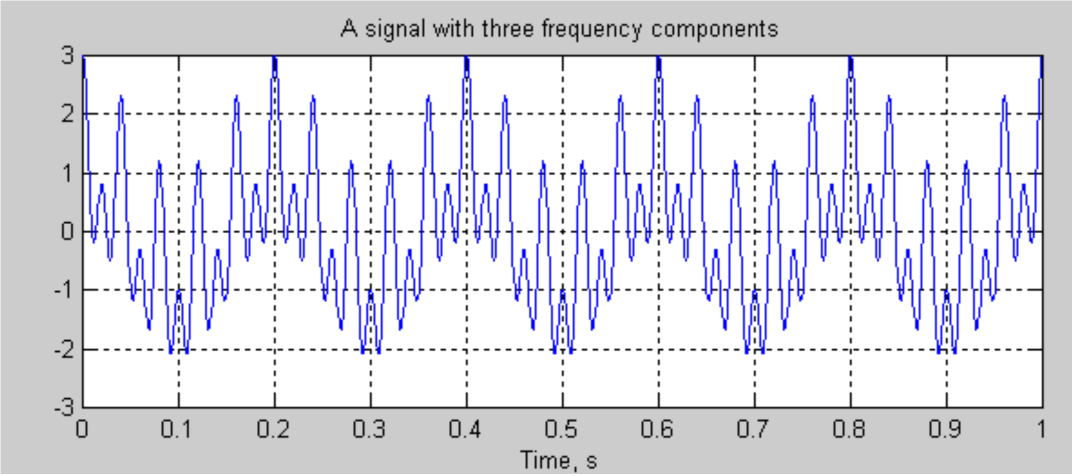


DFT provides excellent localization in the frequency domain but poor localization in the time domain.

Sinais estacionários

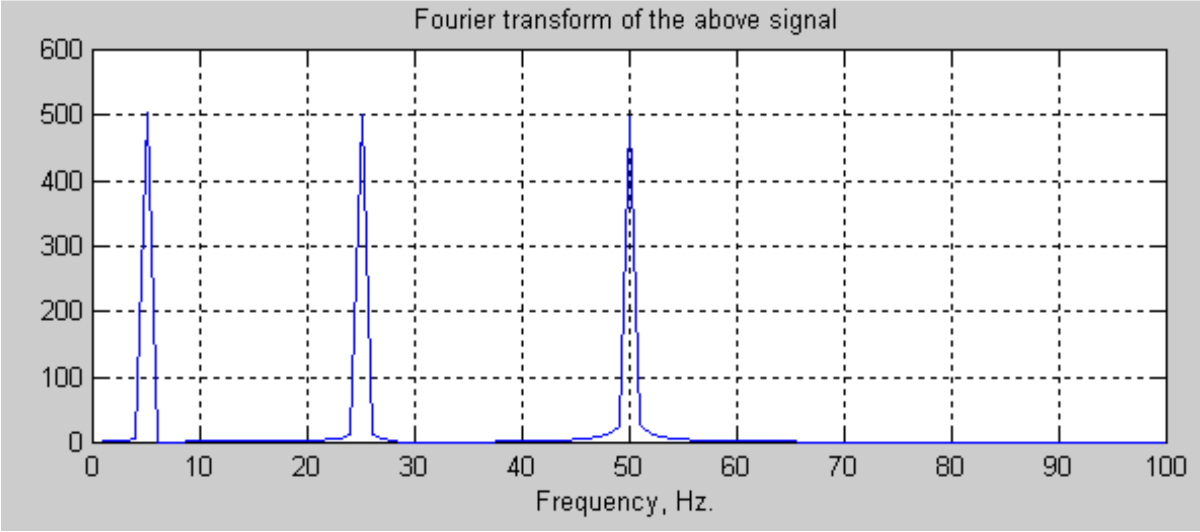
Stationary signal (non-varying frequencies):

$f_4(t)$



Three frequency components, present at all times!

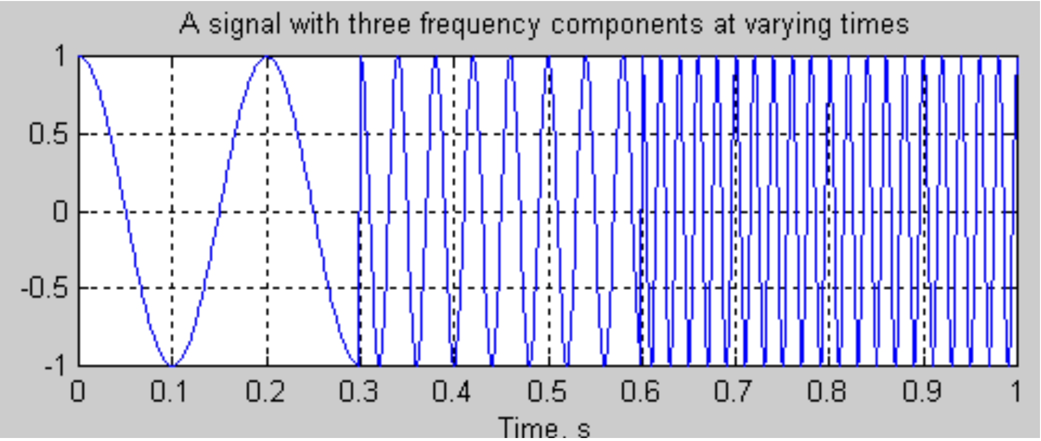
$F_4(u)$



Sinais não-estacionários

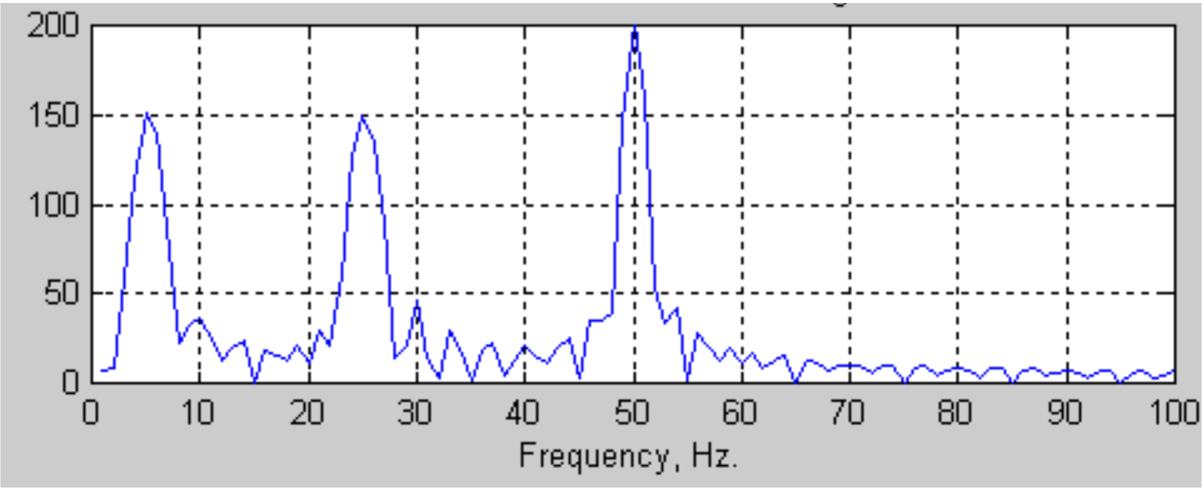
Non-stationary signal (varying frequencies):

$f_5(t)$



Three frequency components, NOT present at all times!

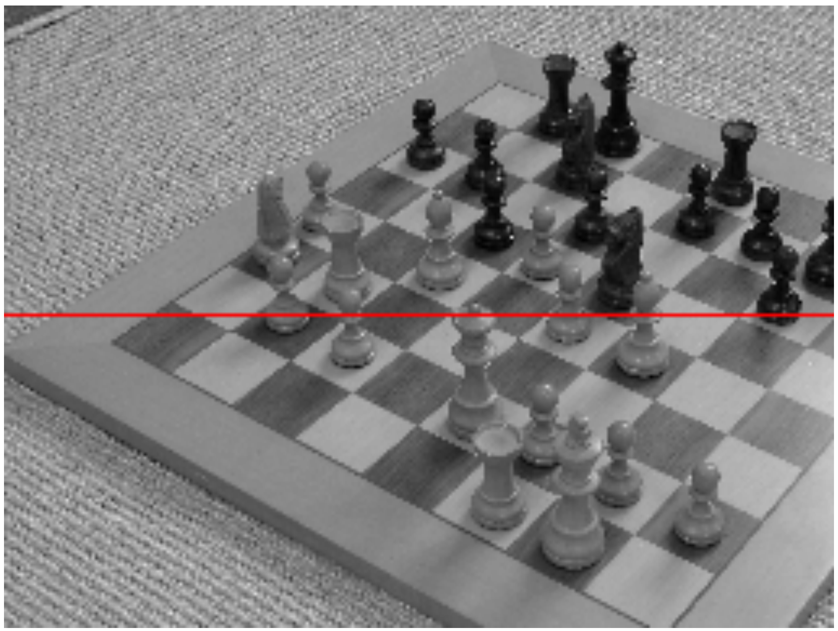
$F_5(u)$

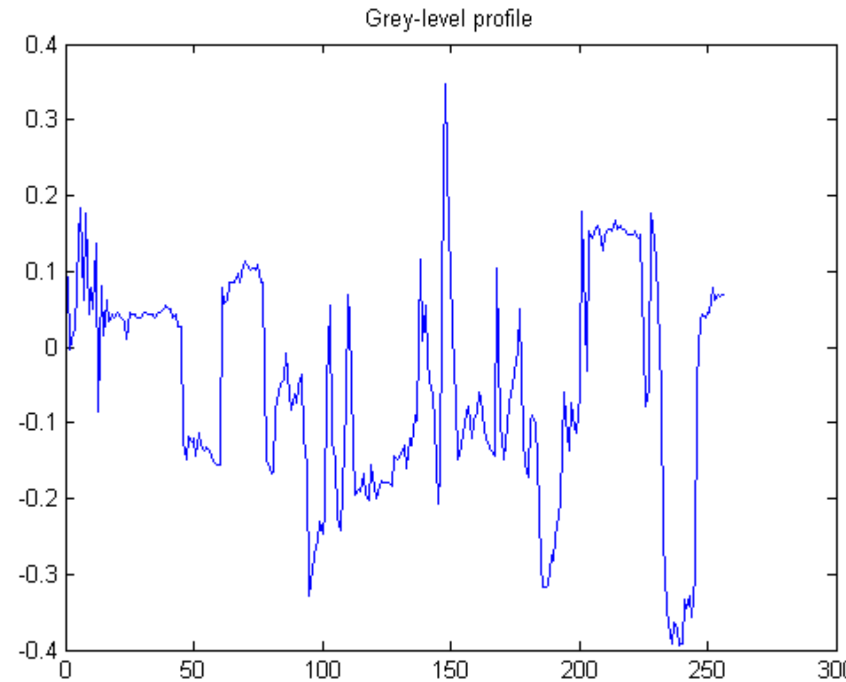


Limitações adicionais

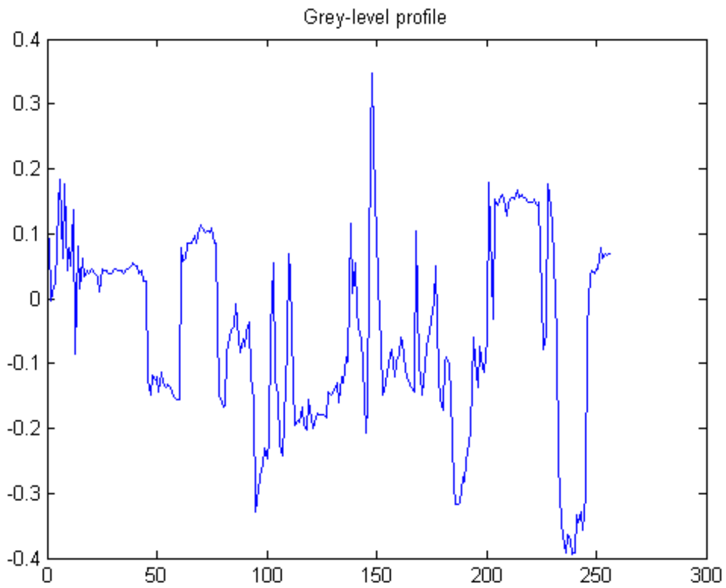
- Not very useful for analyzing time-variant, non-stationary signals.
- Not efficient for representing discontinuities or sharp corners.

