Trabajo Práctico n°3

ANÁLISIS FRECUENCIAL DE SEÑALES

Objetivos del práctico

- **Estudiar el uso de la Transformada de Fourier en Tiempo Discreto (*DTFT*) para el análisis espectral de señales.**

- **Implementación en Matlab de algoritmos de cálculo de la DTFT.**

- **Estudio de la performance de los diferentes tipos de ventanas.**

Estudiar el uso de la DTFT

Se puede calcular en forma exacta si la señal:

- es de longitud finita.
- es de longitud infinita, pero posee una transformada Z racional.

Definición de DTFT

Para una señal en tiempo discreto x(n), la DTFT está definida por:

$$X(\omega) = \sum_{n=-\infty}^{\infty} x(n) . e^{-j\omega n} \quad ; \quad -\pi \leq \omega \leq \pi$$

X(ω) es una función a valores complejos y periódica, con período fundamental 2π.

Problemas en el cálculo de DTFT

- No es posible usar la expresión anterior para calcular la DTFT para señales de longitud infinita, ya que **Matlab** sólo puede manipular vectores de longitud finita.

Excepción: Caso en que pueda obtenerse una expresión analítica de la DTFT, que pueda usarse para el cómputo

Problemas en el cálculo de DTFT

- La DTFT es una función de variable continua en el intervalo $[-\pi,\pi)$. Para el cálculo con **Matlab**, esto representa un problema de *muestreo en frecuencia.*

Realiza el cálculo en un número finito de frecuencias para obtener una aproximación suave de la DTFT

Generalmente se toman frecuencias equiespaciadas en el intervalo [-π,π) , es decir:

$$\omega_k = \frac{2\pi k}{N} \quad ; \quad k = 0,\ldots,N-1$$

con lo que resulta:

$$X(\omega_k) = \sum_{n=-\infty}^{\infty} x(n).e^{-j\omega_k n} =$$

$$= \sum_{n=-\infty}^{\infty} x(n).e^{-j(2\pi k/N)n} \quad ; \quad k = 0,\ldots,N-1$$

Esta expresión todavía no es computable, salvo para señales de longitud finita. En este caso, si la señal es de longitud L , tenemos:

$$X(\omega_k) = \sum_{n=0}^{L-1} x(n).e^{-j\omega_k n}$$

$$= \sum_{n=0}^{L-1} x(n).e^{-j(2\pi k/N)n} \quad ; \quad k = 0, \ldots, N-1$$

Evaluando la expresión anterior en ω_k para k =0, ... , N-1 tenemos:

$$
\begin{aligned}
X(\omega_0) &= x(0)e^{-j\omega_0 0} + x(1)e^{-j\omega_0 1} + \dots + x(L-1)e^{-j\omega_0 (L-1)} \\
X(\omega_1) &= x(0)e^{-j\omega_1 0} + x(1)e^{-j\omega_1 1} + \dots + x(L-1)e^{-j\omega_1 (L-1)} \\
X(\omega_2) &= x(0)e^{-j\omega_2 0} + x(1)e^{-j\omega_2 1} + \dots + x(L-1)e^{-j\omega_2 (L-1)} \\
&\vdots \qquad\qquad \vdots \qquad\qquad\qquad \vdots \qquad\qquad\qquad \vdots \\
X(\omega_{N-1}) &= x(0)e^{-j\omega_{N-1} 0} + x(1)e^{-j\omega_{N-1} 1} + \dots + x(L-1)e^{-j\omega_{N-1} (L-1)}
\end{aligned}
$$

Y escribiéndola en forma matricial, resulta:

$$\begin{bmatrix} X(\omega_0) \\ X(\omega_1) \\ \vdots \\ X(\omega_{N-1}) \end{bmatrix} = \begin{bmatrix} e^{-j\omega_0 0} & e^{-j\omega_0 1} & \dots & e^{-j\omega_0 (L-1)} \\ e^{-j\omega_1 0} & e^{-j\omega_1 1} & \dots & e^{-j\omega_1 (L-1)} \\ \vdots & \vdots & & \vdots \\ e^{-j\omega_{N-1} 0} & e^{-j\omega_{N-1} 1} & \dots & e^{-j\omega_{N-1} (L-1)} \end{bmatrix} \cdot \begin{bmatrix} x(0) \\ x(1) \\ \vdots \\ x(L-1) \end{bmatrix}$$

