Trabajo práctico #1.
Para realizar en la sesión práctica del 24/03/04

Representación de señales continuas y discretas.
Desde la ventana de comandos de MATLAB, usa el comando *help* para averiguar sobre el uso de las siguientes funciones: *sin, cos, square, sawtooth, sinc, exp, subplot, plot, stem, title, xlabel, ylabel, clear*.

Parte A: Representación de señales continuas.
La representación de señales en MATLAB es por definición siempre discreta. Sin embargo es posible representar señales continuas mediante la interpolación de gráficos discretos. En este primer ejercicio deberás representar la señal continua $y(t) = cos(2\pi f\, t)$
1.- genera un vector de tiempo *t* de *0* a *2* segundos en intervalos de *50 ms*
2.- evalúa la función *y(t)* para una frecuencia de $f = 1Hz$
3.- grafica *y(t)* vs *t* usando el comando *plot(t,y)*
4.- usa los comandos *title, xlabel* y *ylabel* para identificar el grafico generado

Parte B: Representación de señales discretas.
En este segundo ejercicio deberás representar la señal discreta en tiempo $y[n] = cos(2\pi f\, n\, dt)$. Observa que esta expresión es exactamente igual a la de *y(t)* si hacemos $t = n\, dt$, donde *n* se denomina el número de muestra y *dt* el período de muestreo (en el trabajo práctico número 6 se abordará con más detalle el problema de muestreo de señales continuas).
En el dominio discreto, el número de muestra *n* constituye la variable independiente de la secuencia *y[n]* y sólo toma valores enteros, es decir la secuencia *y[n]* sólo está definida para valores enteros de *n*. (Cuidado con pensar cosas como $y[½] = 0$, esto es un grave error, *y[½]* simplemente no está definido).
1.- genera un vector de muestras *n* de *0* a *40* muestras
2.- evalúa la secuencia *y[n]* para una frecuencia de $f = 1Hz$ y un período de muestreo de $dt = 50\ ms$
3.- grafica *y[n]* vs *n* usando el comando *stem(n,y,'.')*
4.- usa los comandos *title, xlabel* y *ylabel* para identificar el grafico generado

Parte C: Frecuencia en el dominio de tiempo discreto
Como ya viste en la Parte B, en el dominio discreto, la variable independiente de una secuencia es el número de muestra *n*. De forma que la frecuencia en el dominio de tiempo discreto para la señal *y[n]* está dada por el producto $fd = f\, dt$. A diferencia de el dominio continuo, en el cual la frecuencia puede incrementarse indefinidamente, en el dominio discreto existen frecuencias máximas y mínimas. En este ejercicio se pretenden ilustrar los conceptos de frecuencias discretas máxima y mínima.
Genera nueve gráficos en una misma figura de la secuencia discreta $y[n] = cos(2\pi\, fd\, n)$, usando los valores de frecuencia discreta $fd = 0, 1/8, 2/8, 3/8, 4/8, 5/8, 6/8, 7/8$ y $8/8$.
¿Cuál es el valor de frecuencia máxima?, ¿cuál es el valor de frecuencia mínima?, ¿puedes encontrar expresiones cerradas para todos los valores de frecuencias máximas y frecuencias mínimas?

Parte D: Otras señales discretas de interés
Genera cuatro gráficos en la misma figura con las siguientes señales discretas en tiempo: onda cuadrada, diente de sierra, exponencial de dos lados (*exp(-abs(n))*) y sinc ($sin(\pi n)/\pi n$).

Trabajo práctico #2.
Para realizar en la sesión práctica del 31/03/04

Convolución discreta y ecuaciones en diferencias finitas.
Desde la ventana de comandos de MATLAB, usa el comando *help* para averiguar sobre el uso de las siguientes funciones: *zeros, ones, conv, diff, filter, pause, figure*.

Parte A: Convolución de dos señales discretas.
1.- Considera las secuencias $f1[n] = \delta[n-1] + u[n] - u[n-4]$ y $f2[n] = \delta[n] - \delta[n-1] + \delta[n-4] + \delta[n-5]$. Calcula a mano los coeficientes de la secuencia convolución $fc[n] = f1[n] * f2[n]$. Genera cuatro gráficos sobre una misma figura y presenta en ellos: la secuencia $f1[n]$, la secuencia $f2[n]$, la convolución $fc[n]$ calculada por ti, y la convolución calculada usando la función *conv* de MATLAB.