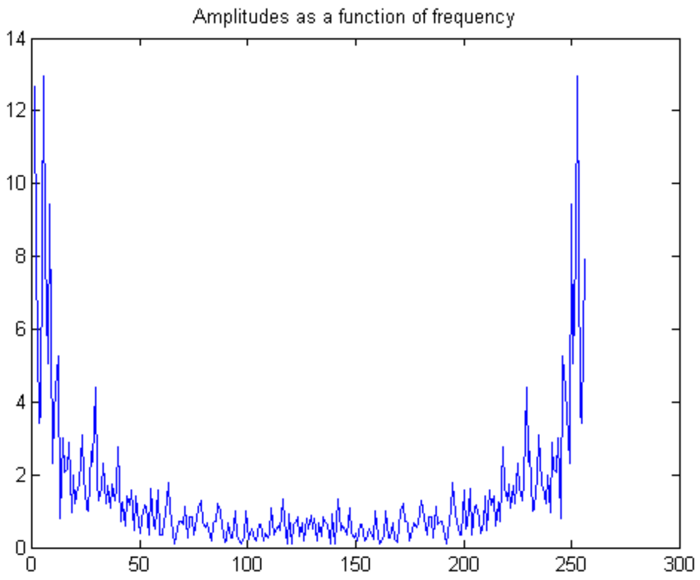
Representação de descontinuidades





Representação de descontinuidades

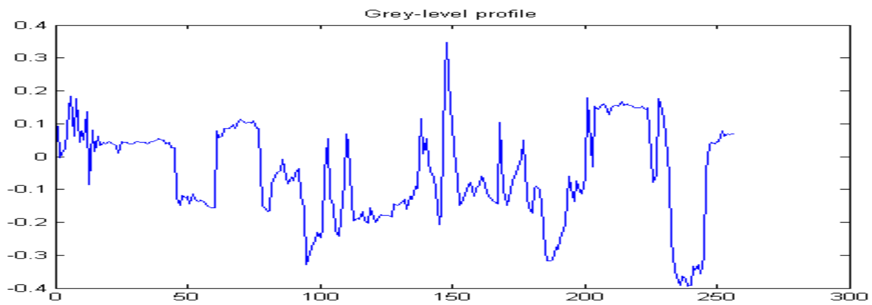


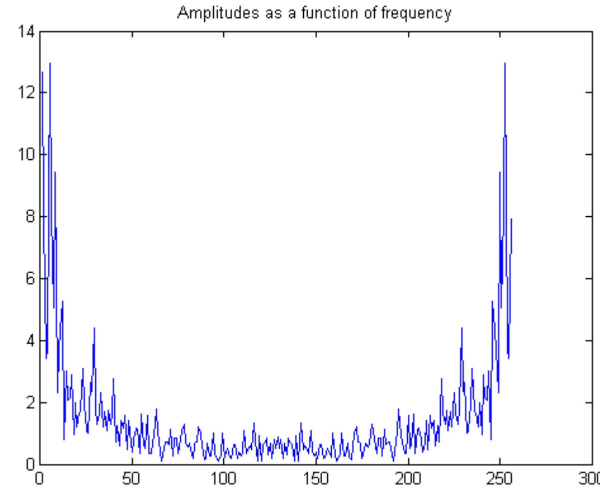


$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) e^{\frac{-j2\pi ux}{N}}, u = 0, 1, \ldots, N-1$$

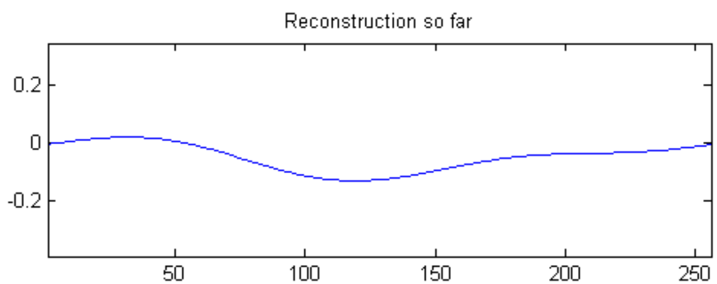
Reconstrução

Original





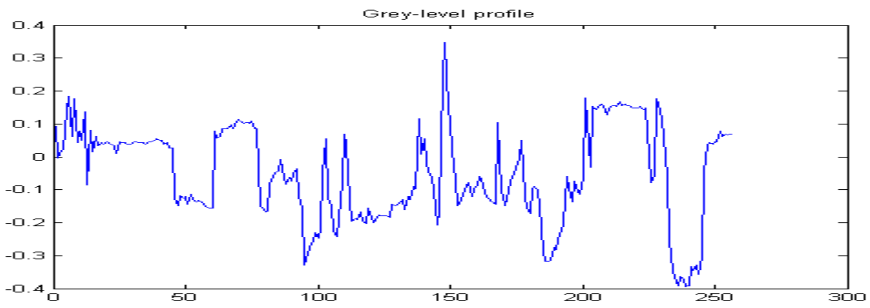
Reconstructed

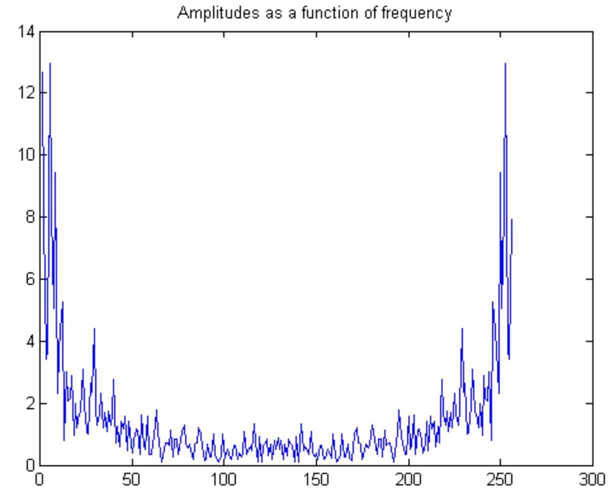


$$f(x) = \sum_{u=0}^{1} F(u) e^{\frac{j2\pi ux}{N}}, x = 0, 1, \ldots, N-1$$

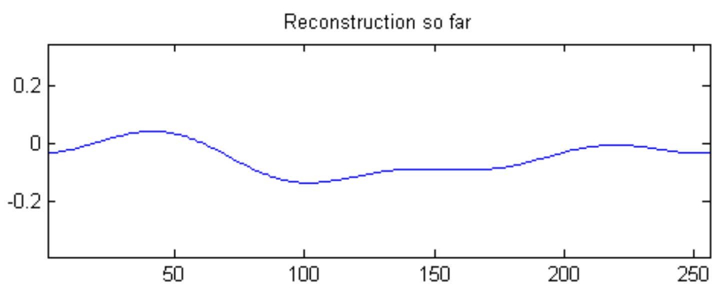
Reconstrução

Original





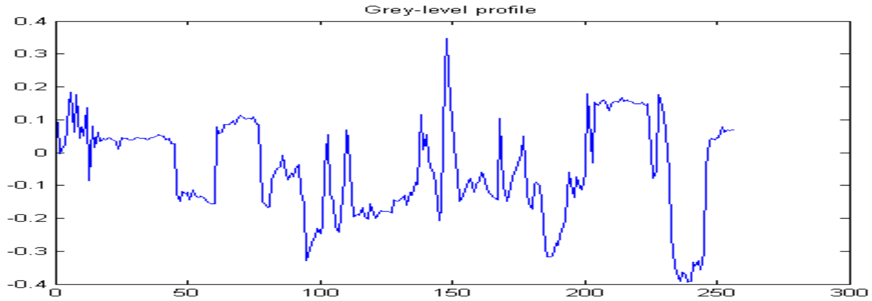
Reconstructed

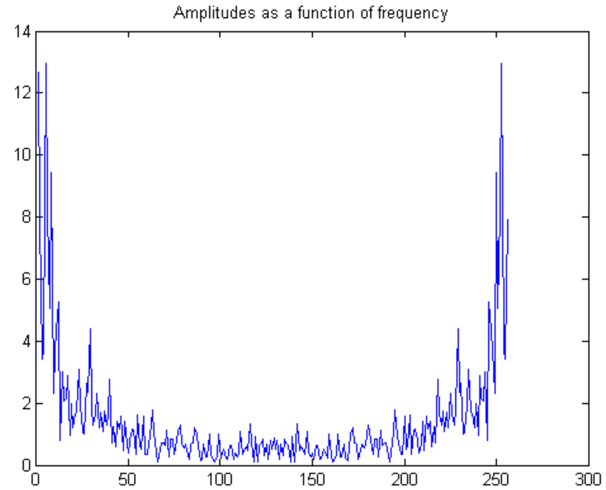


$$f(x) = \sum_{u=0}^{2} F(u) e^{\frac{j2\pi ux}{N}}, x = 0, 1, \ldots, N-1$$

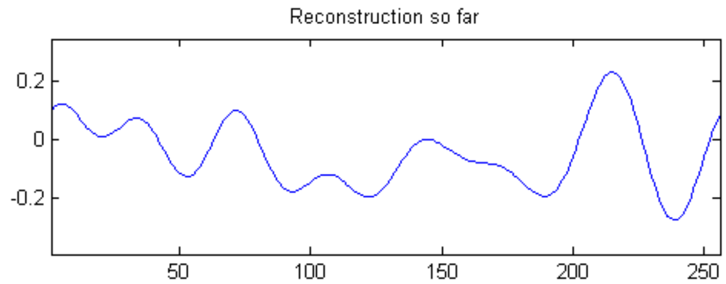
Reconstrução

Original





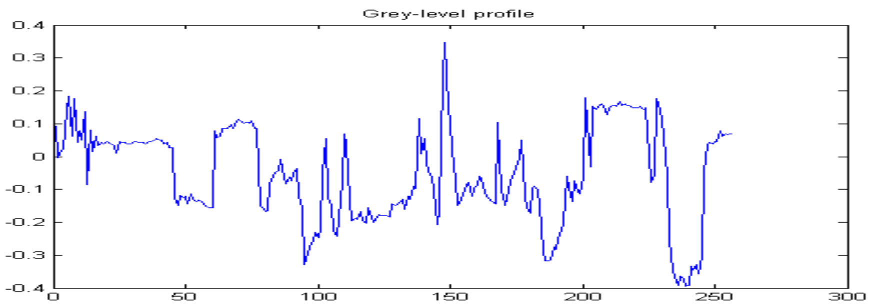
Reconstructed

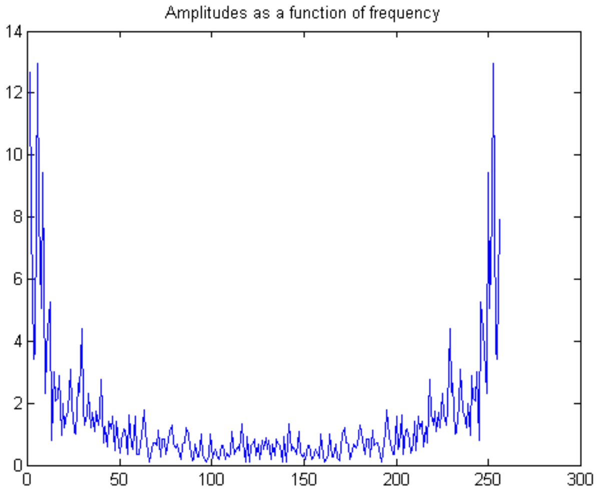


$$f(x) = \sum_{u=0}^{7} F(u) e^{\frac{j2\pi ux}{N}}, x = 0, 1, \ldots, N-1$$

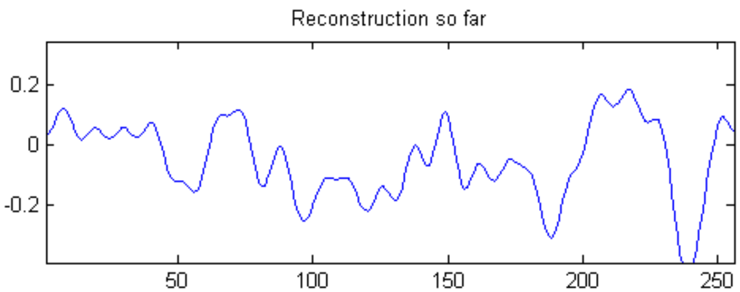
Reconstrução

Original





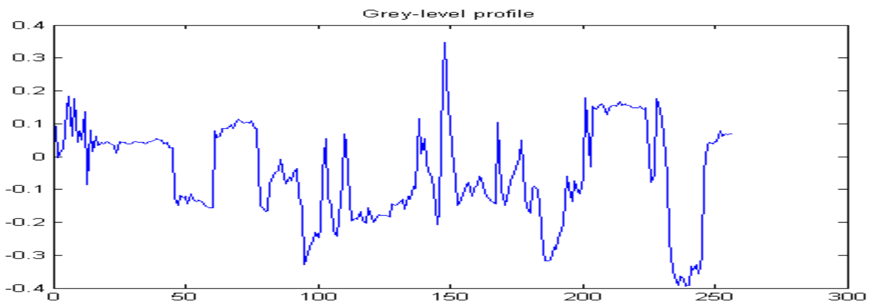
Reconstructed



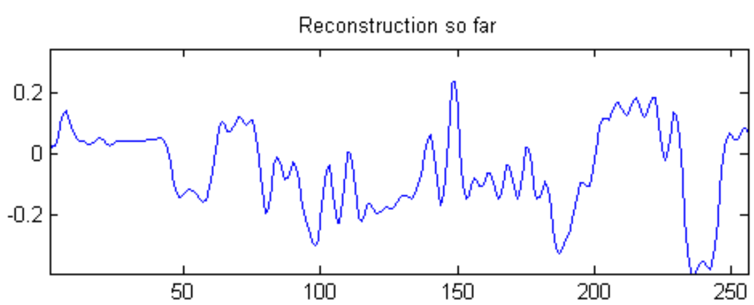
$$f(x) = \sum_{u=0}^{23} F(u) e^{\frac{j2\pi ux}{N}}, x = 0, 1, \ldots, N-1$$