Desarrollo del Práctico

***Problema 1*:** **Implementación de una función que calcule la DTFT para una señal finita en el tiempo.**

***Problema 2*:** **Implementación de una función que calcule la DTFT para una señal con Transformada-Z del tipo racional.**

***Problema 3*:** **Comparación de las diferentes funciones “ventanas” utilizadas normalmente para el truncamiento de señales de duración infinita.**

***Problema 4*:** **Uso de la FFT de Matlab.**

Problema 2

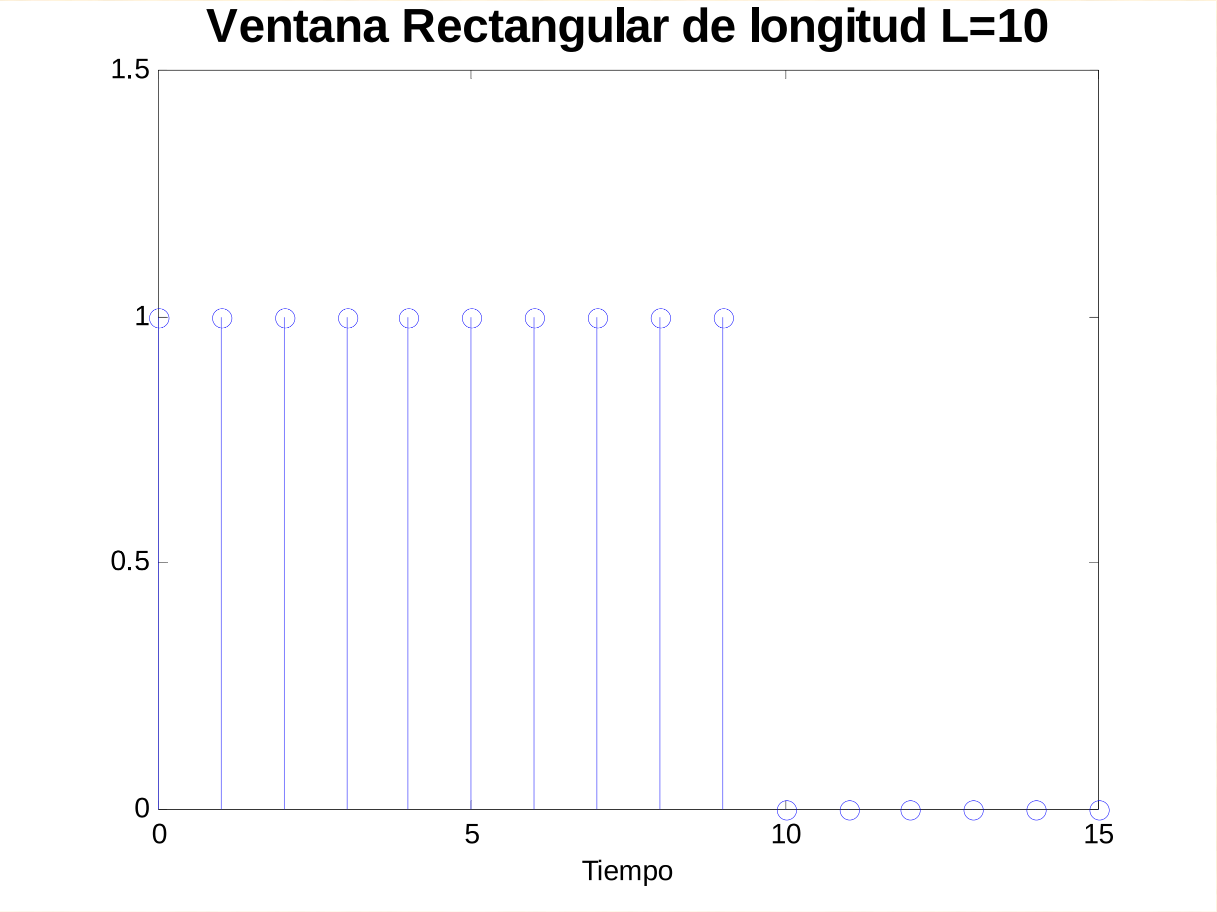
Sea *x(n)* una señal de longitud infinita con Transformada *Z* racional dada por:

$$X(z) = \sum_{n=-\infty}^{\infty} x(n) . z^{-n}$$

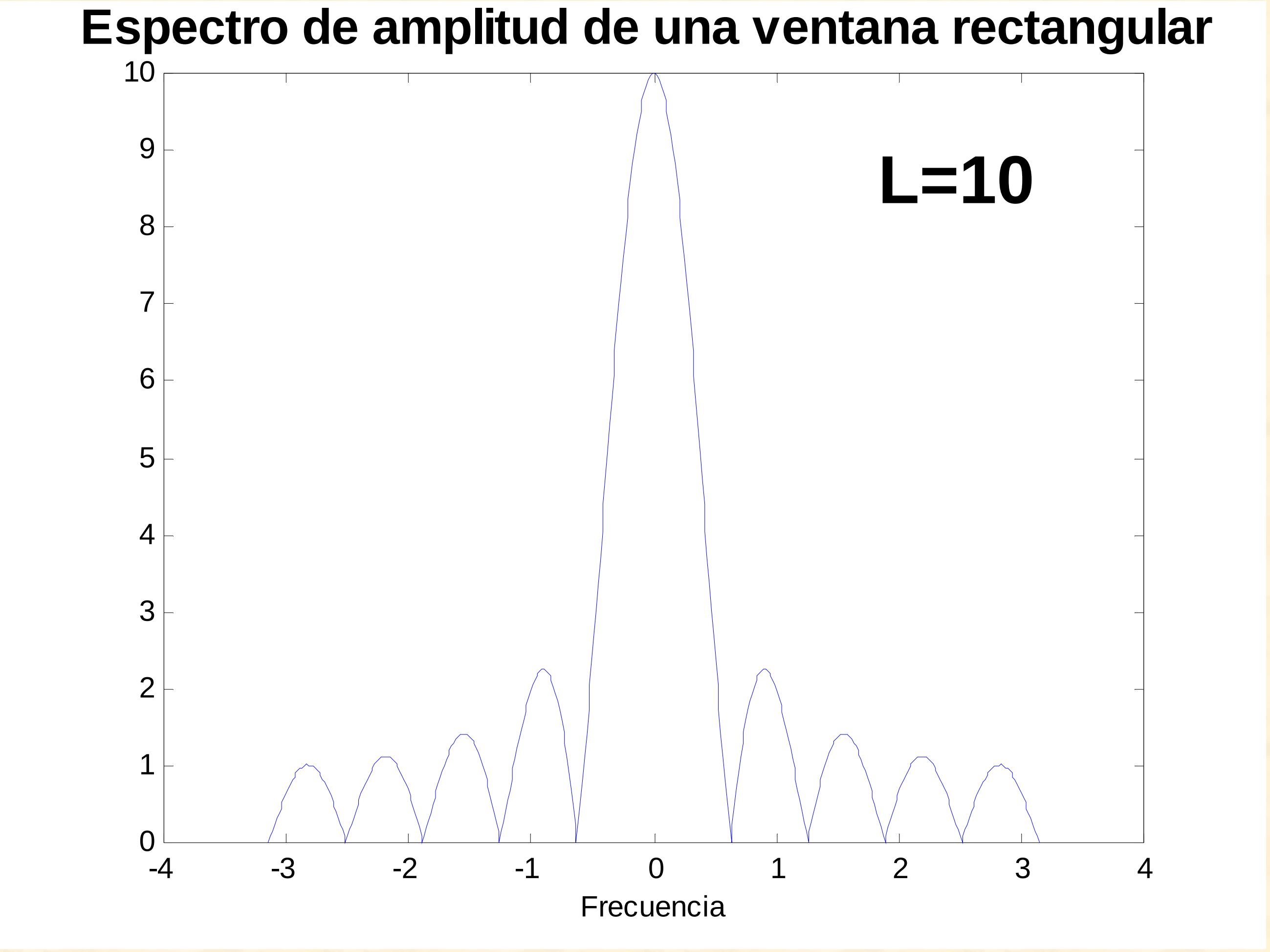
Si la RDC incluye a la circunferencia unitaria, entonces la DTFT de la señal puede calcularse evaluando su *X(z)* en $z = e^{j\omega}$