2.- Considera la siguiente respuesta impulsiva de un sistema LIT, $h[n] = u[n] – 3\, u[n-4] + 2\, u[n-6]$. Calcula a mano la respuesta de dicho sistema a una entrada del tipo escalón unitario $u[n]$. Calcula esa misma respuesta usando la función *conv* de MATLAB (crea una función escalón en MATLAB evaluada en el rango $n = -10:30$). Genera cuatro gráficos sobre una misma figura y presenta en ellos: la respuesta impulsiva $h[n]$, el escalón unitario $u[n]$, la respuesta al escalón calculada por ti, y la respuesta al escalón calculada usando la función *conv*. ¿Qué diferencia observas entre la respuesta al escalón calculada por ti y la calculada en MATLAB? ¿A qué se debe dicha diferencia? ¿Cómo puedes resolver este problema?

Parte B: Diferenciador discreto de primer orden.
Considera el sistema LIT cuya respuesta impulsiva esta definida por $h[n] = \delta[n] - \delta[n-1]$. Este sistema se denomina diferenciador discreto de primer orden. Analiza lo que sucede al hacer la convolución de la respuesta impulsiva de este sistema con cualquier señal discreta y justifica el por qué este sistema recibe dicho nombre. En este ejercicio aplicarás este sistema a un pulso triangular:
1.- genera el pulso triangular definido por $\Delta = 1/5\ (\ r[n+5] – 2\ r[n]\ )\ u[5-n]$
2.- calcula la respuesta del diferenciador al pulso triangular usando la función *conv*
3.- estudia el funcionamiento de la funcionamiento de la función *diff* y utilízala para generar el mismo resultado del paso anterior
4.- genera cuatro gráficos en la misma figura y presenta en ellos: el pulso triangular, la respuesta impulsiva del diferenciador discreto, la respuesta calculada usando la función *conv*, y la respuesta calculada usando la función *diff*.

Parte C: Representación de sistemas LIT con ecuaciones en diferencias
Considera la siguiente ecuación en diferencias: $y[n] = x[n] + 0{,}4\, x[n-1] – A\, y[n-1] – B\, y[n-2]$, según la cual se definen cuatro sistemas discretos: el sistema I con $A=1{,}6$ y $B=0{,}64$; el sistema II con $A= -1{,}6$ y $B=0{,}64$; el sistema III con $A=2{,}4$ y $B=1{,}44$; y el sistema IV con $A= -2{,}4$ y $B=1{,}44$.
1.- estudia la función *filter* y ve como puede ser usada para resolver ecuaciones en diferencias
2.- grafica en una misma figura las respuestas impulsivas de los cuatro sistemas
3.- grafica en una misma figura las respuestas al escalón de los cuatro sistemas
4.- clasifica los sistemas según sean estables o inestables, y oscilatorios o no oscilatorios
5.- para aquellos sistemas que sean estables estima la ganancia y el tiempo de estabilización.

Trabajo práctico #3.
Para realizar en la sesión práctica del 14/04/04

La DTFT y el algoritmo de FFT.
Desde la ventana de comandos de MATLAB, usa el comando *help* para averiguar sobre el uso de las siguientes funciones: *linspace, sum, abs, angle, fft, max, min, for-end, axis, sprintf, hold on, hold off*.

Parte A: La transformada de Fourier en tiempo discreto DTFT.
En este ejercicio usaremos MATLAB para representar la trasnformada de Fourier de una señal discreta en tiempo ó DTFT (recuerda que la DTFT es en realidad una función continua y que MATLAB sólo representa señales discretas, así que lo que realmente harás es calcular los valores de la DTFT para algunos valores de la variable continua ω). Considera la expresión de la DTFT vista en clase:
1.- genera la secuencia $y[n] = \cos(7/32\ \pi n)$ para $n = 0{:}127$
2.- genera un vector de frecuencias ω de 1025 muestras uniformemente espaciadas entre 0 y 2π
3.- evalúa la DTFT de $y[n]$ en los valores de frecuencia contenidos en ω, tratando de usar la menor cantidad de veces el comando *for* (te reto a que evalúes la DTFT sin usar ningún *for*)
4.- genera tres gráficos en la misma figura y presenta en ellos: la secuencia $y[n]$, la magnitud de la DTFT que calculaste, y la fase de la DTFT que calculaste (recuerda nuevamente que la DTFT es una función continua, grafícala usando *plot*).

Parte B: El algoritmo de FFT (Fast Fourier Transform)
La FFT ó Transformada Rápida de Fourier es un algoritmo que implementa la DTFS ó serie de Fourier en tiempo discreto. Recuerda que cualquier representación que hagas de señales y/o sus transformadas en el computador es a fin de cuentas discreta en tiempo (digital para ser exactos). La gran ventaja del algoritmo FFT es su velocidad, y es especialmente rápido cuando el número de muestras de la señal a transformar es una potencia de dos (en el trabajo práctico #4 veremos con más detalle este aspecto). Revisa, usando el comando *help*, la ayuda que ofrece MATLAB sobre la función *fft*.
1.- calcula la DTFS de 128 puntos para la secuencia $y[n]$ usando el comando *fft*
2.- genera tres gráficos en la misma figura y presenta en ellos: la secuencia $y[n]$, la magnitud de la DTFS que calculaste, y la fase de la DTFS que calculaste (la DTFS es una función discreta, grafícala usando *stem*)
3.- explica, justificando tus razonamientos, las diferencias encontradas entre la DTFT calculada en la Parte A y DTFS que acabas de calcular.