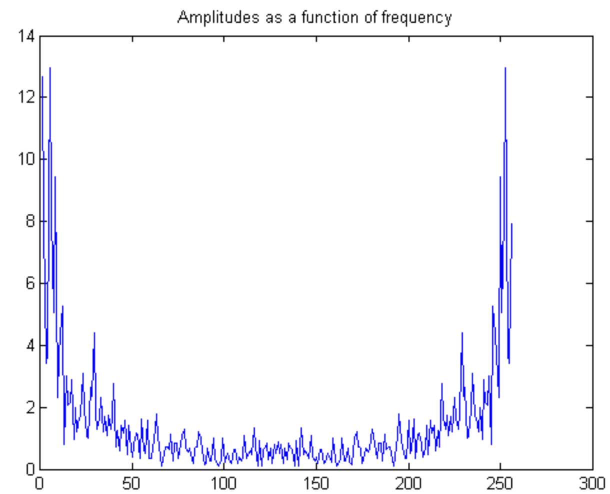
Reconstrução

Original



Reconstructed

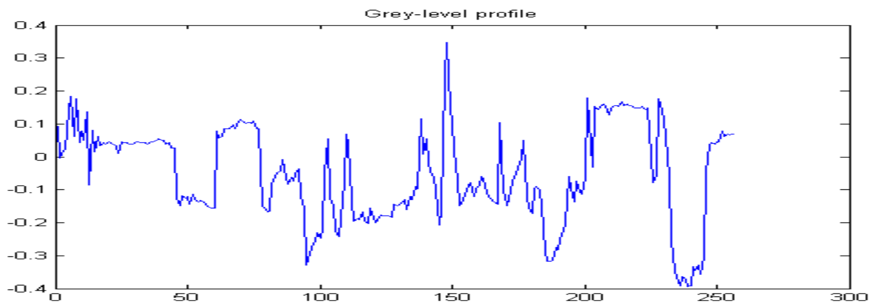


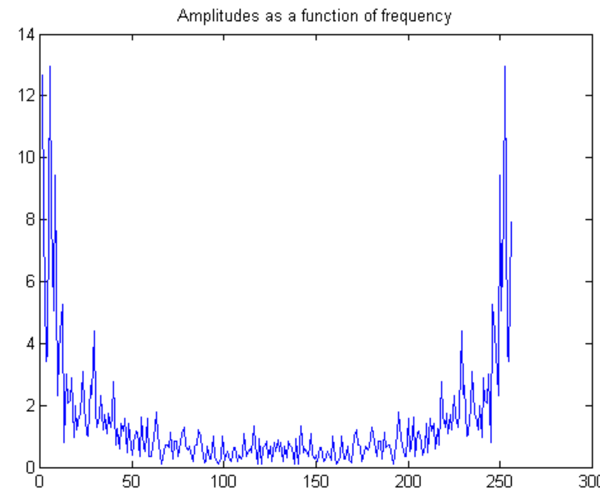


$$f(x) = \sum_{u=0}^{39} F(u) e^{\frac{j2\pi ux}{N}}, x = 0, 1, \ldots, N-1$$

Reconstrução

Original



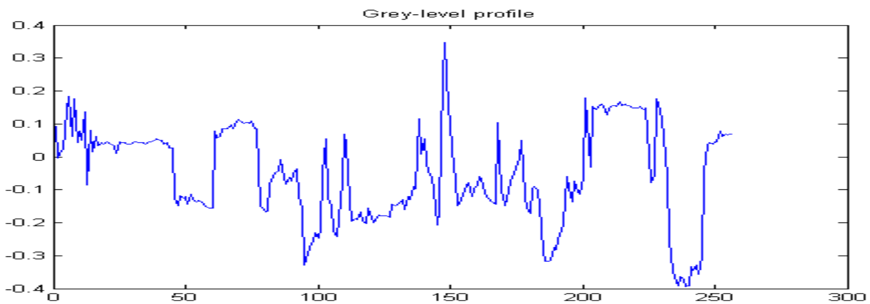


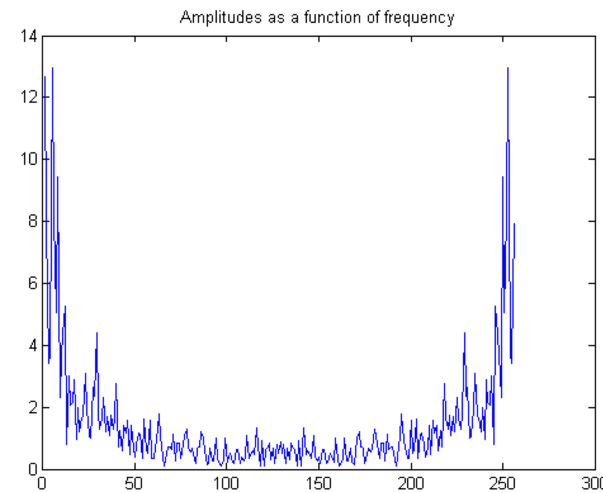
Reconstructed

$$f(x) = \sum_{u=0}^{63} F(u) e^{\frac{j2\pi ux}{N}}, x = 0, 1, \ldots, N-1$$

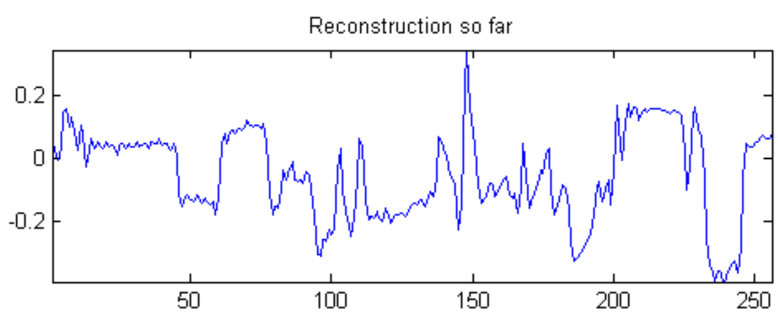
Reconstrução

Original





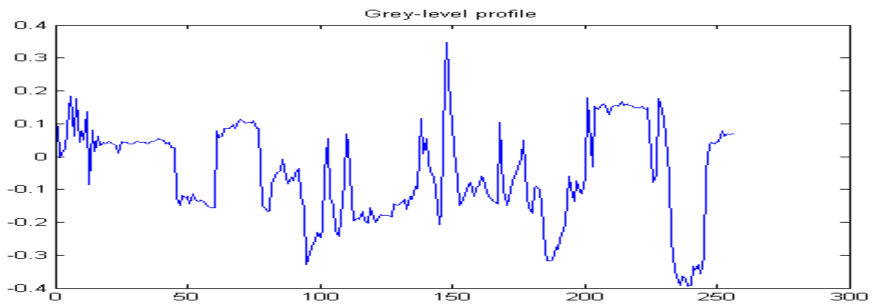
Reconstructed

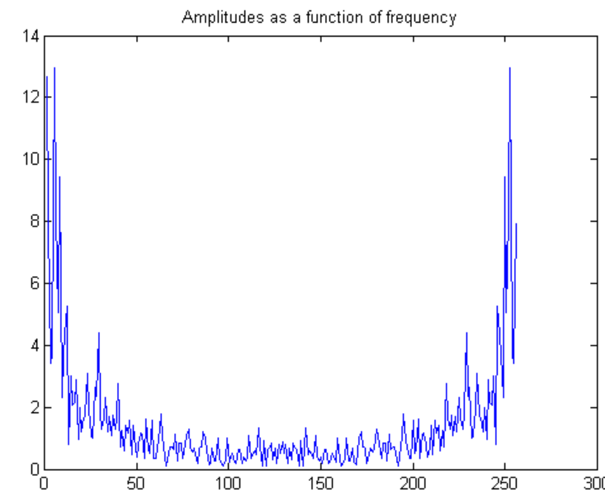


$$f(x) = \sum_{u=0}^{95} F(u)e^{\frac{j2\pi ux}{N}}, x = 0, 1, \ldots, N-1$$

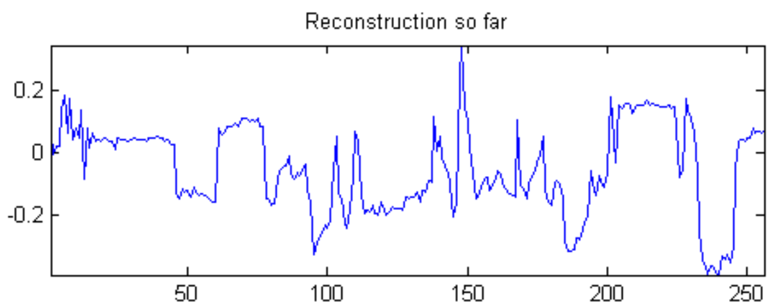
Reconstrução

Original





Reconstructed

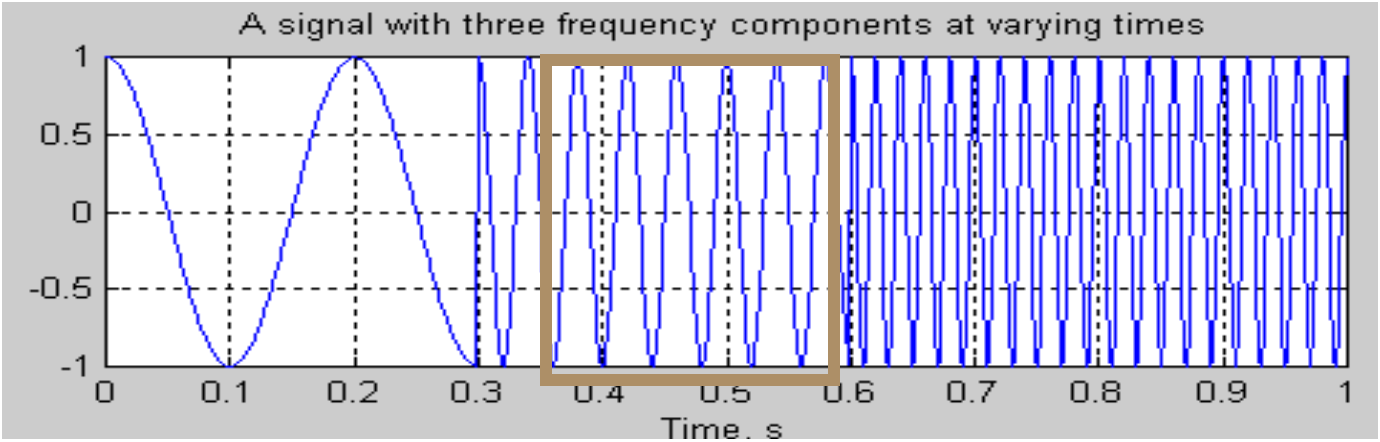


$$f(x) = \sum_{u=0}^{127} F(u) e^{\frac{j2\pi ux}{N}}, \; x = 0, 1, \ldots, N-1$$

A large number of Fourier components is needed to represent discontinuities.

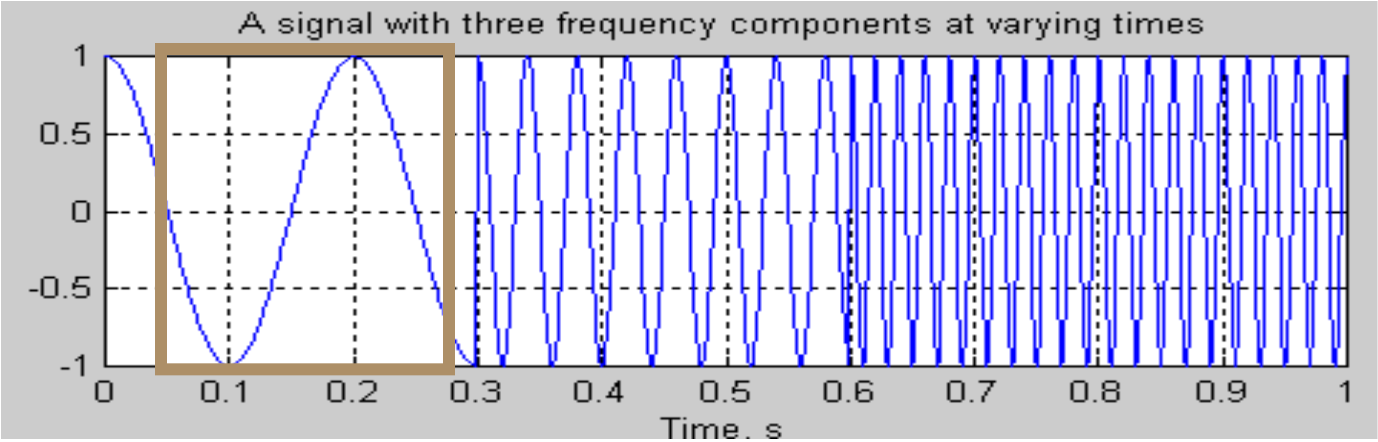
Janelamento

- Segment signal into narrow time intervals (i.e., narrow enough to be considered stationary) and take the FT of each segment.
- Each FT provides the spectral information of a separate time-slice of the signal, providing simultaneous time and frequency information.



Passos na versão janelada

(1) Choose a window of finite length
(2) Place the window on top of the signal at t=0
(3) Truncate the signal using this window
(4) Compute the FT of the truncated signal, save results.
(5) Incrementally slide the window to the right
(6) Go to step 3, until window reaches the end of the signal



Short Time Fourier Transform (STFT)

Time parameter

Frequency parameter

Signal to be analyzed

$$STFT_f^u(t',u) = \int_t \left[f(t) \cdot W(t - t') \right] \cdot e^{-j2\pi ut} dt$$

2D function

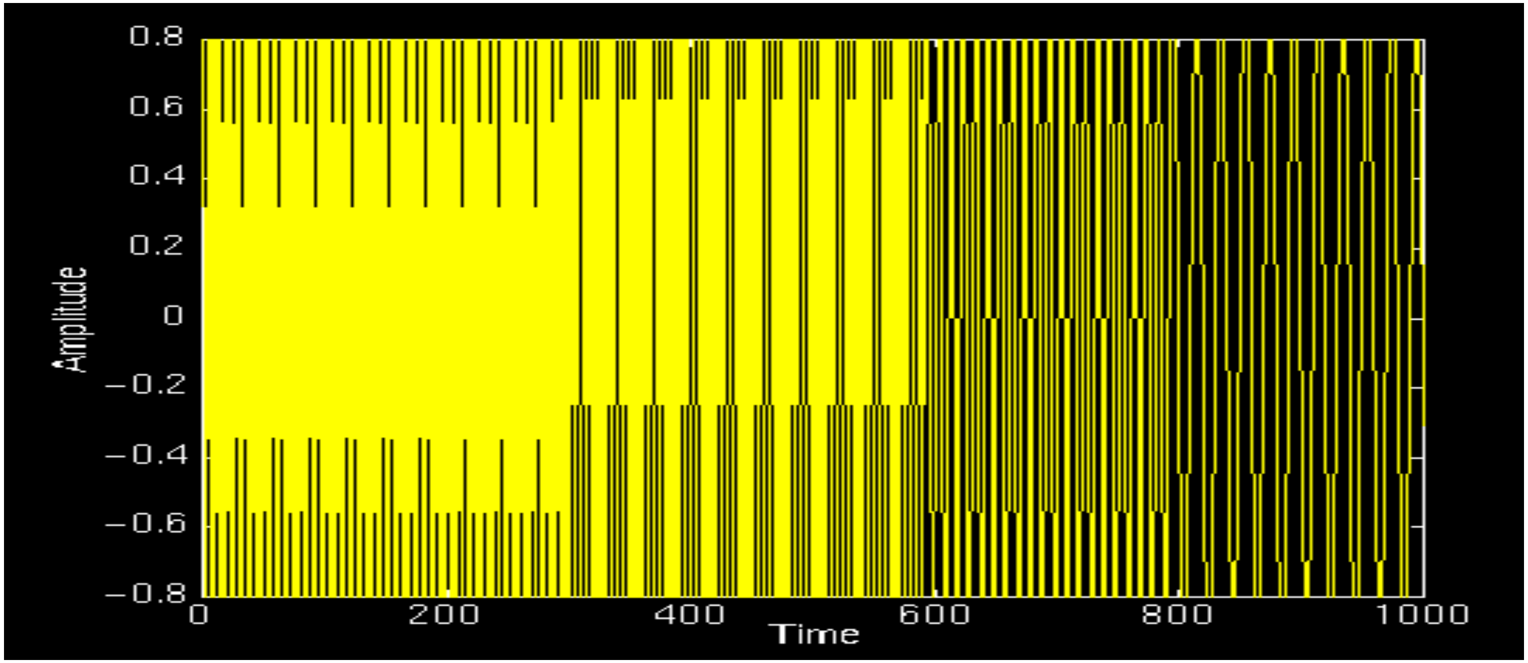
STFT of f(t): computed for each window centered at t=t'

Windowing function

Centered at t=t'