Problema 3

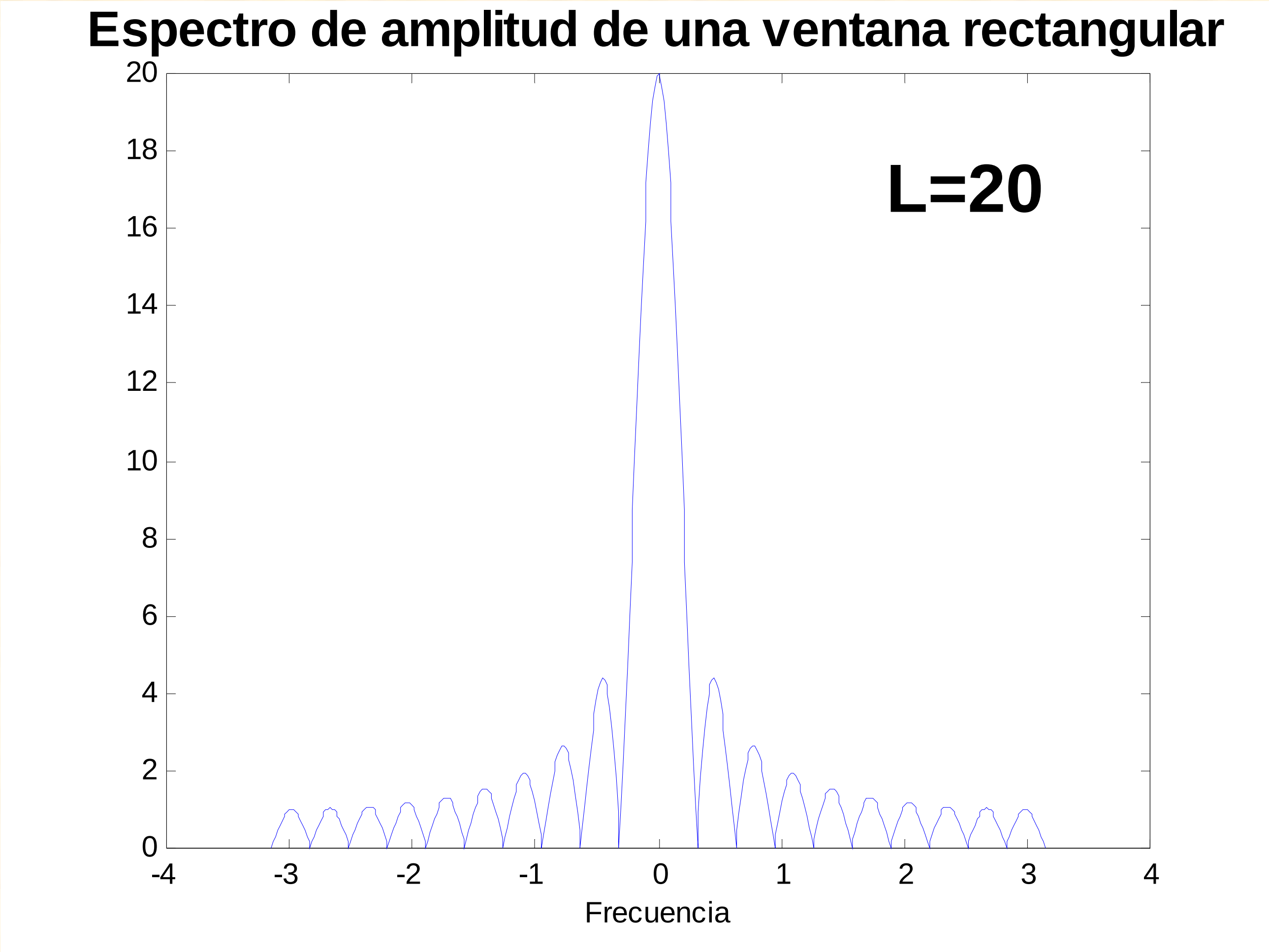
- Ventana Rectangular (`boxcar`)