Parte C: Relación entre la DTFT y la DTFS
En este ejercicio te vas a enfocar en la región del espectro comprendida entre 0 y $\pi/2$. Para tales efectos vas a tener que recalcular la DTFT de la Parte A, pero en esta oportunidad la evaluarás en 513 valores de frecuencia uniformemente espaciadas entre 0 y $\pi/2$.
1.- calcula la DTFS de 128 puntos para la secuencia $y[n]$ usando el comando *fft*
2.- genera dos gráficos en la misma figura y presenta en el primer gráfico las magnitudes de la DTFT y la DTFS, y en el segundo gráfico las fases (nota importante: usa *plot* para la DTFT y *stem* para la DTFS, para poder hacer esto tendrás que necesariamente usar los comandos *hold on* y *hold off*)
3.- repite los pasos 1 y 2 con DTFS de 256 y de 512 puntos.
4.- ¿Qué de lo visto en las sesiones de teoría puedes confirmar con estos gráficos?
5.- ¿Notaste algo extraño en los gráficos de las fases? Describe lo observado y trata de darle una buena explicación.

Trabajo práctico #4.
Para realizar en la sesión práctica del 21/04/04

Convolución en el dominio de la frecuencia.
Desde la ventana de comandos de MATLAB, usa el comando *help* para averiguar sobre el uso de las siguientes funciones: *length, size, real, imag, ifft, rand, cputime, legend.*

Parte A: Convolución en los dominios del tiempo y la frecuencia.
En este ejercicio usaremos MATLAB para realizar la convolución de dos señales discretas usando dos métodos diferentes: la convolución en el dominio del tiempo y la convolución vía FFT. Esta última es una implementación que aprovecha la velocidad del algoritmo FFT y la propiedad de convolución de la transformada de Fourier (aquella que dice que convolucionar dos señales en el dominio del tiempo es equivalente a multiplicar sus tranformadas en el dominio de la frecuencia).
1.- genera un índice de tiempo discreto *n* de *100* muestras
2.- genera una señal *x[n]* que consista en la suma de tres sinusoides de igual amplitud; la primera sinusoide debe tener *5* ciclos exactos sobre el índice de tiempo *n*, la segunda debe tener *10* ciclos exactos y la tercera *25* ciclos exactos
3.- genera el filtro digital *f[n]* de la siguiente forma: *f = sin(2*pi*n/10).*exp(-abs(n-50)/20)*
4.- calcula *ytime[n]*, la convolución entre *x[n]* y *f[n]* usando la función *timeconv*; esta función no existe en MATLAB y debes crearla en un archivo aparte “timeconv.m” usando el siguiente código:

```
function result=timeconv(f1,f2)
n1=length(f1); n2=length(f2);
newf2=[zeros(1,n1-1),fliplr(f2),zeros(1,n1)];
for m=1:(n1+n2-1), result(m)=sum(f1.*newf2(m:(m+n1-1))); end
result = fliplr(result);
```

(esta función implementa en el dominio del tiempo de una forma sencilla, aunque poco eficiente desde el punto de vista del manejo de memoria, la convolución de dos señales discretas)
5.- calcula *yfreq[n]*, la convolución entre *x[n]* y *f[n]* vía FFT usando las funciones *fft* e *ifft*, recuerda lo visto en clase sobre la necesidad de añadir ceros al final de las secuencias para este tipo de operación
6.- genera dos gráficos sobre una misma figura y presenta: la secuencia *x[n]* y el filtro digital *f[n]*
7.- genera dos gráficos sobre una misma figura y presenta en el primero *yfreq[n]* y *ytime[n]*, y en el segundo la diferencia *yfreq[n] - ytime[n]*
8.- genera cuatro gráficos sobre una misma figura y presenta en ellos las magnitudes de los espectros de las señales *x[n]*, *f[n]*, *yfreq[n]* y *ytime[n]*.
9.- Observa todos los gráficos generados, ¿qué puedes comentar sobre la operación realizada por el filtro digital *f[n]* sobre la señal *x[n]*?

