3 La transformada de Fourier

En esta sección se presentan algunas rutinas de Matlab de interés en relación con la utilización de transformadas de Fourier en análisis de circuitos.

En la primera de ellas se indica cómo representar el espectro de amplitudes y el de fases de una función periódica expresada como serie de Fourier en su versión trigonométrica. Los coeficientes de los elementos han de ser obtenidos aparte y las funciones que los describen son datos en la rutina de Matlab correspondiente. Podría considerarse el caso análogo en el que se parte de la serie de Fourier en su versión exponencial. Sin embargo, tal situación es ignorada en este texto, y ello por dos motivos. Por una parte, a raíz de lo expuesto el lector interesado puede deducir fácilmente cómo efectuar la representación correspondiente. Por otro lado, consideramos que la versión trigonométrica, en la que únicamente existen frecuencias positivas en el desarrollo en serie, es más ajustada a la realidad física que la serie de términos exponenciales, que incluye frecuencias negativas. El primer apartado se completa con la indicación acerca de cómo reconstruir una señal periódica a partir de una serie de Fourier de términos trigonométricos.

El segundo apartado se refiere a la representación del módulo de una función descrita mediante su transformada de Fourier y a cómo calcular la energía de la señal correspondiente de acuerdo con el teorema de Parceval.

La sección se completa, como las dos anteriores, con algunos ejercicios propuestos que el lector puede intentar resolver para comprobar el grado en el que ha asimilado los contenidos de aquélla.

3.1 Espectros y reconstrucción de una señal

A lo largo de este apartado nos referiremos a la rutina que vamos a presentar a continuación y a los resultados que proporciona.

```
%%%%% COEFICIENTES DE LA SERIE DE FOURIER       %%%%%

clear all;                   % Elimina variables utilizadas en otras rutinas

% Número de términos de la serie (entero, > 0), excluido av
n = 8;
k = 1: n;

% Coeficientes de la serie (expresiones algebraicas en función de k)
av = 7*pi;                   % (constante real)
ak = (6./k).*sin((4/3)*pi*k); bk = (6./k).*(1 - cos((4/3)*pi*k));
```

```
% Periodo de la función (> 0)
T0 = 0.12566;

% Número de periodos a representar (entero, > 0)
np = 4;

% Base de tiempos
tinicial = 0;                 % Instante inicial del primer periodo
inicial = tinicial - (np/2)*T0; final = - inicial; puntos = 1000;
t = linspace (inicial, final, puntos);

% Cálculo de módulos y fases
A = sqrt(ak.^2 + bk.^2); fase = atan2(bk, ak);

% Representación de módulos
subplot (3, 1, 1);
stem (0, av);
grid on;
xlabel ('Componente', 'FontName', 'Times', 'Fontsize', 14);
ylabel ('Módulo', 'FontName', 'Times', 'Fontsize', 14);
axis ([-0.5, n+0.5, -0.2*max(A), (3/2)*max(max(A), av)]);
hold on;
stem (k, A(k));
title ('Componentes de Fourier', 'FontName', 'Times', 'Fontsize', 24);

% Representación de fases
subplot (3, 1, 2);
stem (k, (180/pi)*fase(k));
grid on;
xlabel ('Componente', 'FontName', 'Times', 'Fontsize', 14);
ylabel ('Fase (º)', 'FontName', 'Times', 'Fontsize', 14);
axis ([-0.5, n+0.5, -(3/2)*abs(min((180/pi)*fase)), (3/2)*abs(max((180/pi)*fase))]);

% Señal reconstruida
subplot (3, 1, 3);
senyal = av;
k = 1;
while k<=n
senyal = senyal + A(k)*cos((2*pi*k/T0)*t + fase(k));
k = k + 1;
end
plot (t, senyal, 'b', 'LineWidth', 2);
grid on;
xlabel ('Tiempo (s)', 'FontName', 'Times', 'Fontsize', 14);
ylabel ('Señal', 'FontName', 'Times', 'Fontsize', 14);
axis ([inicial, final, -(3/2)*abs(min(senyal)), (3/2)*abs(max(senyal))]);

clear all;                    % Elimina las variables utilizadas en esta rutina
```

Como puede observarse, el usuario ha de facilitar los siguientes datos:

-Las expresiones matemáticas que definen el valor medio (av) o término inicial de la serie y los coeficientes del desarrollo en serie de Fourier original (ak, bk).