Exemplo

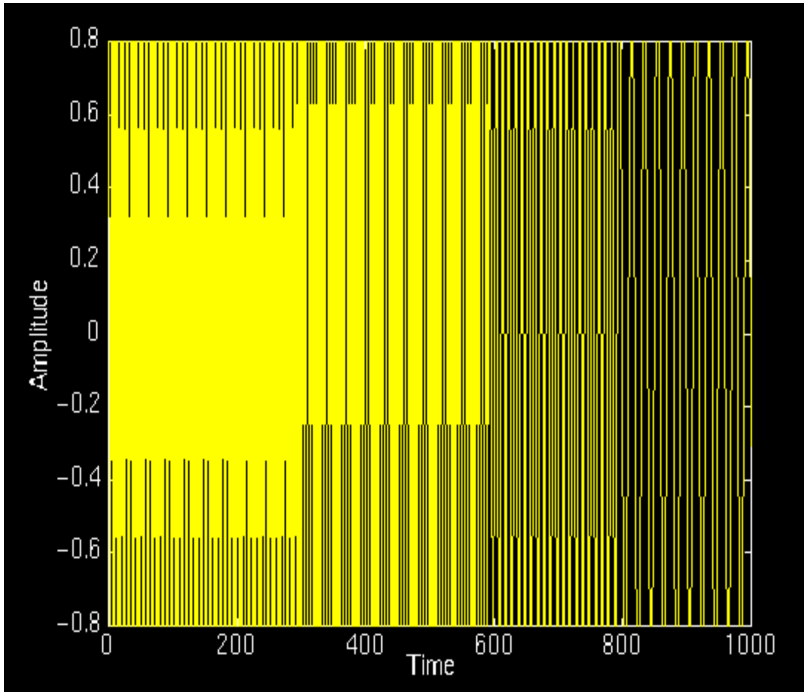
f(t)



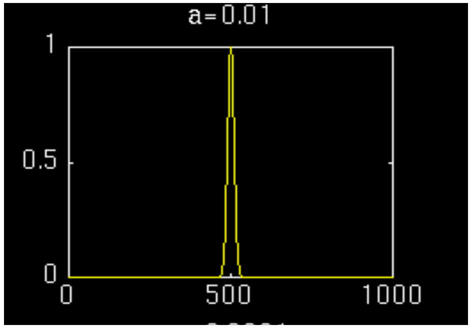
[0 – 300] ms → 75 Hz sinusoid
[300 – 600] ms → 50 Hz sinusoid
[600 – 800] ms → 25 Hz sinusoid
[800 – 1000] ms →10 Hz sinusoid

Exemplo

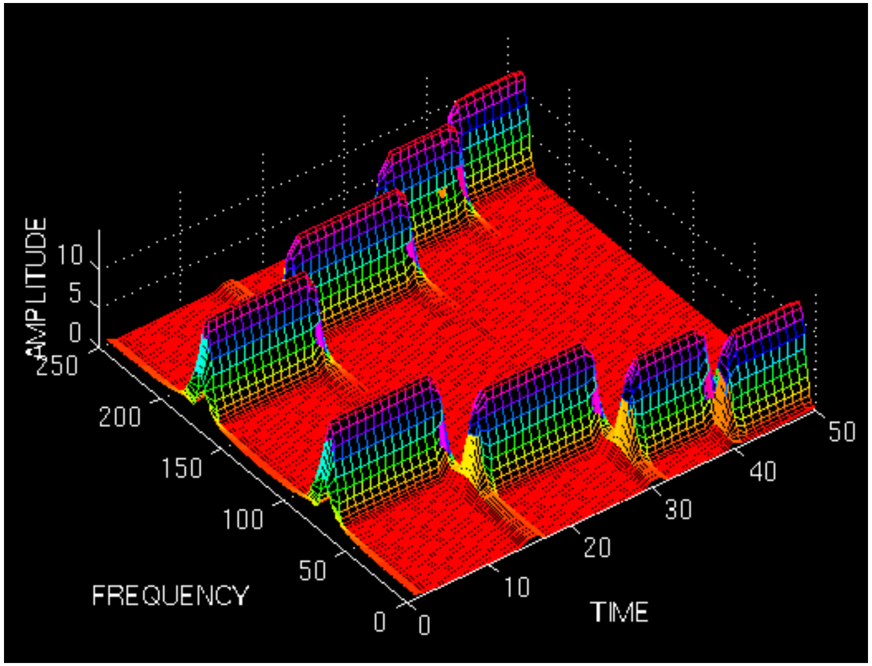
f(t)



W(t)



$$STFT_f^u(t',u)$$



scaled: t/20

Tamanho da janela

$$STFT_f^u(t',u)=\int_t \left[f(t)\cdot W(t-t')\right]\cdot e^{-j2\pi ut}dt$$

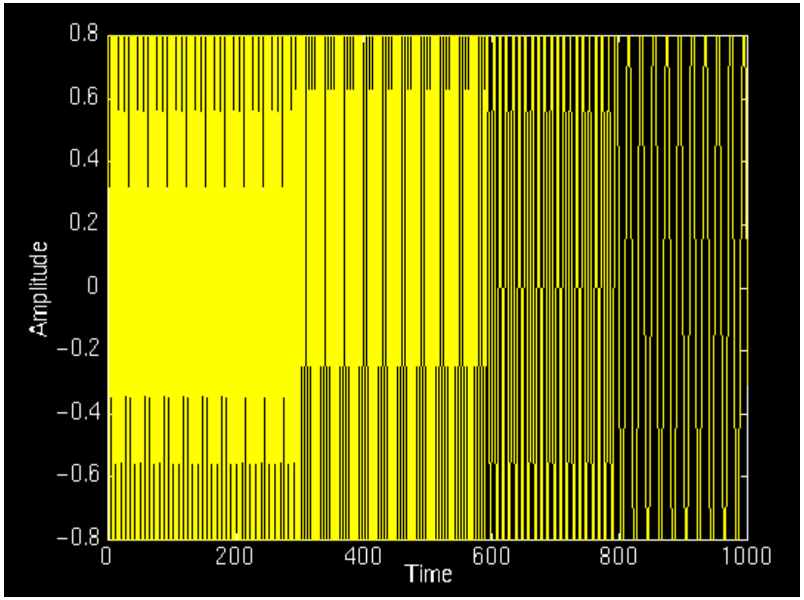
***W(t)* infinitely long:** STFT turns into FT, providing excellent frequency localization, but no time localization

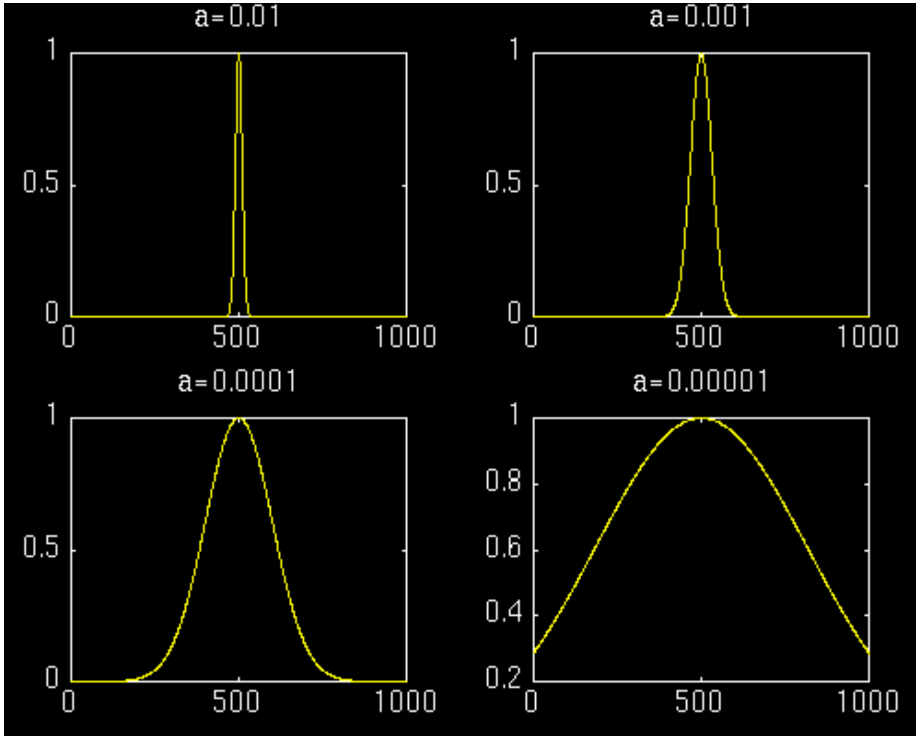
***W(t)* infinitely short:** results in the time signal (with a phase factor), providing excellent time localization but no frequency localization

$$STFT_f^u(t',u)=\int_t \left[f(t)\cdot \delta(t-t')\right]\cdot e^{-j2\pi ut}dt = f(t')\cdot e^{-jut'}$$

Exemplo

different size windows

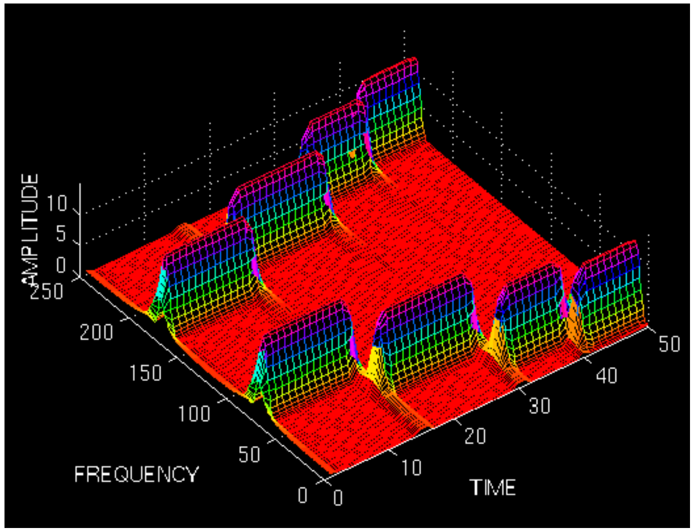


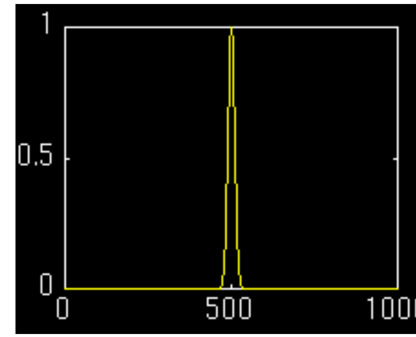


(four frequencies, non-stationary)