Parte B: Velocidad de cálculo en los dominios del tiempo y la frecuencia
En este ejercicio deberás ingeniártelas para mostrar que es más barato, desde el punto de vista computacional, 1.transformar dos señales, 2.multiplicar las transformadas, y 3.antitransformar el resultado; que realizar la convolución directamente en el dominio del tiempo. Debes generar un gráfico en el que compares los tiempos de cómputo empleados para realizar convoluciones en el tiempo y vía FFT para secuencias discretas de distinta longitud. Considera las siguientes sugerencias:
1.- usa señales aleatorias de longitud 2^n, para generar este tipo de señales usa la función *rand*
2.- estudia la función *cputime* y cómo puede ser usada para medir el tiempo de cómputo una operación
3.- para tener mejores estimados de los tiempos de cómputo repite muchas veces (100 ó más) una operación y calcula el tiempo de ejecución promedio.

Trabajo práctico #5.
Para realizar en la sesión práctica del 28/04/04

Tratamiento de señales estocásticas.
Desde la ventana de comandos de MATLAB, usa el comando *help* para averiguar sobre el uso de las siguientes funciones: *randn, cumsum, hist, mean, std, conj, fftshift, save, load, wavrecord, wavplay*.

Parte A: Algunas señales aleatorias de interés.
En este ejercicio debes generar tres señales aleatorias: ruido uniforme, ruido blanco ó gaussiano y movimiento browniano (esta señal se genera integrando el ruido blanco), y construir histogramas para cada una de ellas. Grafica en una misma figura las tres señales y sus respectivos histogramas. Responde las siguientes preguntas:
1.- ¿Cuál ó cuáles de estas señales son estacionarias? ¿Por qué?
2.- ¿Qué dicen los histogramas de estas señales? ¿Tienen la forma que esperabas?
3.- Calcula la media de aquellas señales estacionarias, ¿cuánto vale en cada caso?
4.- ¿Cómo harías para generar ruido blanco con una media y una desviación estándar dadas?

Parte B: Funciones de correlación y espectros de potencia
Genera un índice de tiempo *n* de *256* muestras y genera las siguientes señales: ruido blanco de media cero y desviación estándar unitaria, onda cuadrada de amplitud unitaria contaminada con ruido blanco de media cero y desviación estándar *0,7*, y suma de tres sinusoides de amplitud unitaria y frecuencias angulares $k\pi/32$ (con $k = 1, 2, 3$) contaminada con ruido blanco de media cero y desviación estándar 0,7. Para cada una de estas tres señales realiza los siguientes cómputos:
1.- calcula la magnitud de transformada de Fourier de la señal
2.- calcula el espectro de potencia como el producto de la transformada de Fourier con su conjugada
3.- calcula la función de autocorrelación como la transformada inversa del espectro de potencia
4.- genera cuatro gráficos sobre la misma figura y presenta en ellos: la señal, su función de autocorrelación, su transformada de Fourier (entre 0 y π) y su espectro de potencia (entre 0 y π)
5.- comenta sobre los resultados obtenidos

Parte C: Señal de voz
En este ejercicio grabarás y manipularás una señal de tu propia voz. Usa la función *wavrecord* para digitalizar una palabra, unas 15000 muestras deben ser más que suficiente. Una vez grabada la señal grafícala y define la mejor manera de cortarla para eliminar los posibles ruidos y/o segmentos sin sonido que se encuentren antes y después del contenido de la palabra. Corta la señal y guárdala en un archivo usando el comando *save*. Este archivo debes conservarlo, ya que en trabajos prácticos posteriores lo volverás a usar. Una vez guardada la señal de voz en el archivo, realiza las siguientes actividades:
1.- borra el *workspace* de MATLAB usando el comando *clear*
2.- recupera la señal de voz del archivo en que la guardaste
3.- grafica en una misma figura la señal de voz y su espectro de amplitudes
4.- contamina la señal de voz con ruido blanco
5.- reproduce, usando la función *wavplay*, la señal de voz original y la señal de voz contaminada
6.- escribe algunos comentarios y/o conclusiones sobre lo observado y escuchado.

Trabajo práctico #6.
Para realizar en la sesión práctica del 05/05/04

Muestreo y reconstrucción de señales.
Desde la ventana de comandos de MATLAB, usa el comando *help* para averiguar sobre el uso de las siguientes funciones: *upsample, downsample, interp, decimate, resample, convn, if-else-end.*