-El número de términos de la serie que han de ser tenidos en cuenta (n) además del inicial.

-El periodo de la función (T0).

-El número de periodos de la función que han de ser representados en la gráfica cuando se efectúe la reconstrucción de la señal.

-El instante (tinicial) en el que comienza el periodo más próximo a t = 0.

Con estos datos la rutina calcula y representa los módulos y las fases de los distintos términos. Obsérvese que en estos cálculos la rutina utiliza las dos instrucciones que se mencionan seguidamente.

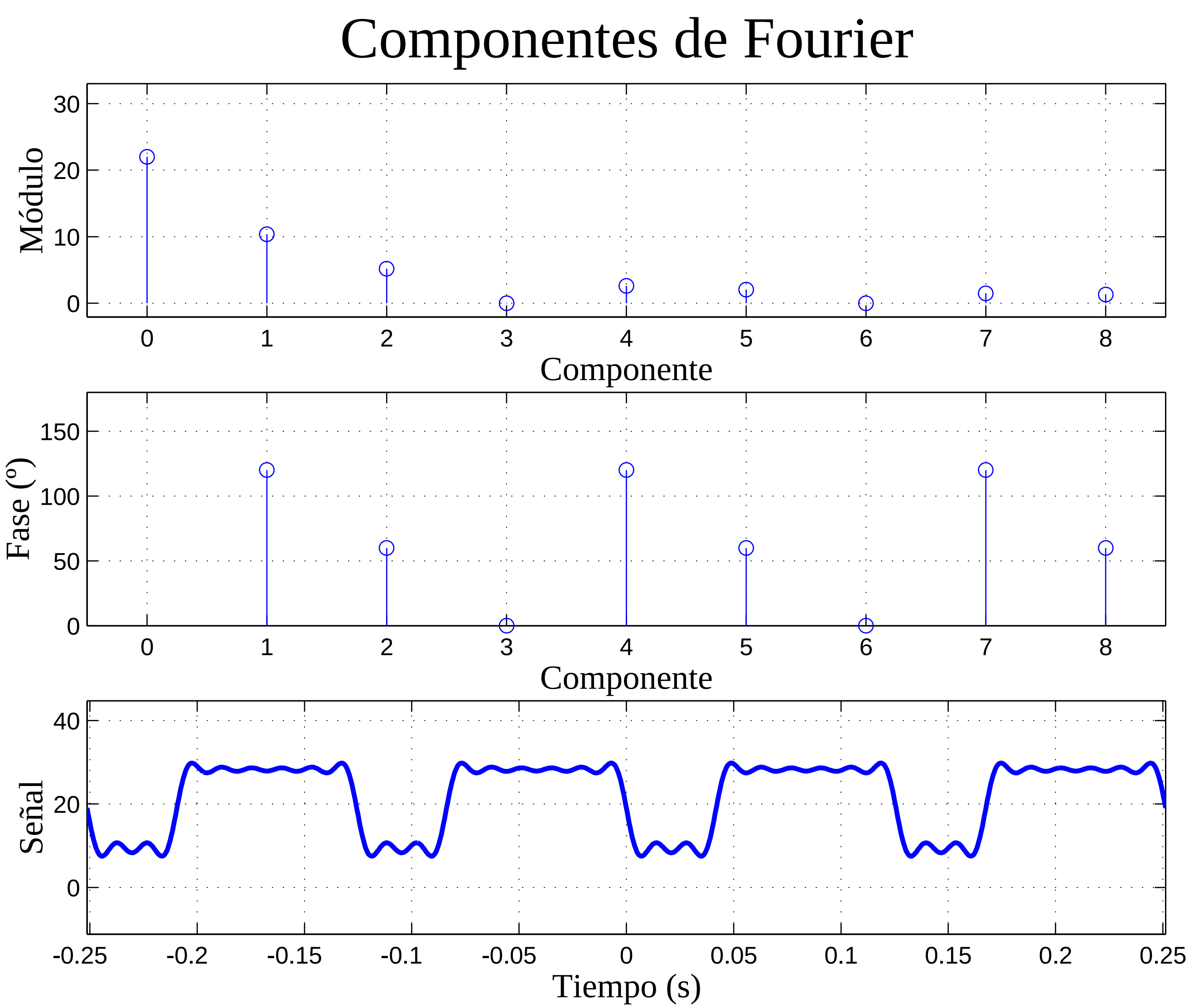
```
sqrt(x)                % Obtiene la raíz cuadrada de x

atan2(y, x)            % Obtiene el arco cuya tangente es el cociente entre y y x
                       % teniendo en cuenta sus signos respectivos,
                       % con lo que el ángulo proporcionado está comprendido entre -π y π
```