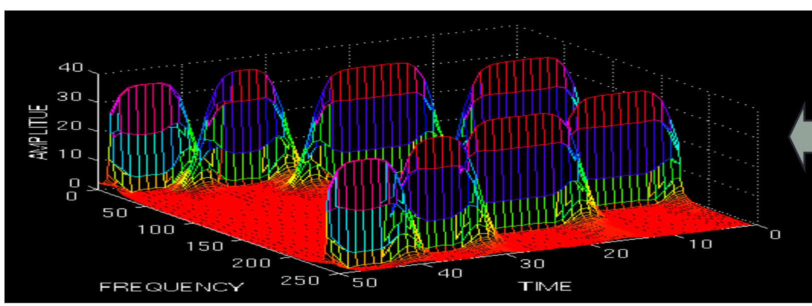
Exemplo

$$STFT_f^u(t',u)$$

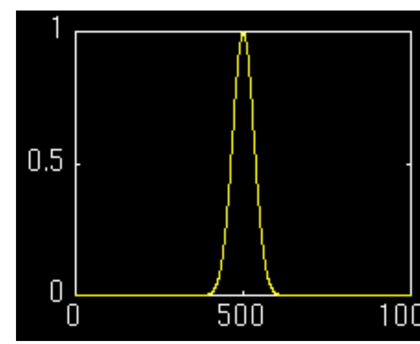




$$STFT_f^u(t',u)$$

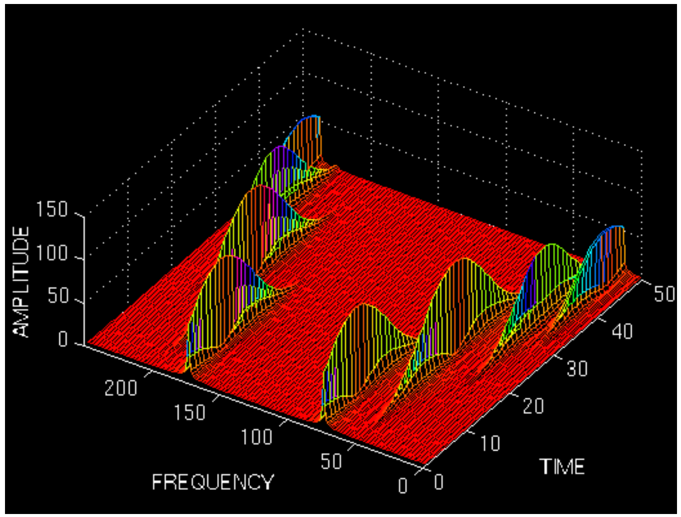


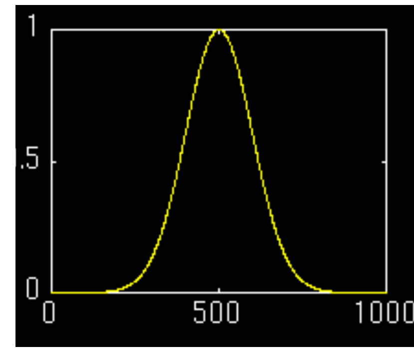
scaled: t/20



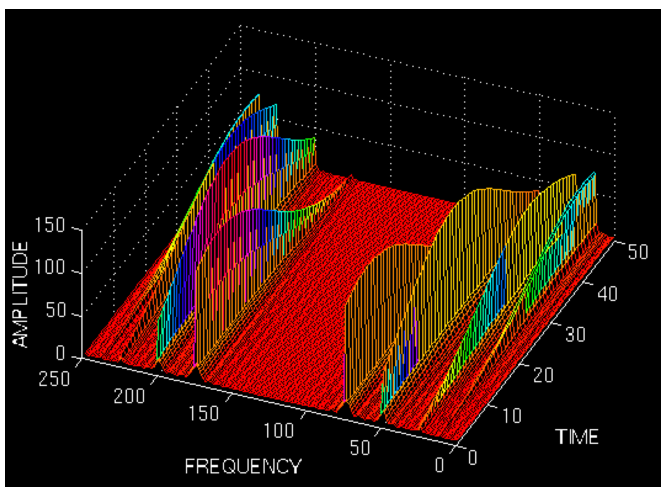
Exemplo

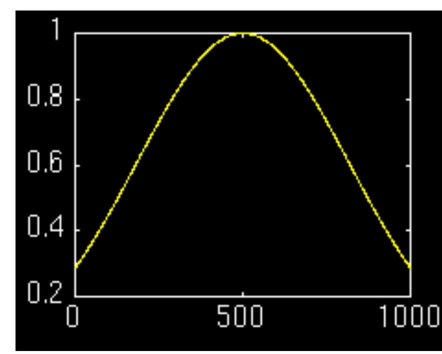
$$STFT_f^u(t',u)$$





$$STFT_f^u(t',u)$$

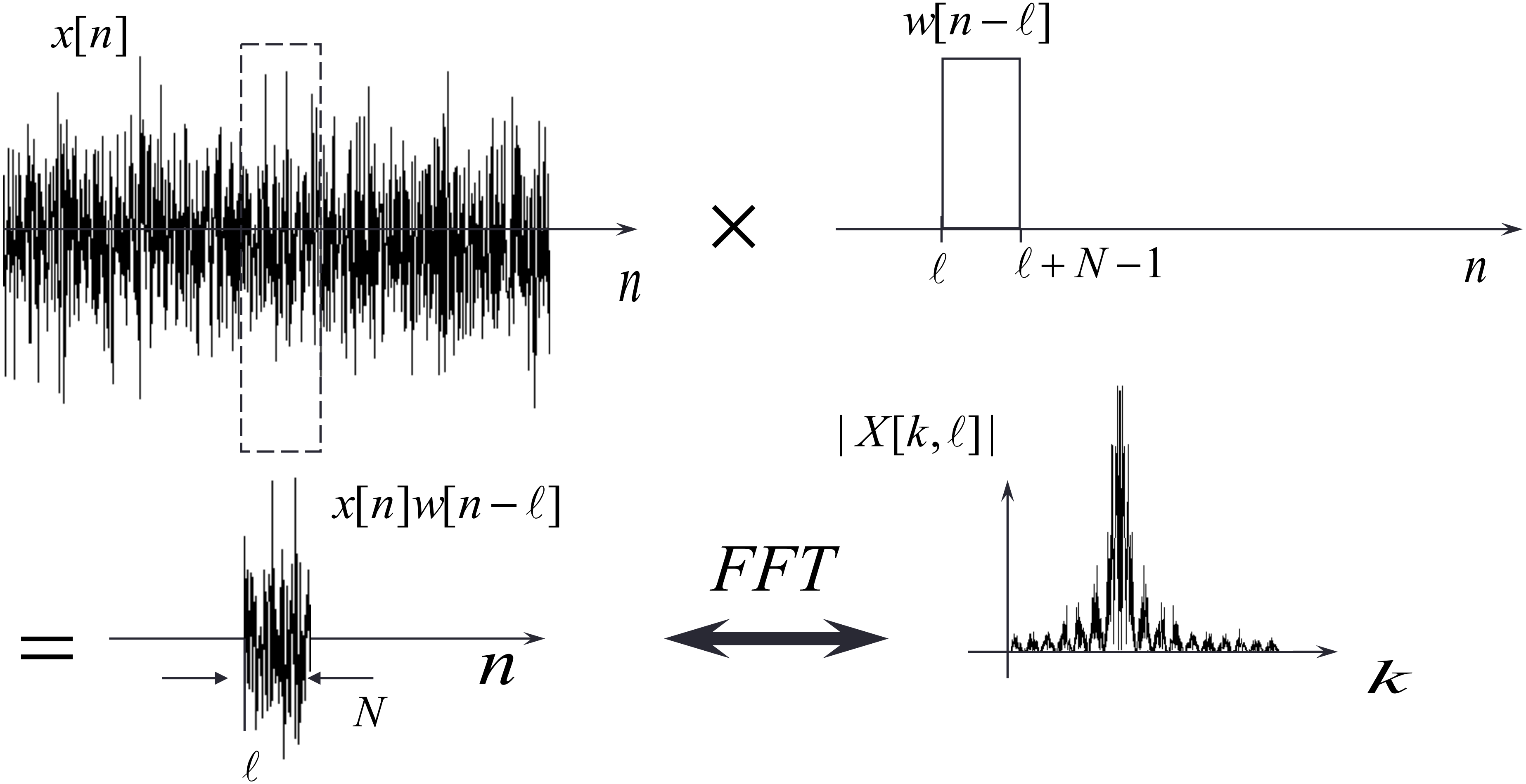




scaled: t/20

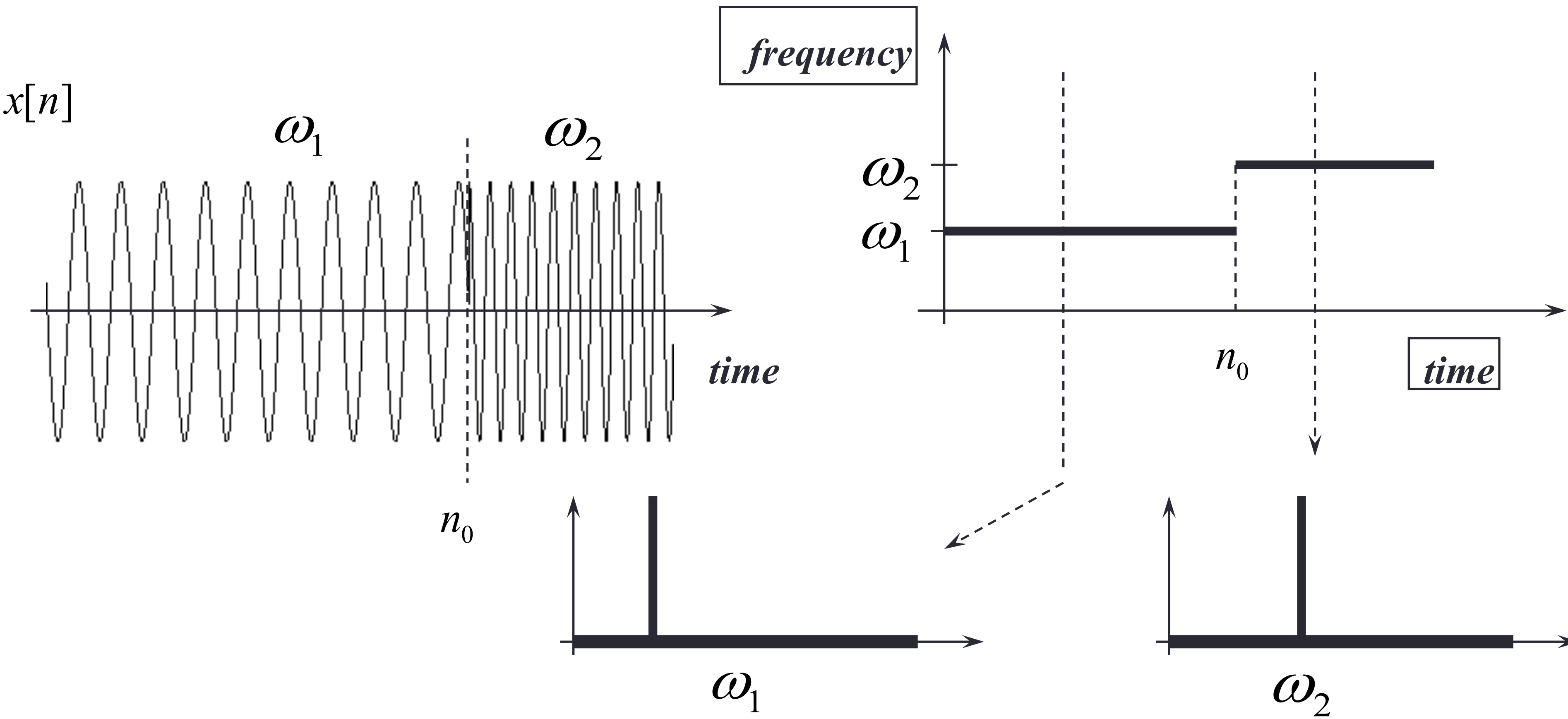
Análise espectral

- Given a signal we take the FFT on a window sliding with time

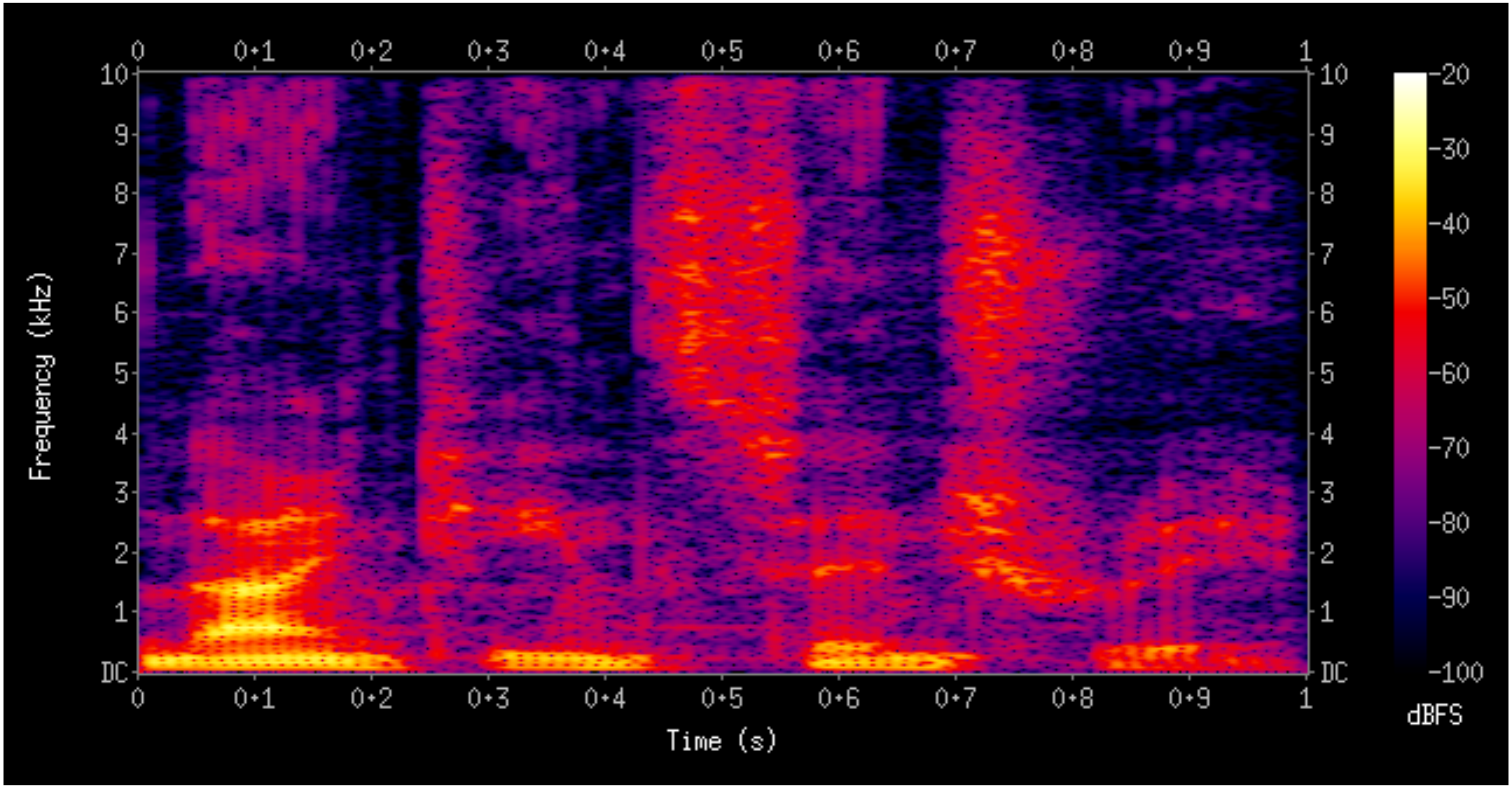


Espectograma ideal

- The evolution of the magnitude with time is called Spectrogram.