Parte A: El teorema de muestreo.
En este ejercicio explorarás en detalle el teorema de muestreo y el conocido fenómeno de *aliasing* ó solapamiento de espectros. Para empezar genera un vector de tiempo *t* de *201* muestras uniformemente espaciadas entre *–2 seg* y *2 seg*. Igualmente, genera un vector de frecuencia *f* para señales de tiempo continuo entre *–25 Hz* y *25 Hz* y con el mismo número de muestras que *t*. (Recuerda que en MATLAB las señales son siempre discretas por definición, pero aquí estaremos representando una señal continua).Una vez hecho esto, procede como se indica a continuación:
1.- genera la señal $y(t) = sin(2\pi k\ t + \pi/4)\ exp(-abs(k\ t))$, usando $k = 5$
2.- grafica en una misma figura la señal *y(t)* y su espectro de amplitud entre *–25 Hz* y *25 Hz*
3.- observa detenidamente el espectro, ¿cuál es aproximadamente la frecuencia máxima de *y(t)*?; de acuerdo con ese valor ¿da tu mejor estimado de la frecuencia de Nyquist para esta señal?
4.- muestrea la señal “continua” que generaste en el paso 1 usando un período de muestreo de *0,04 seg* (ayuda: si haz hecho todo bien hasta ahora, esto equivale a hacer cero las muestras pares de la señal)
5.- genera el filtro de reconstrucción *g(t)* según la expresión vista en la sesión de teoría
6.- recupera la señal original calculando la convolución entre la señal muestreada y el filtro (te recomiendo usar la función *convn* con la opción *‘same’*)
7.- calcula el error relativo de reconstrucción, definido como: $error = \Sigma(y_{original}-y_{recuperada})^2 / \Sigma(y_{original})^2$
8.- repite los pasos 4 al 7 usando períodos de muestreo *0,04 + k 0,02* para *k* desde *1* hasta *18* y genera un gráfico de el error relativo de reconstrucción vs el período de muestreo, ¿qué te dice el gráfico? ¿puedes inferir a partir de que momento comienza a ocurrir el *aliasing*? ¿cómo se relaciona este resultado con la frecuencia de Nyquist que estimaste en el paso 3?
9.- sólo para los períodos de muestreo *0,06, 0,10, 0,16* y *0,20*; genera tres gráficos en una misma figura y grafica: en el primero, la señal original *y(t)* y las muestras tomadas; en el segundo, los espectros de amplitud (entre *–25 Hz* y *25 Hz*) de la señal muestreada y del filtro de reconstrucción correspondiente; y en el tercero, la señal recuperada y las muestras de la señal original a partir de las cuales hiciste la reconstrucción.

Parte B: Remuestreo de señales discretas.
Según lo visto en clase, para cambiar de la tasa de muestreo de una señal discreta no es necesario convertir de digital a analógico y luego muestrear de nuevo. Esta operación puede hacerse completamente en el dominio discreto. Revisa el funcionamiento de la función *resample* de MATLAB y realiza las siguientes actividades:
1.- lee el archivo que contiene la señal de voz que grabaste en el trabajo práctico #5
2.- usa la función *resample* para cambiar la tasa de muestreo de la voz por factores de 3/4 y 4/3
3.- escucha la señal original y las dos nuevas señales generadas, ¿qué diferencias percibes entre las tres señales? ¿cómo puedes explicar lo que escuchaste?
4.- grafica en una sola figura los espectros de magnitud de las tres señales en el rango de frecuencia angular comprendido entre *0* y $\pi/4$; ¿qué diferencias observas entre los tres espectros?

Trabajo práctico #7.
Para realizar en la sesión práctica del 12/05/04

La Transformada Z para el análisis de sistemas LIT.
Desde la ventana de comandos de MATLAB, usa el comando *help* para averiguar sobre el uso de las siguientes funciones: *roots, freqz, zplane, zp2tf, tf2zp, log10*.

Parte A: Respuesta en frecuencia y diagrama de polos y ceros.
Considera los sistemas discretos descritos por las siguientes ecuaciones en diferencias:
Sistema 1: $y[n] = 0{,}5\, x[n] + 1{,}2728\, y[n-1] - 0{,}81\, y[n-2]$
Sistema 2: $y[n] = 0{,}9009\, x[n] + 1{,}001\, x[n-1] - 0{,}9\, y[n-1]$
Sistema 3: $y[n] = x[n] + 1{,}8507\, x[n-1] + 2{,}17\, x[n-2] + 1{,}4045\, x[n-3] + 0{,}5184\, x[n-4]$
Sistema 4: $y[n] = x[n] + x[n-1] + x[n-2]$
Sistema 5: $y[n] = 0{,}2\, x[n] - 0{,}1848\, x[n-1] + 0{,}05\, x[n-2] - 1{,}1\, y[n-1]$
y realiza las siguientes actividades para cada uno de ellos (nota importante: ten cuidado cuando copies los coeficientes de las ecuaciones en diferencias, no se acepta la vieja excusa 'es que copie mal'):
1.- escribe la expresión analítica de la función de transferencia $H(z)$ y calcula el valor de $h[0]$ usando el teorema del valor inicial de la Transformada Z
2.- calcula la respuesta impulsiva $h[n]$ para valores de n comprendidos entre -10 y 30, y verifica que el valor de $h[0]$ coincida con el que calculaste en el paso anterior
3.- calcula la respuesta en frecuencia $H(e^{j\omega})$ para valores de w comprendidos entre 0 y 2π
4.- genera cuatro gráficos en una misma figura y grafica en ellos: el diagrama de polos y ceros de $H(z)$, la respuesta impulsiva $h[n]$, la amplitud en decibeles de la respuesta en frecuencia $H(e^{j\omega})$, y la fase en radianes de la respuesta en frecuencia (nota: el valor en decibeles, dB, de una amplitud k se calcula como $20\ Log_{10}(abs(k))$)
5.- de acuerdo con lo observado en las gráficas, da la descripción más completa posible del sistema en cuestión, esta descripción debe incluir aspectos tales como causalidad, estabilidad, tipo de fase, etc...