Una instrucción similar a la segunda, atan(y/x), reduce todos los ángulos al intervalo comprendido entre $-\pi/2$ y $\pi/2$, lo cual no resulta correcto en este caso. En este sentido conviene recordar que todas las instrucciones trigonométricas, tanto directas como inversas, operan con los ángulos expresados en radianes.

A partir de los cálculos realizados hasta el momento la rutina presenta por separado los espectros de amplitudes y de fases. Obsérvese que el eje de abscisas es el índice de cada término (k), o, lo que es equivalente, el orden del armónico al que le corresponde una amplitud y una fase dadas (la frecuencia de cada armónico es $2\pi k/T0$). Los índices se estructuran como vector mediante la instrucción

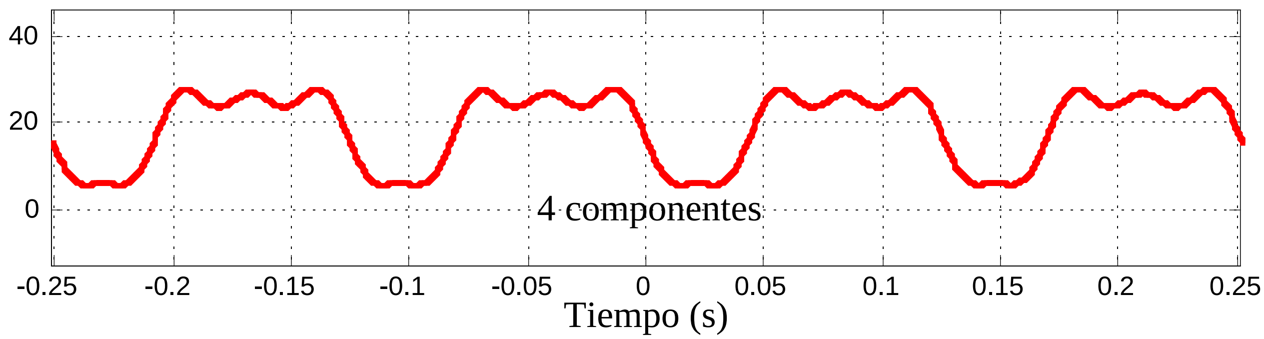
```
k = ini:inc:fin        % Define un vector de valores de k

% ini: primer valor del vector
% fin: último valor del vector
% inc: diferencia entre dos valores consecutivos del vector. Si es la unidad, puede omitirse
```

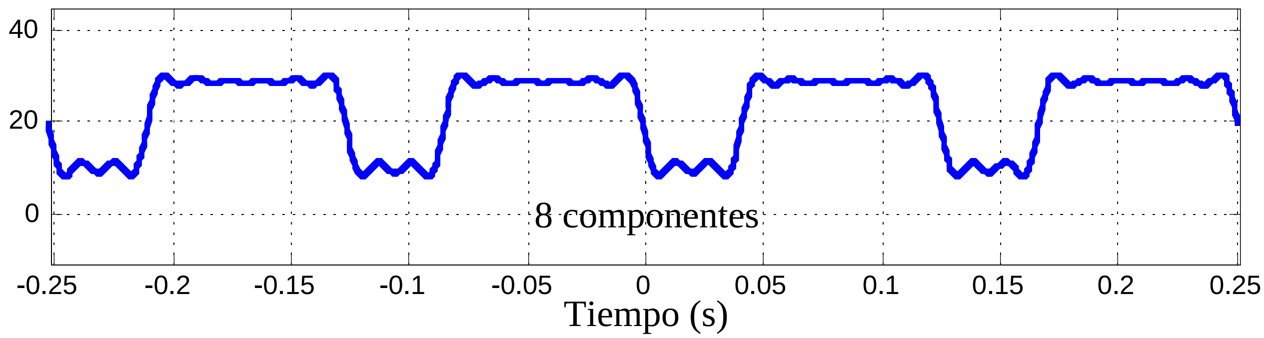
Finalmente, la rutina reconstruye la señal original (la que ha sido expresada como serie de Fourier) mediante un bucle en el que en cada ciclo se va añadiendo un nuevo término a la serie hasta completar todos los especificados. En las condiciones indicadas la ejecución de la rutina produce los resultados que se muestran seguidamente. Obsérvese que la serie de Fourier es la equivalente de un tren periódico de pulsos rectangulares.