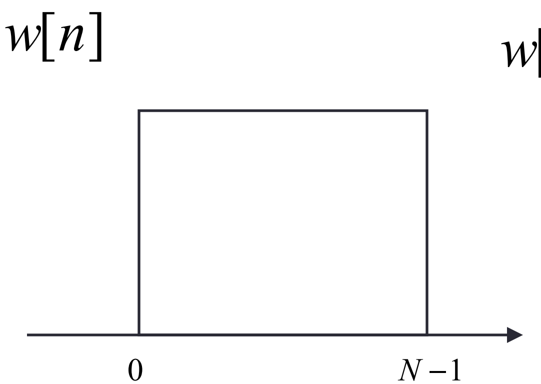


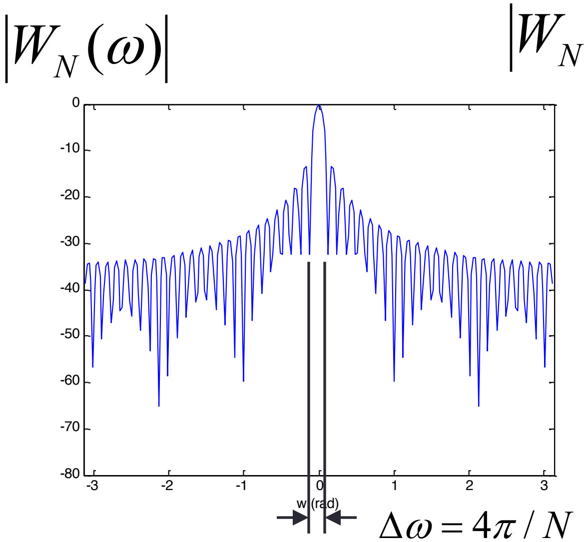
Espectograma



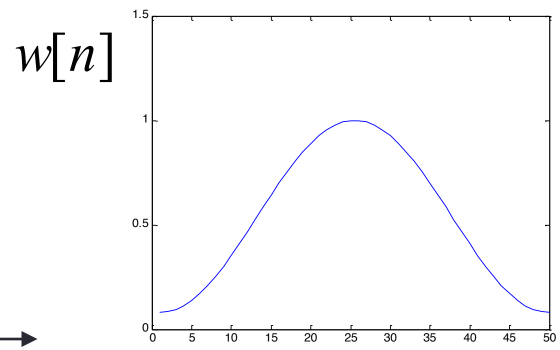
Janelas

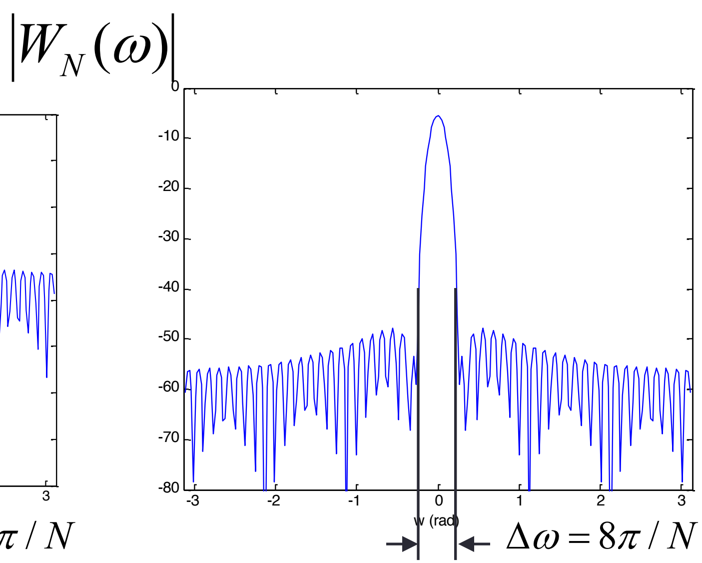
Rectangular



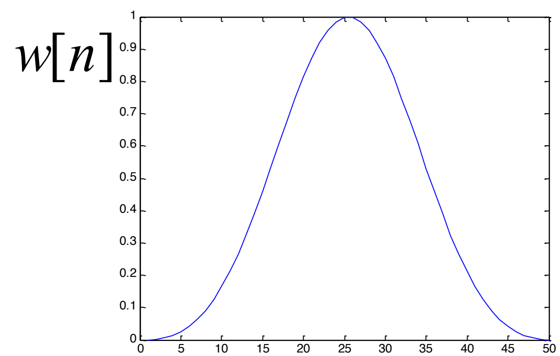


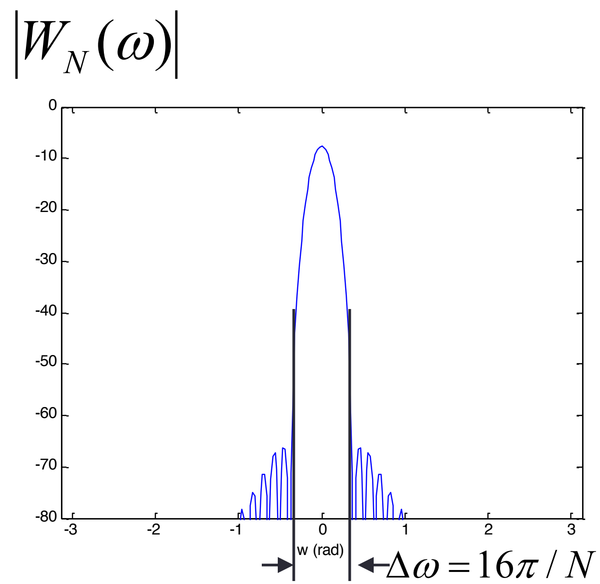
Hamming





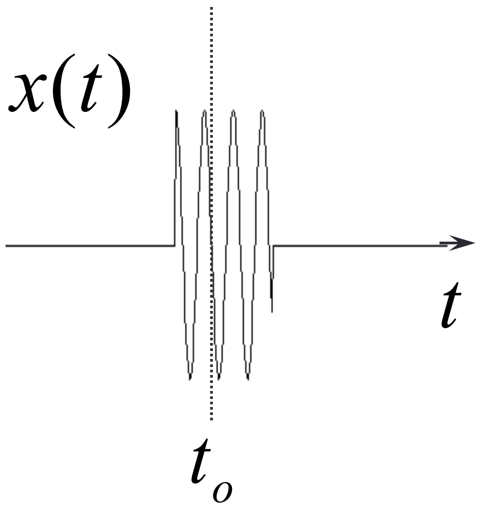
Blackman



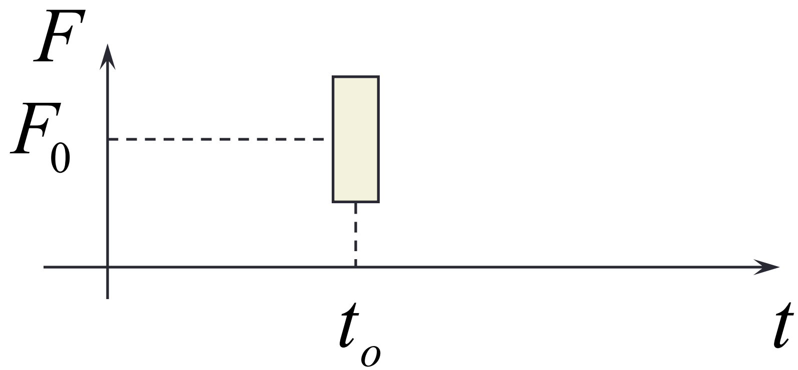


Princípio da incerteza

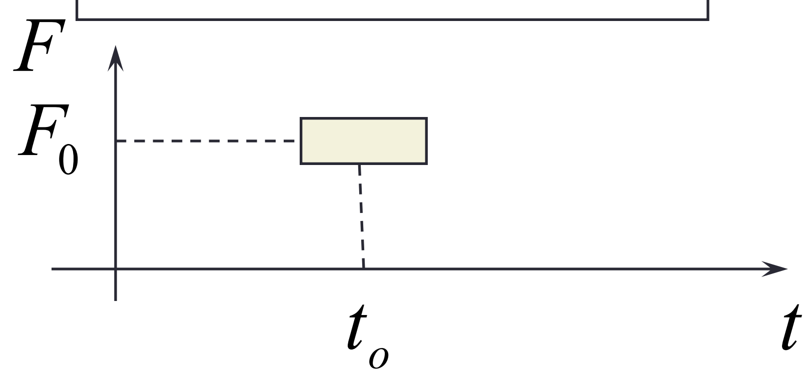
- **<u>Either</u> you have good resolution in <u>time</u> <u>or</u> in <u>frequency</u>, not both**.



either *good localization in time* ...



... or *good localization in freq.*

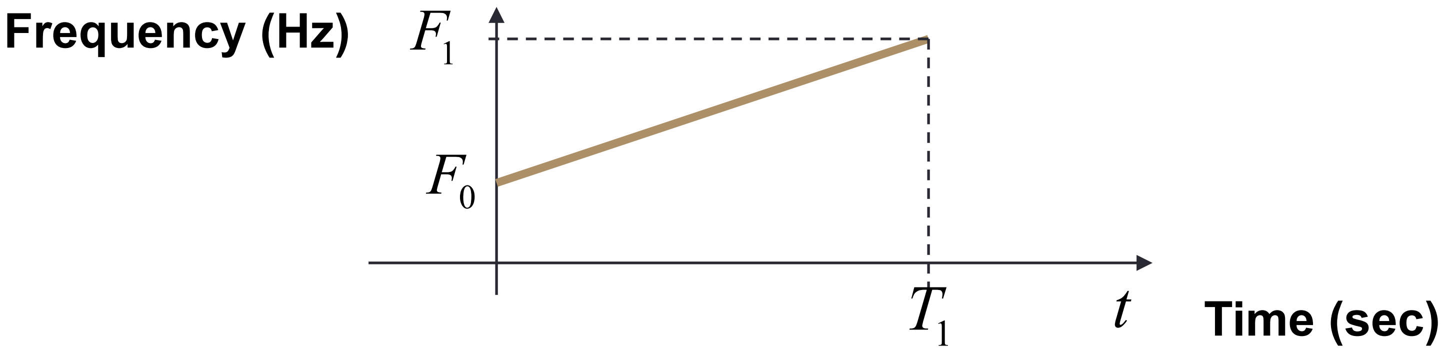


Chirp

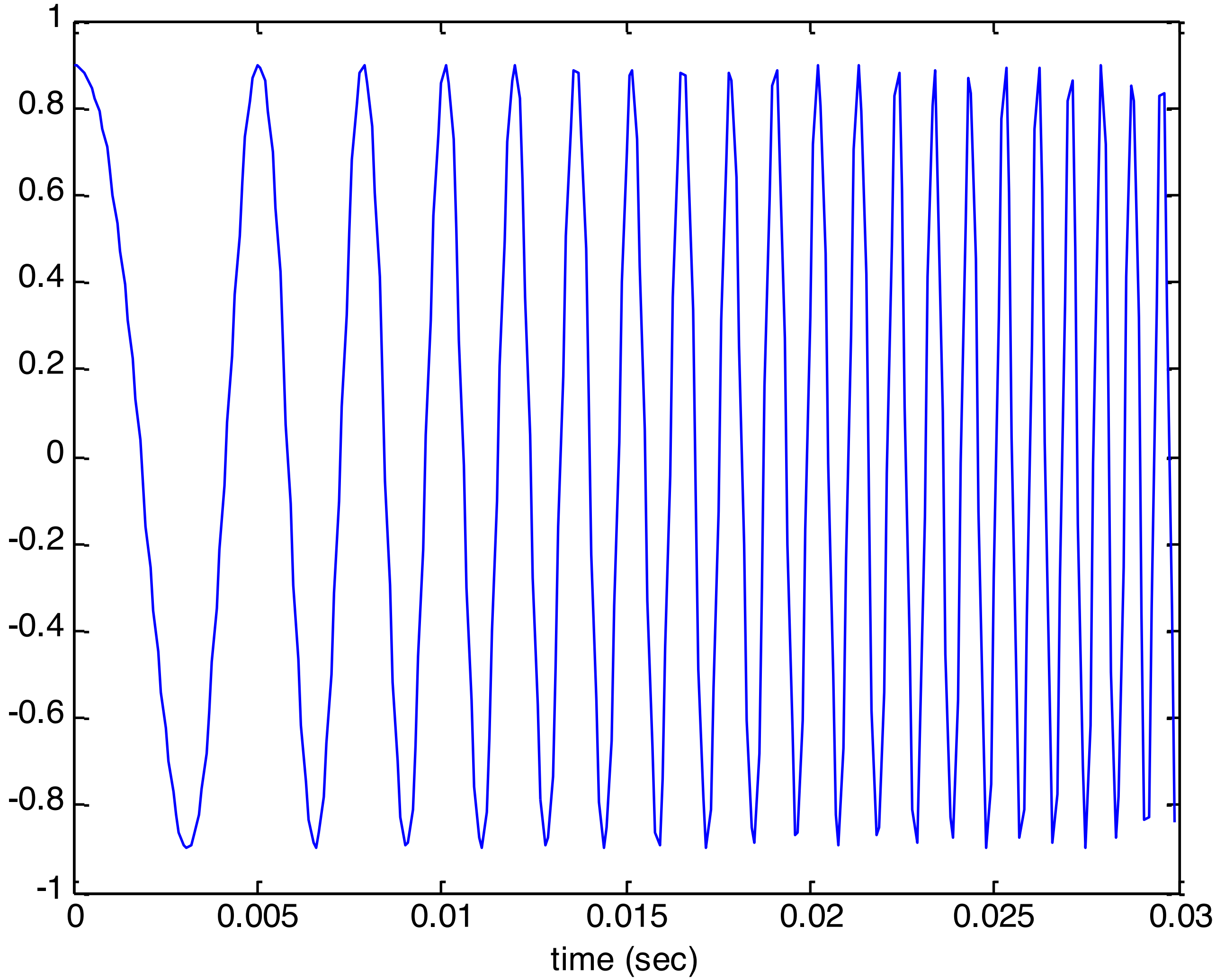
A "Chirp" is a sinusoid with time varying frequency, with expression

$$x(t) = A\cos\left(2\pi F(t)t + \alpha\right)$$

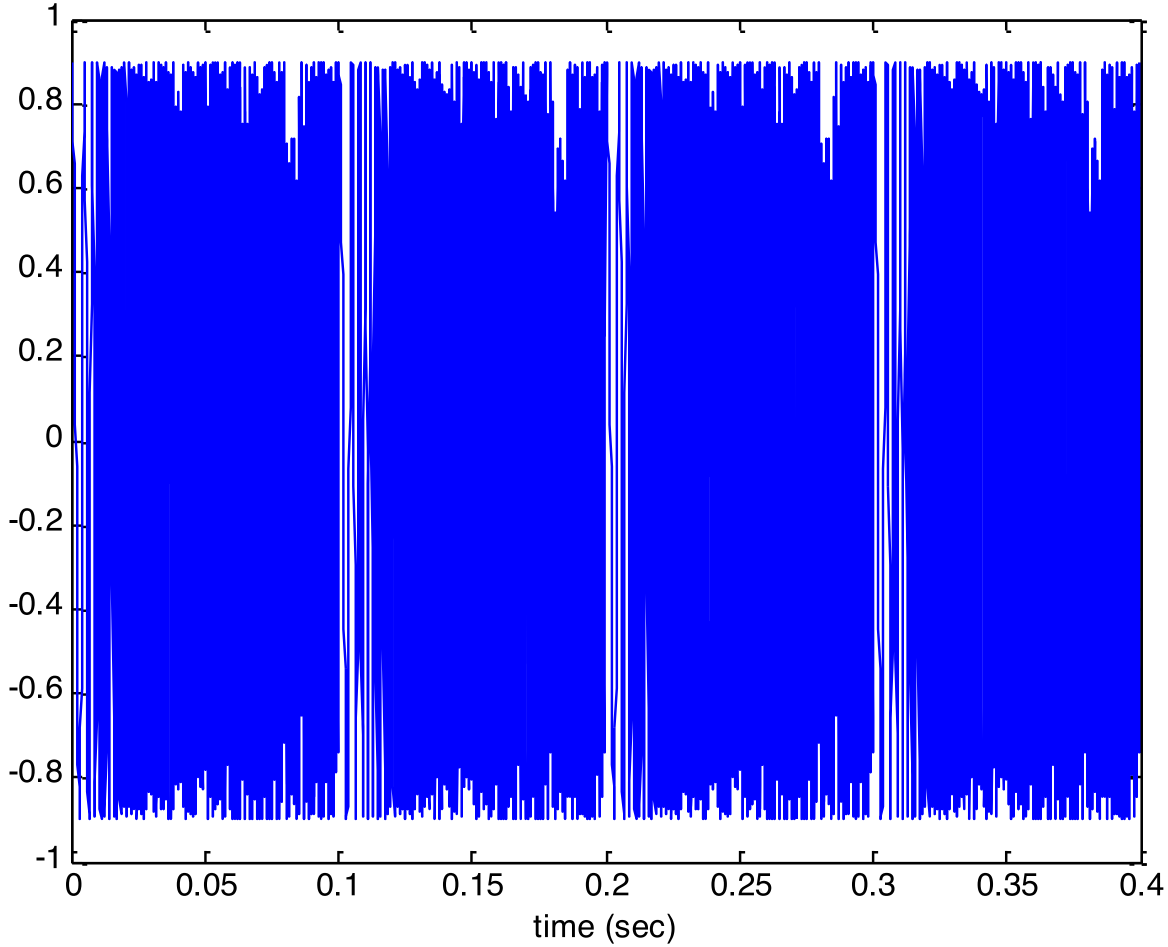
If the frequency changes linearly with time, it has the form shown below:



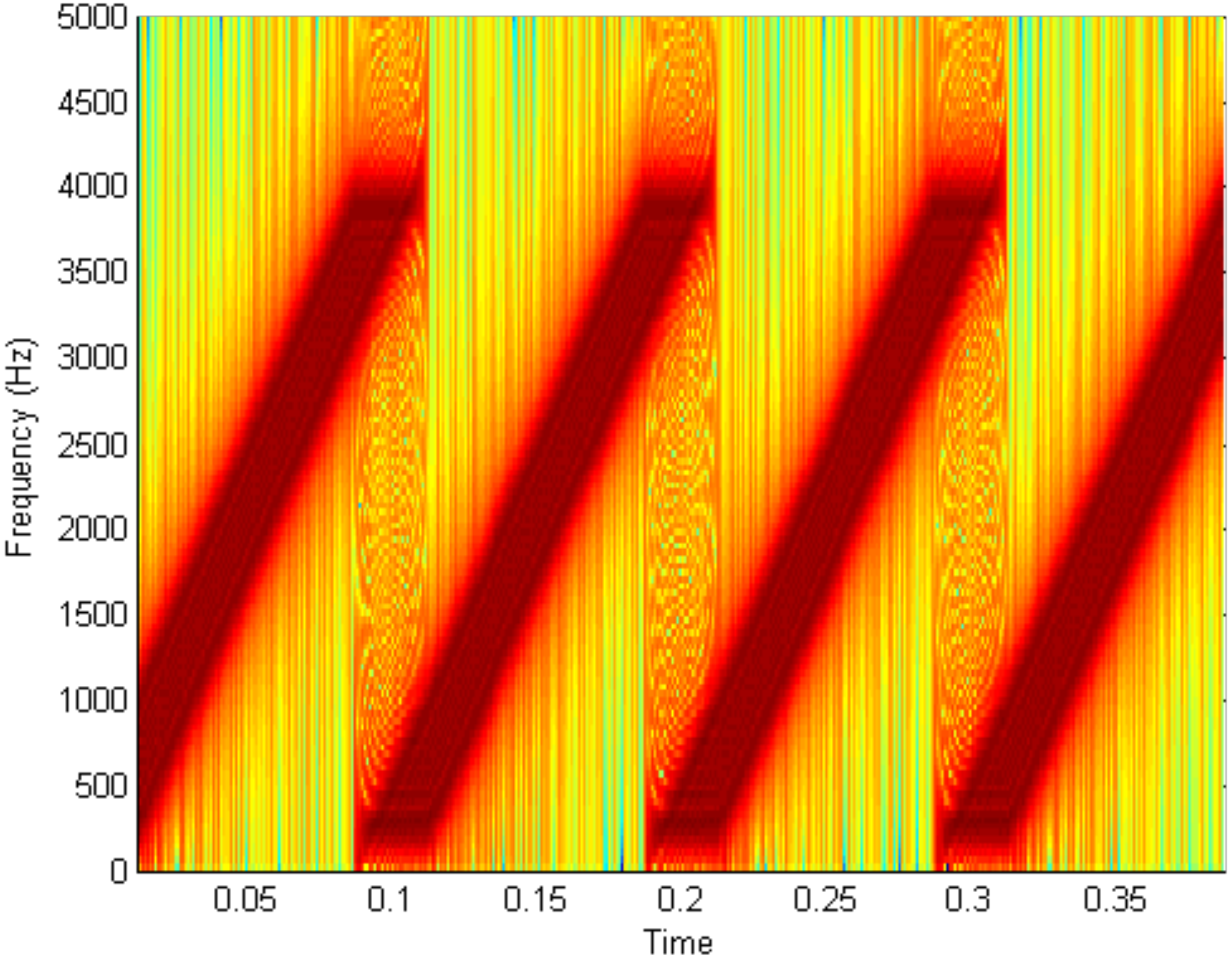
Exemplo



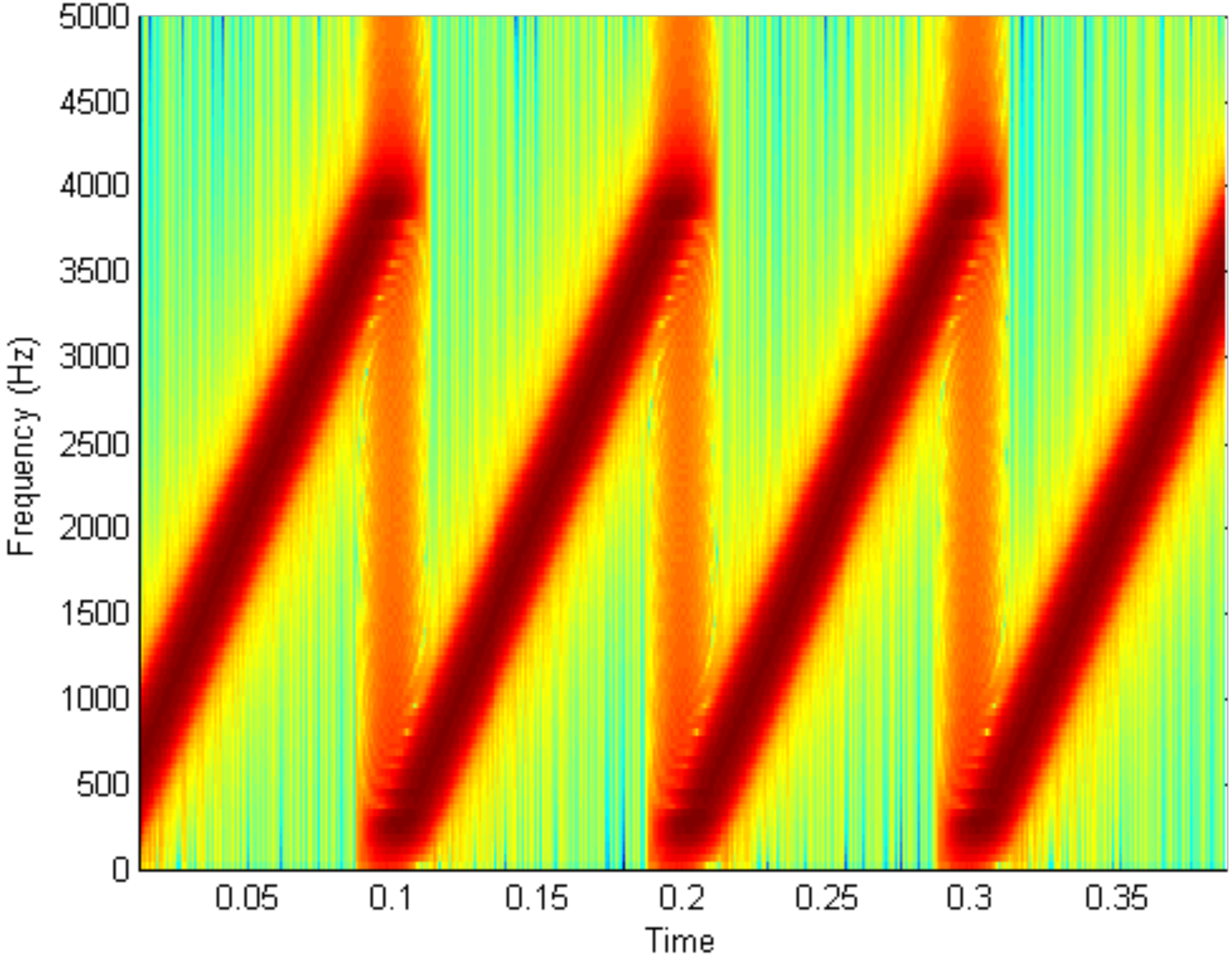
Exemplo: 4 repetições



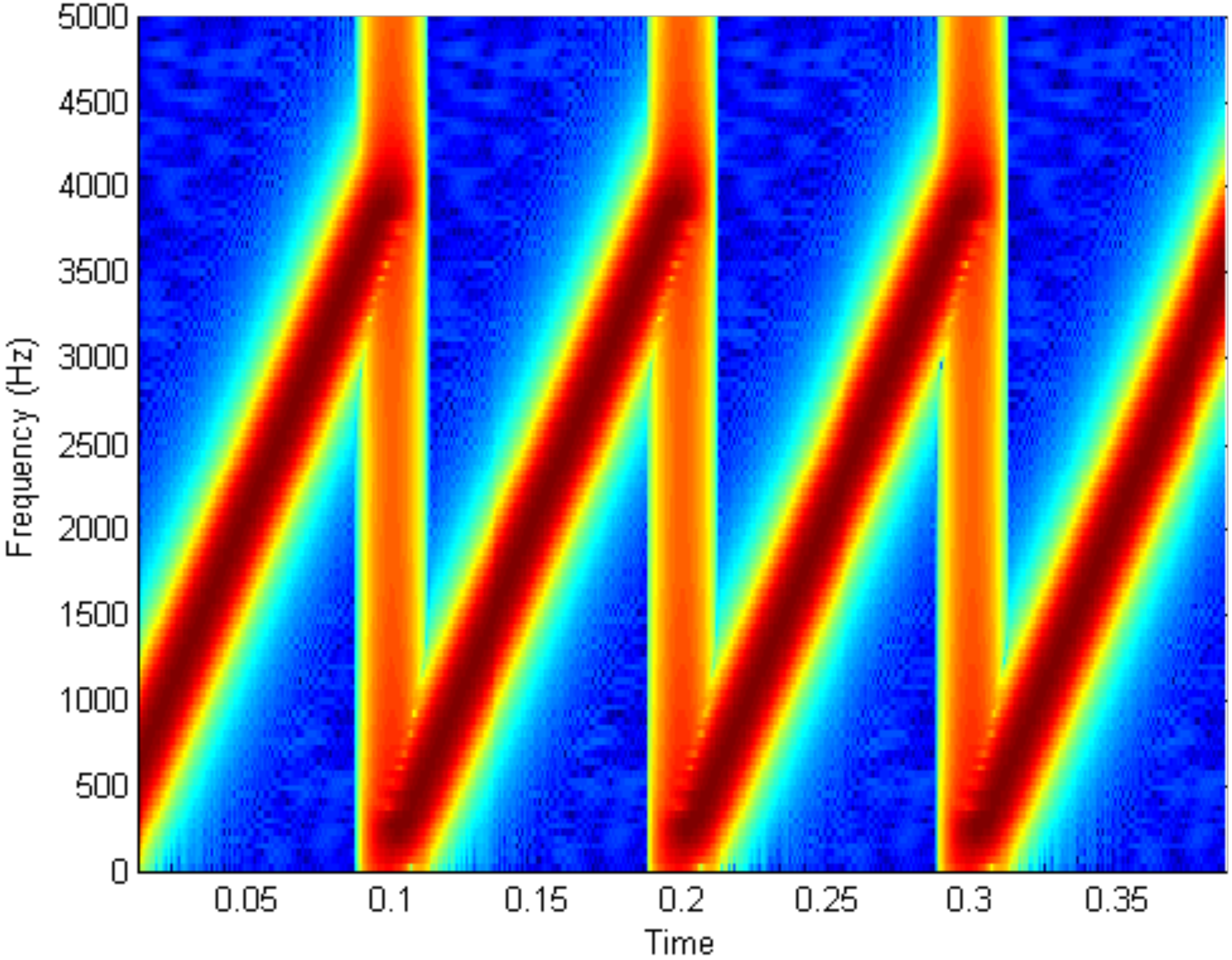
Espectograma



Espectograma com filtro *Hamming*

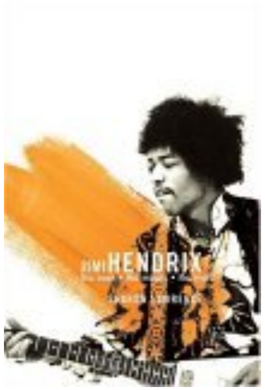


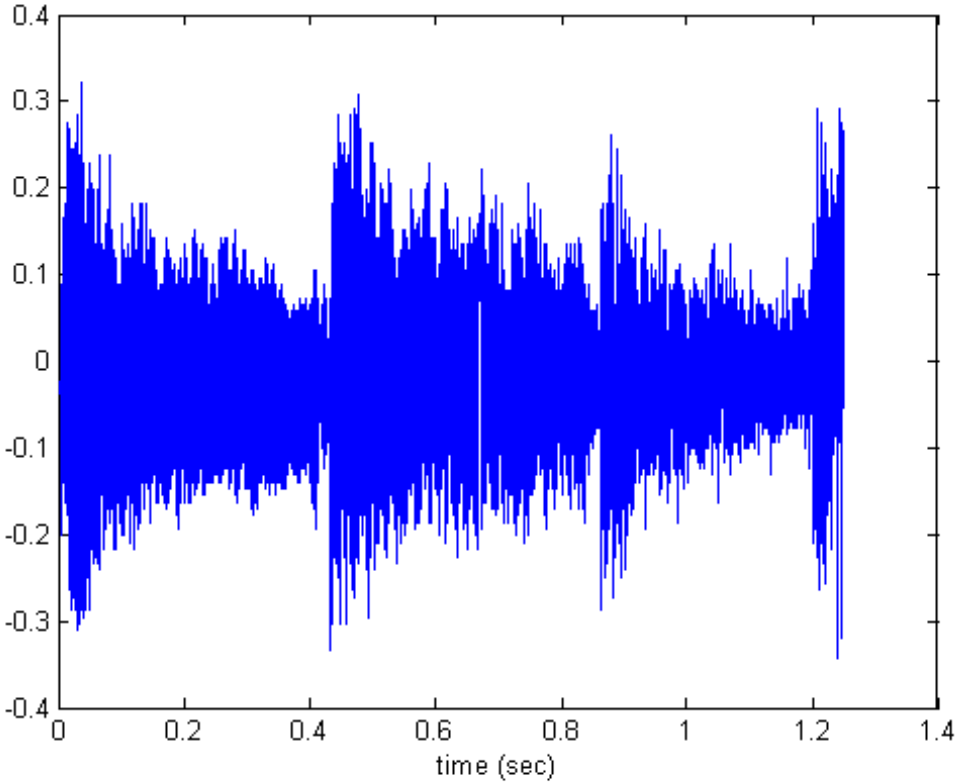
Espectograma com filtro *Blackman*



Sinal de áudio

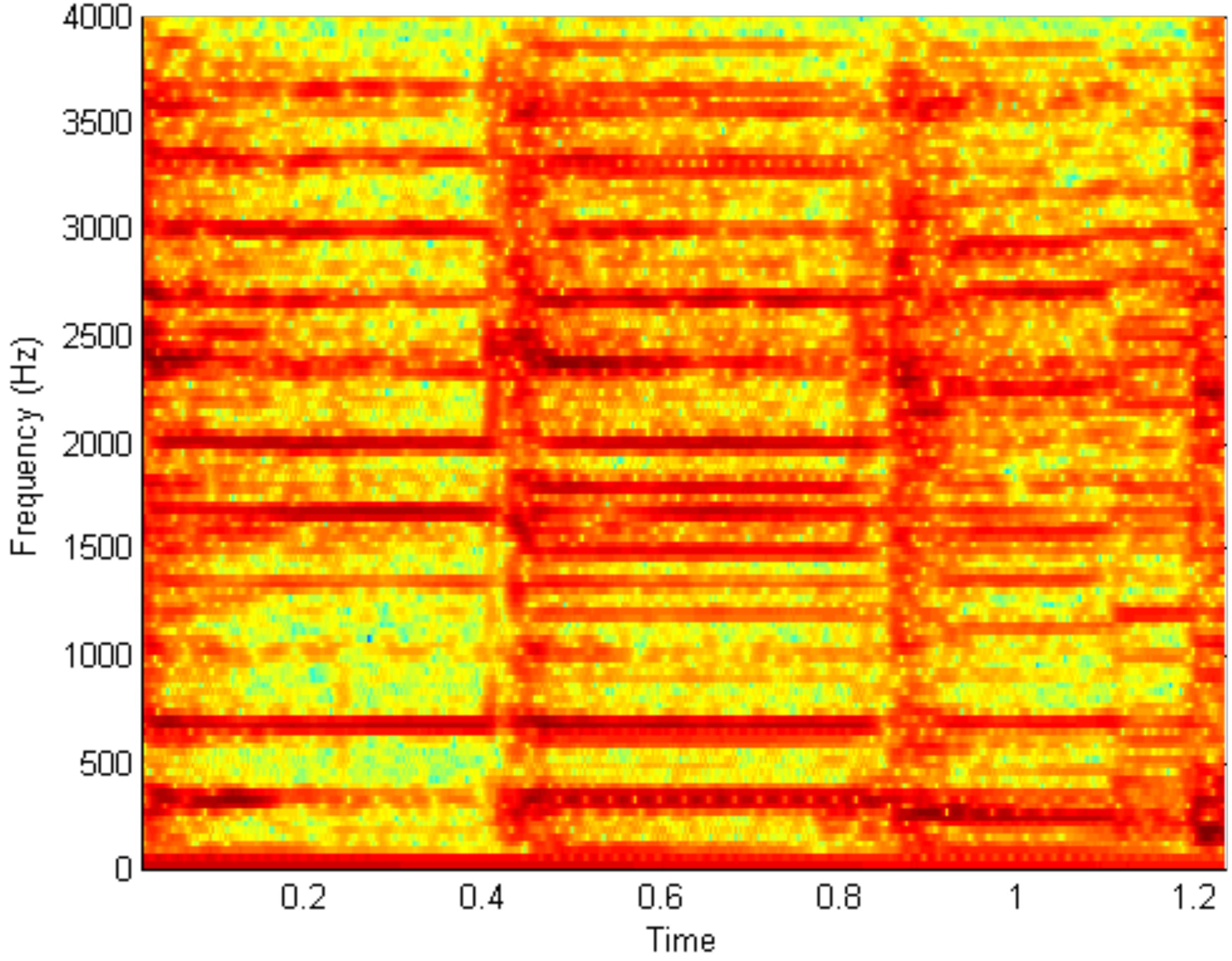
spectrogram(y(2001:12000), blackman(512), 500, 512,Fs,'yaxis');





Espectograma

spectrogram(y(12001:22000), hamming(256), 250, 256,'yaxis');



Separação de frequências

Since this signal contains music we expect to distinguish between musical notes. These are the frequencies associated to it (rounded to closest integer):