Parte B: Sistemas estables de segundo orden
En este ejercicio estudiarás la relación existente entre la ubicación de los polos en el plano complejo de cinco sistemas estables de segundo orden y el tipo de respuesta impulsiva de dichos sistemas. Las ubicación de los polos de los cinco sistemas considerados ($k = 0, 1, 2, 3, 4$) es $0{,}8\, e^{\pm jk\pi/4}$
1.- grafica en una misma figura las respuestas impulsivas y los diagramas de ceros y polos de los cinco sistemas considerados
2.- ¿qué puedes concluir de los resultados obtenidos?, justifica bien tus respuestas

Parte C: Promedio móvil de longitud 5
Considera el sistema cuya respuesta impulsiva está definida por $h[n] = u[n] - u[n-5]$. Este tipo de sistema se conoce con el nombre de filtro de promedio móvil (moving average) y es un excelente ejemplo de sistema de fase lineal.
1.- obtén expresiones analíticas para la respuesta en frecuencia $H(e^{j\omega})$ y para la ubicación de los ceros de la función de transferencia $H(z)$
2.- calcula la respuesta en frecuencia y la ubicación de los ceros usando MATLAB
3.- genera cuatro gráficos en una misma figura con las mismas especificaciones de los gráficos generados en el paso 4 de la Parte A, pero en este caso muestra sobrepuestos en las mismas gráficas los valores obtenidos con MATLAB y los valores calculados a partir de las expresiones analíticas obtenidas por ti en el paso 1.

Trabajo práctico #8.
Para realizar en la sesión práctica del 26/05/04

Diseño de filtros de respuesta impulsiva finita (FIR).
Desde la ventana de comandos de MATLAB, usa el comando *help* para averiguar sobre el uso de las siguientes funciones: *hanning, hamming, blackman, kaiser, filter, filtfilt, fir1, fir2, remez, mod*.

Parte A: Diferenciador en tiempo discreto.
Considera la siguiente respuesta en frecuencia: $H(e^{j\omega}) = j\omega\, e^{-j\omega M/2}$, $-\pi < \omega < \pi$; la cual corresponde a un sistema diferenciador en tiempo discreto con retardo $M/2$. Es fácil verificar que la respuesta impulsiva de este sistema tiene una extensión infinita. En este ejercicio diseñarás un filtro digital FIR que aproxime dicha respuesta en frecuencia. Procede de la siguiente manera:
1.- halla una expresión analítica para la respuesta impulsiva $h[n]$ del sistema en cuestión
2.- estudia con detenimiento la respuesta impulsiva obtenida, ¿qué ocurre con la muestra $n = M/2$ cuando M es par?, calcula el límite correspondiente para poder determinar el valor de $h[M/2]$ con M par
3.- considera $M = 10$ y calcula los valores de $h[n]$ para $n = 0{:}M$
4.- construye el filtro deseado $fd[n]$ multiplicando $h[n]$ por una ventana *hamming* de $M+1$ muestras
5.- calcula la DTFS de $fd[n]$ usando 256 puntos
6.- genera cuatro gráficos en una misma figura y presenta: en el primero, la ventana *hamming* usada; en el segundo, la respuesta impulsiva del filtro diseñado $fd[n]$; en el tercero, el espectro de amplitudes del filtro diseñado y el del filtro ideal; y en el cuarto, el espectro de fases del filtro diseñado y el del ideal
7.- repite los pasos del 3 al 6 considerando ahora $M = 9$
8.- de los dos filtros diseñados, ¿cuál crees tú que aproxima mejor al diferenciador ideal? ¿por qué?, guarda los coeficientes de dicho filtro porque lo vas a usar en la Parte B.