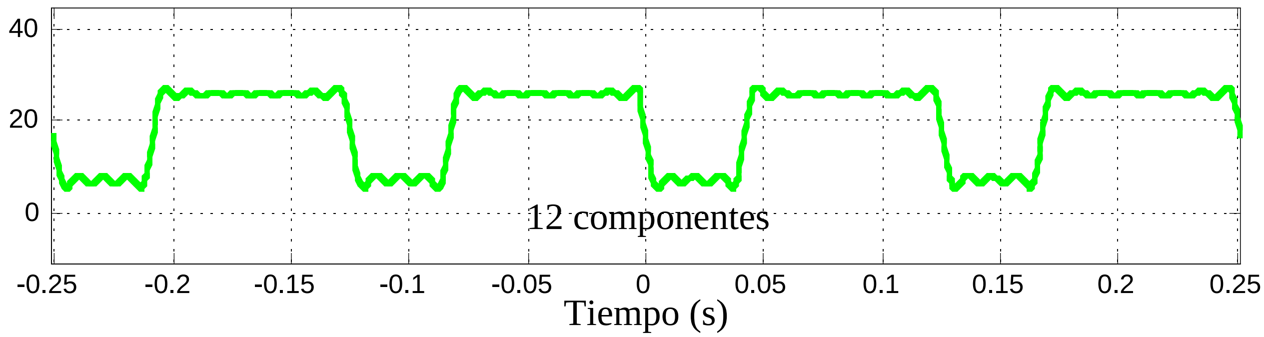


Obviamente, la calidad de la reconstrucción de una señal periódica a partir de su serie de Fourier (o, lo que es lo mismo, el grado de aproximación con la que una serie de Fourier representa una señal periódica) depende del número de términos que contenga la serie. En la rutina que conduce a la última figura se han considerado 8 términos (siempre sin contar el que corresponde al valor medio de la función). En la figura siguiente se utiliza la misma rutina para comparar lo que sucede cuando se eligen distintos números de términos. De cada representación se ha seleccionado únicamente el recuadro correspondiente a la reconstrucción de la señal. Como puede observarse, cuanto mayor es dicho número, mayor es la calidad de la reconstrucción. Téngase en cuenta que la identidad entre una función periódica y su desarrollo en serie de Fourier se produce cuando se consideran infinitos términos en esta última.

Reconstrucción de una señal







3.2 Módulo de la transformada de Fourier

La rutina que se presenta en este apartado permite representar el módulo de la transformada de Fourier de una función no periódica. La transformada de Fourier ha de ser definida por el usuario en una función (funcfourier). Como puede observarse, esta función tiene dos entradas:

-El rango de frecuencias, definido en la rutina principal.

-Un número cualquiera. Si este número es 1, la función devuelve el módulo de la transformada de Fourier; en cualquier otro caso, devuelve el cuadrado de dicho módulo.

El usuario puede definir el tipo de variación (lineal o logarítmica) de la frecuencia angular, tomada como eje de abscisas de la representación. La instrucción para efectuar la representación (plot o semilogx) ha de ser coherente con tal definición.

La rutina principal también calcula la energía de la señal cuando está conectada a una resistencia de 1 Ω, para lo que utiliza el teorema de Parseval. La fómula que determina la energía se implementa mediante la instrucción

`quad (‘function’, vinicial, vfinal, [], [])` % Instrucción de integración

% `function`: nombre de la función a integrar
% `vinicial, vfinal`: límites inferior y superior, respectivamente, de la integral
% `[]`: parámetro relacionado con la precisión del cálculo, ignorado en este apartado