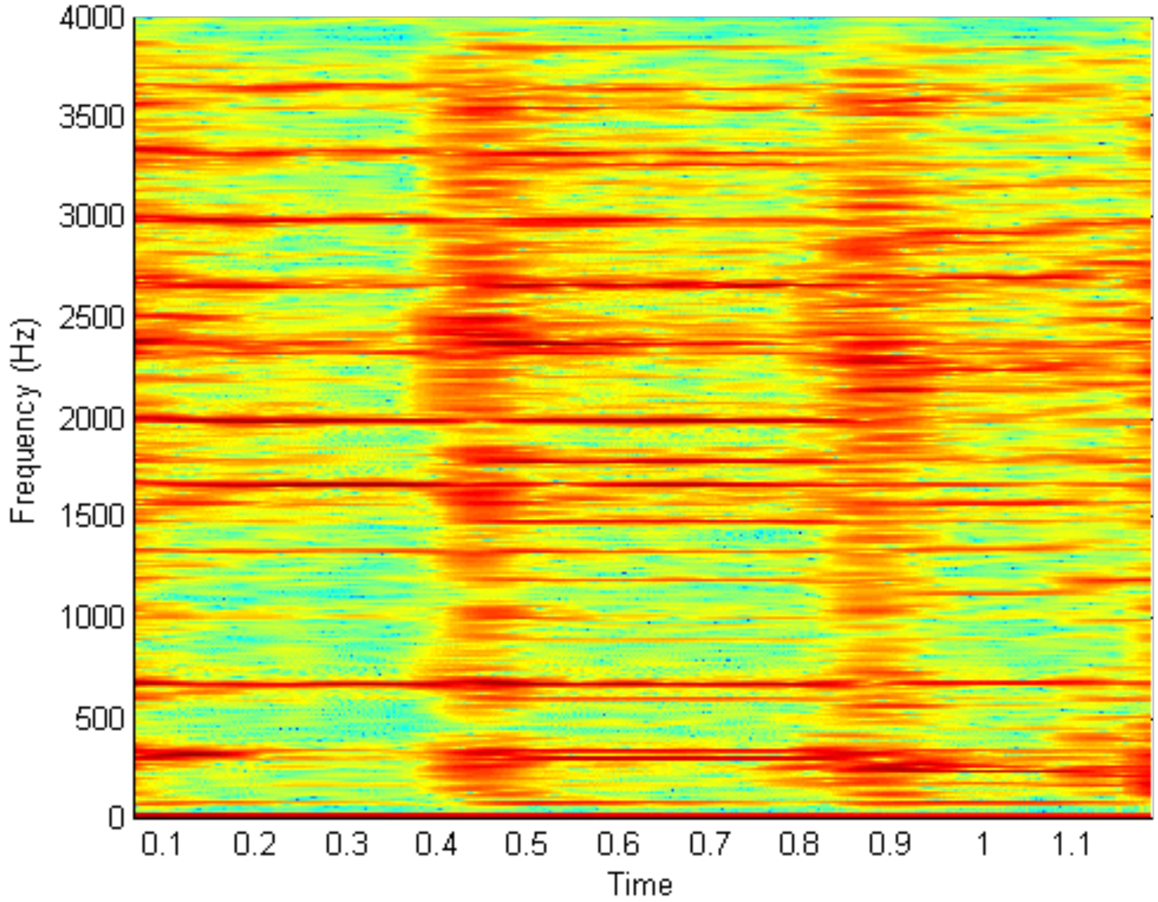
Notes	C	Db	D	Eb	E	F	Gb	G	Ab	A	Bb	B
Freq. (Hz)	262	277	294	311	330	349	370	392	415	440	466	494

Desired Frequency Resolution*:** $\Delta F \approx 2\frac{F_S}{N} \leq 15Hz$

This yields a window length of at least *N*=1024

Espectograma (janela estendida)

spectrogram(y(12001:22000), hamming(1024), 1000, 1024,'yaxis');



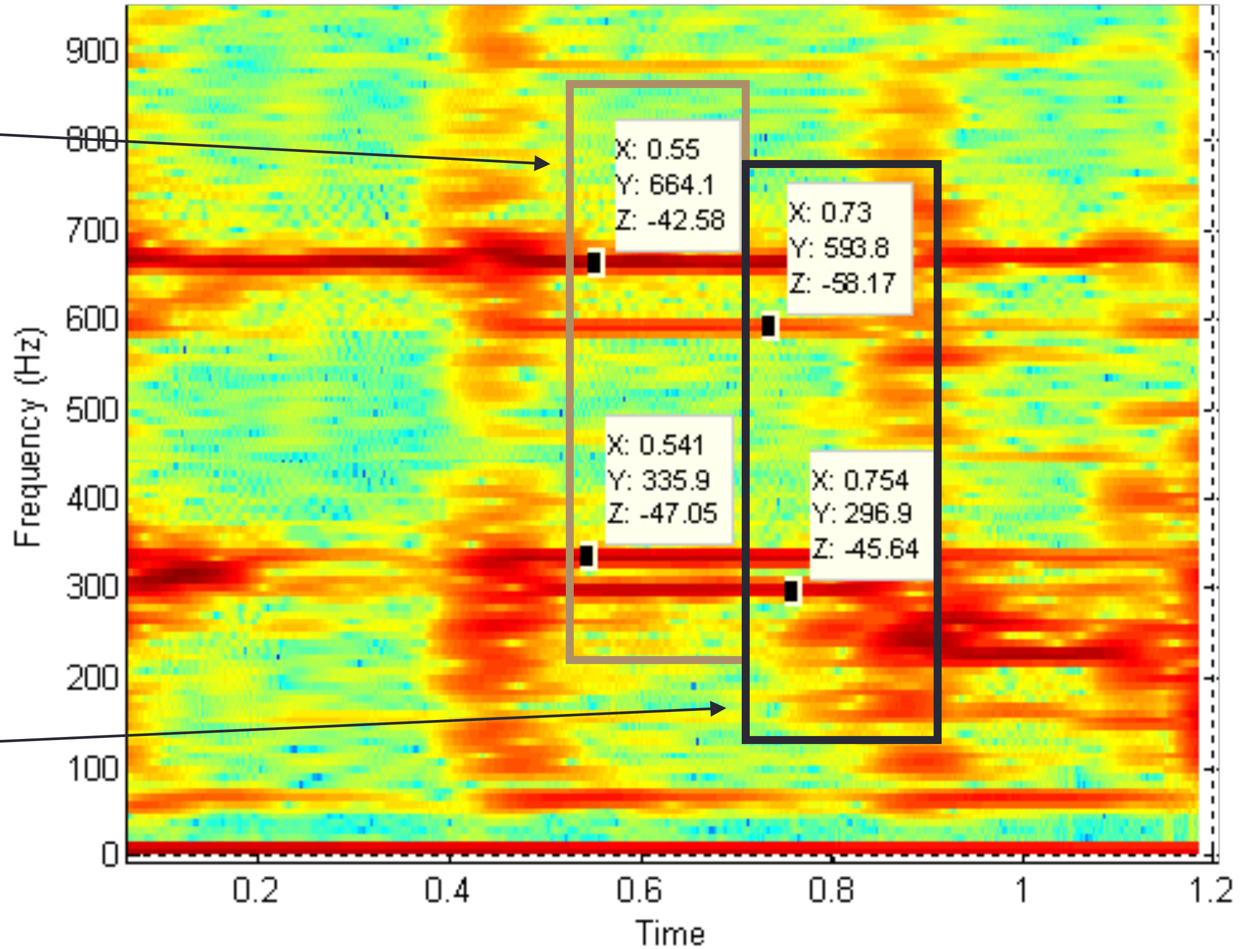
Espectograma

spectrogram(y(12001:22000), hamming(1024), 1000, 1024, 'yaxis');

Closest Notes:

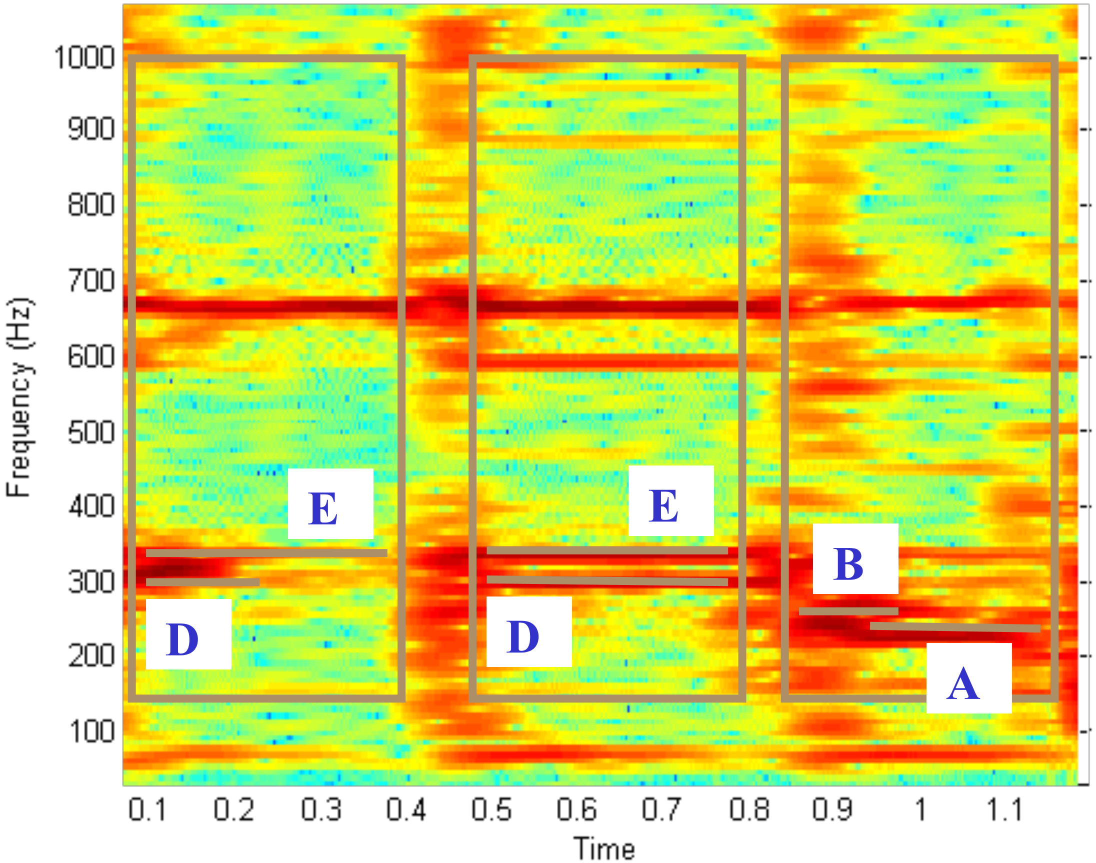
E: 330Hz, 660Hz

D: 294Hz, 588Hz



Validação


Moderately Slow ♩ = 88
1
Gtr. 1
*N.C.
mf
E
N.C.
E5


Frequency (Hz)
Time
E
D
E
D
B
A

Tarefa de casa

- Leitura do livro-texto capítulo 5 e exercícios