Parte B: Filtrado de una señal de voz.
En este ejercicio vas a aplicar el filtro que consideraste mejor aproximación en la Parte A a la señal de voz que grabaste en el trabajo práctico #5. Para este fin, vas a usar la función *filtfilt*. Investiga sobre lo que hace dicha función. ¿Qué diferencia hay entre usar *filtfilt* y *filter*? ¿Cuál es la ventaja fundamental ofrece *filtfilt*? Procede de la siguiente manera:
1.- lee el archivo que contiene la señal de voz
2.- genera una versión filtrada de la señal de voz usando *filtfilt* y el filtro diseñado en la Parte A
3.- escucha las dos señales usando la función *wavplay* (nota: es muy probable que la intensidad de la señal filtrada sea muy inferior, puedes multiplicarla por un factor para compensar su intensidad)
4.- grafica en una misma figura la señal de voz original y la señal de voz filtrada
5.- grafica en otra figura los correspondientes espectros de amplitud para un rango de ω entre 0 y $\pi/4$
6.- comenta sobre las diferencias que pudiste escuchar y observar entre ambas señales

Parte C: Filtro óptimo de respuesta arbitraria.
En este ejercicio diseñarás un filtro de respuesta arbitraria usando el algoritmo de *Parks-McClellan*. La respuesta deseada es la siguiente: amplitud unitaria en el intervalo de frecuencias ω comprendido entre 0 y $\pi/3$, amplitud nula en el intervalo entre $\pi/3$ y $3\pi/4$, y amplitud $0{,}5$ entre $3\pi/4$ y π.
1.- considera el diseño para tres longitudes de filtro diferentes: 11 muestras, 31 muestras y 61 muestras
2.- presenta, en una misma figura, los gráficos correspondientes a la respuesta impulsiva y a la respuesta en frecuencia obtenida (conjuntamente con la deseada), para los tres casos considerados.

Trabajo práctico #9.
Para realizar en la sesión práctica del 02/06/04

Diseño de filtros de respuesta impulsiva infinita (IIR).
Desde la ventana de comandos de MATLAB, usa el comando *help* para averiguar sobre el uso de las siguientes funciones: *butter*, *cheby1*, *cheby2*, *besself*, *disp*, *text*, *round*.

Parte A: Filtro Butterworth digital.
En este ejercicio diseñarás un filtro Butterworth digital de segundo orden y lo usarás para tratar una señal contaminada con un ruido coloreado (¡no blanco!). Procede de la siguiente manera:
1.- diseña analíticamente un filtro Butterworth digital pasabajos de segundo orden, con frecuencia de corte $\omega_c = \pi/3$; utiliza el método de la transformación bilineal y obtén una expresión para *H(z)*
2.- diseña, utilizando MATLAB, un filtro con exactamente las mismas especificaciones; compara los coeficientes de la respuesta en frecuencia *H(z)* de este filtro con los del diseñado en el paso 1
3.- genera cuatro gráficos en una misma figura y presenta en ellos el diagrama de polos y ceros del filtro diseñado, los coeficientes de su respuesta impulsiva para valores de n entre -10 y 30, y la amplitud (en decibeles) y la fase (en radianes) de su respuesta en frecuencia para el rango de frecuencias comprendido entre 0 y π (señala sobre estos mismos gráficos la ubicación de la frecuencia de corte ω_c)
4.- genera una sinusoide de 256 muestras que tenga un total de ocho ciclos exactos
5.- genera una señal ruidosa *y[n]* contaminando la sinusoide generada en el paso anterior con ruido distribuido uniformemente entre $-½$ y $½$
6.- pasa la señal *y[n]* tres veces por el filtro Butterworth diseñado en el paso 2
7.- grafica en una misma figura la señal ruidosa, la señal filtrada y los respectivos espectros de amplitud en decibeles; observa la acción del filtro sobre la señal ruidosa y escribe tus conclusiones.

Parte B: Ecualización de un canal de transmisión con distorsión.
En este ejercicio simularás el paso de la señal de voz que grabaste en el trabajo práctico #5 a través de un canal de transmisión con una fuerte distorsión lineal. Luego diseñarás un ecualizador para compensar los efectos del canal y recuperar la señal de voz original. Procede de la siguiente forma:
1.- investiga sobre el concepto de distorsión lineal y los tipos de distorsión lineal existentes; igualmente averigua que es un ecualizador, como funciona y como se diseña
2.- considera la siguiente ecuación en diferencias, la cual describe al canal de transmisión considerado:
$$y[n] = 0.25\,x[n] - 0.9284\,x[n-1] + 1.2677\,x[n-2] - 0.5153\,x[n-3] - 0.6368\,x[n-4] + 1.0498\,x[n-5] - 0.6413\,x[n-6] + 0.1551\,x[n-7] + 4.6001\,y[n-1] - 8.644\,y[n-2] + 8.2823\,y[n-3] - 4.0448\,y[n-4] + 0.8056\,y[n-5]$$
(nota: ten mucho cuidado al copiar los coeficientes de la ecuación, un pequeño error en uno de los coeficientes puede significar un canal de transmisión totalmente diferente)
3.- genera cuatro gráficos en una misma figura y presenta en ellos el diagrama de polos y ceros, la respuesta impulsiva (para n entre 0 y 300), y la amplitud (en dB) y fase de la respuesta en frecuencia del canal de transmisión considerado
4.- observa con detenimiento los gráficos generados; ¿qué características fundamentales tiene el canal de transmisión?, ¿qué tipo de distorsión esperas que ocurra?, ¿puedes diseñar un ecualizador?, ¿cómo?
5.- diseña un ecualizador para el canal de transmisión considerado y presenta en una misma figura dos gráficos: en el primero presenta las amplitudes en dB de las respuestas en frecuencia del canal y el ecualizador, y en el segundo gráfico presenta las fases; ¿qué observación interesante puedes hacer?
6.- genera una señal distorsionada pasando por el canal de transmisión la señal de voz que grabaste en el trabajo práctico #5, luego genera una señal ecualizada pasando por el ecualizador la señal distorsionada; escucha las tres señales: original, distorsionada y ecualizada; ¿qué escuchas?
7.- grafica en una misma figura las tres señales y sus respectivos espectros de amplitud para el rango de frecuencias comprendido entre 0 y $\pi/4$; ¿qué conclusiones puedes sacar de los gráficos?

