



Técnicas Digitales III

Trabajo práctico: Filtrado digital tipo IIR

1) El objetivo de este ejercicio es el de comparar las dimensiones de filtros FIR e IIR para un mismo tipo de filtrado.

a) Ejecute la función `iir_vs_fir.m`.

b) Analice las funciones `fir_kaiser_3400_44100.m` e `iir_elliptic_3400_44100.m` ¿Qué tipos de filtros implementan ambas funciones?.

c) Observe ambas respuestas en frecuencia. ¿Qué diferencias hay entre ambas respuestas? Grafique además las respuestas en fase y compárelas.

d) ¿Cuál es la dimensión del numerador del filtro FIR y cuántos coeficientes presenta la matriz SOS del filtro IIR? ¿A qué conclusión puede abordar?

2) El objetivo de este ejercicio es el de observar la respuesta de dos filtros IIR en punto fijo. Uno de ellos ha sido configurado correctamente para evitar problemas de *overflow*; el otro no.

a) Ejecute en MATLAB el comando

```
>> Hd = iir_simulink
```

Observe la variable que se han creado en el Workspace.

b) Ejecute en Simulink el archivo `iir.slx`. Analice el diseño y explique qué función cumple cada bloque. Ejecute el modelo y observe las señales en el *Scope*.

c) Haga doble click sobre los bloques *Sine Wave* y observe los campos *Data Types*. ¿Qué precisión numérica presentan ambas señales?.

d) Haga doble click sobre los bloques *IIR Low-pass 200 Hz GOOD* e *IIR Low-pass 200 Hz BAD*. Observe el campo *Data Types*. ¿Qué diferencias observa?.

e) Proponga diferentes soluciones para que el segundo filtro IIR también funcione correctamente.

3) El objetivo de este ejercicio es el de analizar el desempeño de filtros IIR de 2do orden tipo Direct I y Direct II descritos en lenguaje C bajo el entorno de MATLAB.

MATLAB permite ejecutar funciones desarrolladas en lenguaje C usando una función *wrapper* cuya finalidad es la actuar como interfaz entre MATLAB y C. Este wrapper debe adaptar las variables de entrada y salida entre ambos lenguajes.

a) Abra la función wrapper `iir_matlab_wrapper.c` y analice las diferentes secciones. Al final de la misma verá que se invoca a la función `iir_filter_I_float()`.

b) Las funciones en C que implementan diferentes filtros IIR de 2do orden se encuentran en el archivo `iir_filters.c`, donde hay varios tipos de filtros IIR. Ejecute en consola el comando

```
>> mex iir_matlab_wrapper.c iir_filters.c
```

La función `mex` es la encargada de compilar los archivos en C bajo MATLAB.

c) Abra en el editor de MATLAB el script `iir_matlab_to_C.m` y ejecútelo. Este se encarga de invocar la función `iir_matlab_wrapper` y de analizar su salida. Concéntrese en comparar las señales de salida para las funciones implementadas en MATLAB y en C.

d) Comente la línea de la función `iir_filter_I_float()`, descomente la línea de la función `iir_filter_II_float()` y vuelva a compilar. Ejecute nuevamente `iir_matlab_to_C.m`. Observe la señal filtrada de salida, ¿detecta alguna diferencia respecto a la salida para el filtrado tipo Direct I?

e) Abra el archivo `iir_filters.c` y analice las funciones que implementan los filtros IIR tipo Direct I y Direct II. Observe cómo la estructura de cada función se condice con el esquema de los filtros IIR tipo Direct I y Direct II, respectivamente, vistos en teoría ¿Qué diferencias presentan?

f) Escriba funciones similares en C para filtrado IIR tipo Direct I y Direct II pero en precisión punto fijo. Verifique su correcto funcionamiento bajo MATLAB.

g) Escriba funciones en C para implementar filtrado IIR tipo Direct I y Direct II de orden N. Verifique su correcto funcionamiento bajo MATLAB.