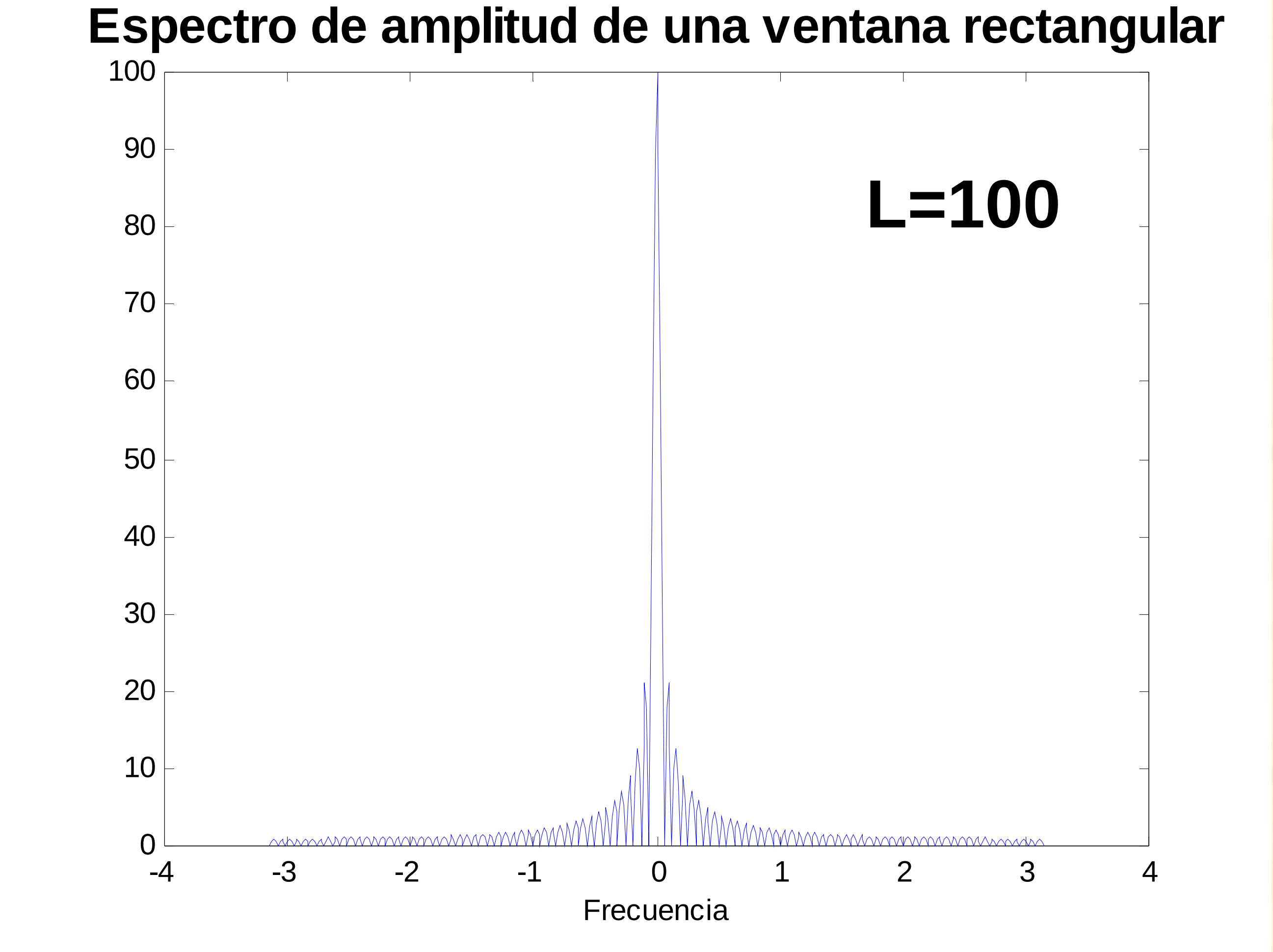
Implementando la función del Problema 1, podemos obtener los espectros correspondientes:



Tomando una ventana del doble de longitud de la anterior, tenemos:

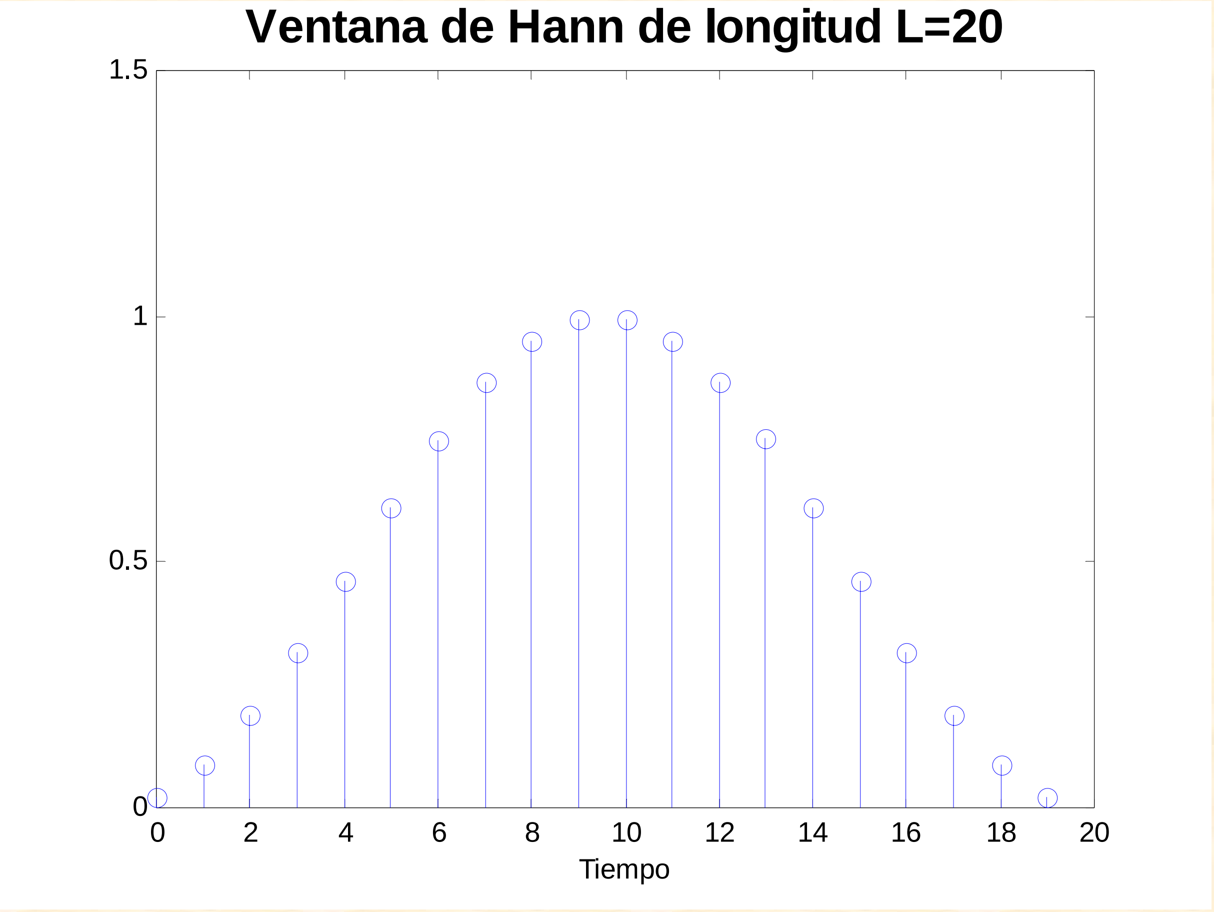


Finalmente, para una ventana de longitud L=100, resulta:



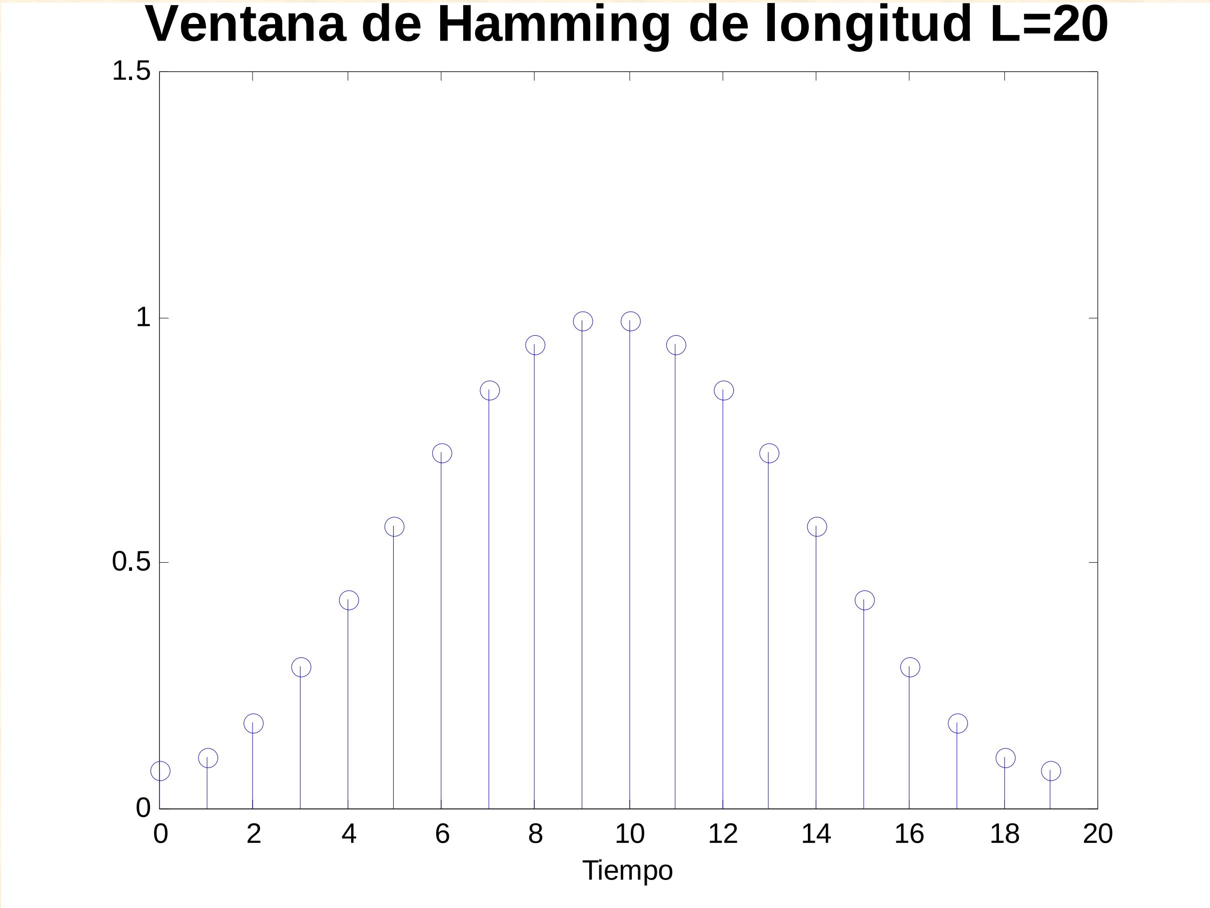
Problema 3

- Ventana de Hann (`hanning`)



Problema 3

- **Ventana de Hamming (`hamming`)**



Problema 3

- Considere la señal de longitud infinita:

$$x(n) = \cos\left(\frac{\pi}{2} n\right)$$

Calcule el espectro de dicha señal usando distintas ventanas

$$\hat{x}(n) = x(n).w(n)$$

que en el dominio transformado queda:

$$\hat{X}(\omega) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\lambda).W(\omega - \lambda)\, d\lambda$$

Definición de DFT

Para una señal en tiempo discreto x(n), de longitud finita L, su DFT con N puntos está definida por:

$$X(k) = \sum_{n=0}^{N-1} x(n) \, . \, e^{-j(2\pi k/N)n} \quad ; \quad k = 0, \ldots, N-1$$

Implementación Matlab de la FFT

- **Matlab** implementa la FFT en la función **fft**.

- **Dada una señal guardada en el vector x, el sig. Comando:**

  >> y = fft(x,N);

  Calcula la DFT con N=length(x) puntos de la secuencia x. El número de puntos de la DFT puede especificarse independientemente de la longitud de la secuencia mediante el argumento de entrada adicional de la función N.

Problema 4

El archivo de datos **tp3_1.mat** contiene un vector x que consiste de las muestras de una señal inmersa en ruido, muestreada con una frecuencia de F_s=1 KHz. Se asume que el ruido es blanco Gaussiano

Usar la función fft de Matlab para calcular y graficar la DTFT de la señal x , usando una ventana de longitud apropiada. Estudiar el uso de diferentes ventanas y de longitudes de ventana.