Trabajo práctico #10.
Para realizar en la sesión práctica del 09/06/04

Estimación de parámetros vía deconvolución.

En este trabajo práctico vas a realizar un ejercicio que es práctica común en muchas aplicaciones de la vida real. Se trata de el problema de estimar los parámetros reales de un sistema a partir de una medición realizada (en el fondo, la Parte B del trabajo práctico #9 que realizaste la semana pasada constituye un problema de este tipo: estimaste los parámetros reales de una señal de voz a partir de una medición de la misma, la salida del canal de transmisión).

En este caso particular nos vamos a concentrar en un problema de prospección comúnmente enfrentado en la industria petrolera. Se trata de la estimación de la resistividad eléctrica de las distintas capas del subsuelo a partir de la medición realizada con una herramienta de *logging* (son herramientas que se introducen en los pozos petroleros para medir las propiedades del susbsuelo en función de la profundidad). Este problema es de mucho interés, ya que la presencia de arenas en las que se puede acumular el petróleo está generalmente relacionada a valores bajos de la resistividad eléctrica.

1.- considera el modelo de resistividad eléctrica del subsuelo presentado en la siguiente tabla:

Número de la capa	1	2	3	4	5	6	7
Espesor de la capa (mts)	40	12	12	6	12	10	36
Resistividad eléctrica	95	40	90	5	85	5	95

2.- considera una herramienta cuya respuesta impulsiva es *h(x) = exp(–((x-64)/15)^2)* con *x* en metros

3.- genera una representación discreta en distancia para la respuesta impulsiva de la herramienta utilizando una tasa de muestreo de un metro, quédate con los valores de *h[n]* para *n* entre *1* y *128*, y normalizalos de forma que $\Sigma h[n] = 1$

4.- genera una representación para el modelo del subsuelo utilizando la misma tasa de muestreo

5.- calcula una medición del susbsuelo utilizando la herramienta de *logging* y presenta en una misma gráfica el modelo original del subsuelo y la medición; comenta los resultados obtenidos

6.- construye un filtro de deconvolución dividiendo la Transformada de Fourier de una señal base *g[n]* entre la Transformada de Fourier de la respuesta impulsiva de la herramienta *h[n]*; te recomiendo usar una versión normalizada ($\Sigma g[n] = 1$) de la siguiente señal base *g[n] = exp(–((x-64)/2)^2)*

7.- genera dos gráficos en una misma figura y presenta: en el primero, los espectros de amplitud (en dB) de *h[n]* y del filtro de deconvolución; y en el segundo, los espectros de fase

8.- estima el modelo de resistividad eléctrica pasando la medición calculada en el paso 5 por el filtro de deconvolución diseñado en el paso 6

9.- presenta en una misma gráfica el modelo de resistividad original, la medición de la herramienta y el modelo estimado en el paso anterior, ¿qué conclusiones puedes sacar de los resultados obtenidos?; escribe tus comentarios.

Notas de interés:

1.- En el paso 6, es probable que tengas que sumarle a la Transformada de Fourier de *h[n]* una constante muy pequeña (por ejemplo 10^{-12}) con el objeto de evitar una posible “división por cero” al hacer la división de las Transformadas. Este proceso se denomina *spectral whitening* y es equivalente a añadir ruido blanco de muy baja potencia a la señal *h[n]*.

2.- Te recomiendo ampliamente que en este ejercicio realices todas las convoluciones en el dominio de la frecuencia, así te evitarás un terrible problema de precisión numérica (especialmente en el paso 8). Sólo por curiosidad, échale un vistazo a la respuesta impulsiva del filtro de deconvolución.

3.- En la práctica el problema de prospección es todavía más complicado por dos razones básicamente: primero, el modelo real del subsuelo es mucho más complejo; y segundo, por lo general la respuesta impulsiva de la herramienta de medición está mal caracterizada o es totalmente desconocida.