Como puede observarse, en la rutina que sigue se añade un parámetro (2) a la instrucción para calcular la integral. Es el parámetro que, de acuerdo con lo indicado anteriormente, selecciona la salida de la subrutina de función que interesa (el cuadrado del módulo en este caso). Obviamente, el usuario ha de especificar los límites de la integración. Se sugiere al lector que preste atención a la forma en la que en la subrutina de función se establece el resultado que aquélla ha de entregar a la rutina principal.

```
function x = funcfourier(w, tipo)

     A = 120;
     c = 24;
     trans = A./(c + i*w);
     if (tipo == 1)
      x = abs(trans);
     else
      x = (abs(trans)).^2;
     end

return;
```

```
%%%%% TRANSFORMADA DE FOURIER       %%%%%

clear all;                   % Elimina variables utilizadas en otras rutinas

% Rango de frecuencias
inicial = -3; final = 3; puntos = 1000;
w = linspace(inicial, final, puntos);          % Lineal
% w = logspace(inicial, final, puntos);        % Logarítmico

% Representación gráfica
modulo = funcfourier(w, 1);
plot(w, modulo, 'b', 'LineWidth', 2);          % Lineal
% semilogx(w, modulo, 'b', 'LineWidth', 2);    % Logarítmico
grid on;
xlabel('Frecuencia angular (rad/s)', 'FontName', 'Times', 'Fontsize', 14);
ylabel('Módulo', 'FontName', 'Times', 'Fontsize', 14);
axis([w(1), w(puntos), -(1/4)*max(modulo), (3/2)*max(modulo)]);

% Energía de la función
vinicial = 24;                          % Extremo inferior de integración (> 0)
vfinal = 48;                            % Extremo superior de integración
ener = quad('funcfourier', vinicial, vfinal, [], [], 2)/pi

clear all;                   % Elimina las variables utilizadas en esta rutina
```

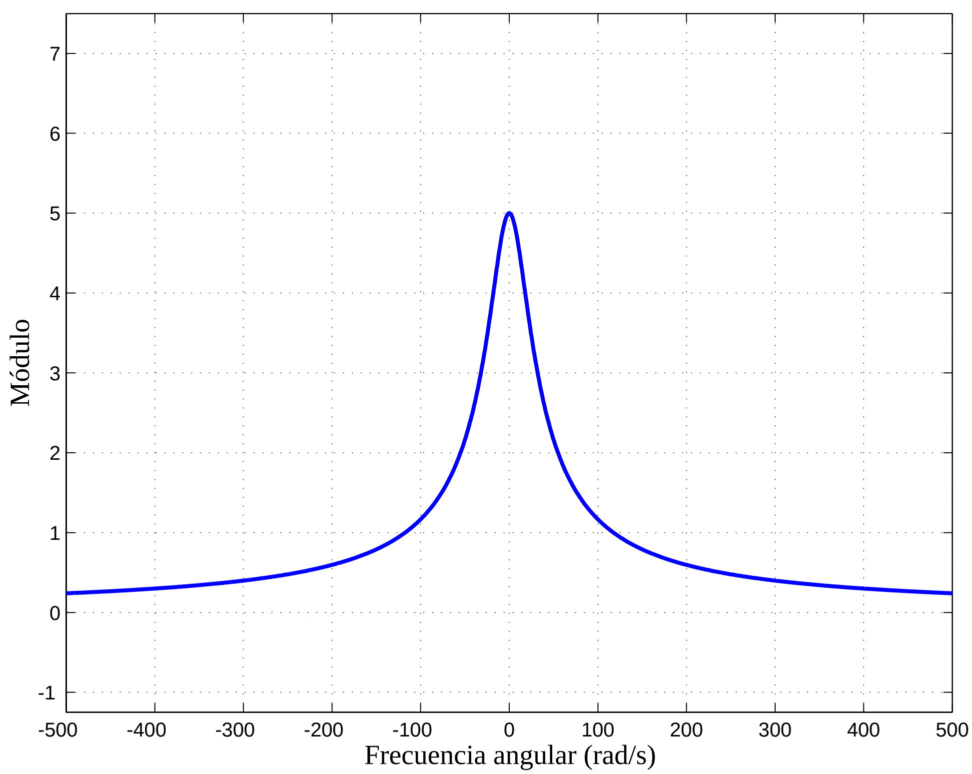
La rutina y la subrutina indicadas proporcionan el siguiente resultado:

Transformada de Fourier

$$F(\omega) = \frac{A}{c + j\omega}$$

$A = 120;\ c = 24$

```
ener = 61.4498
```

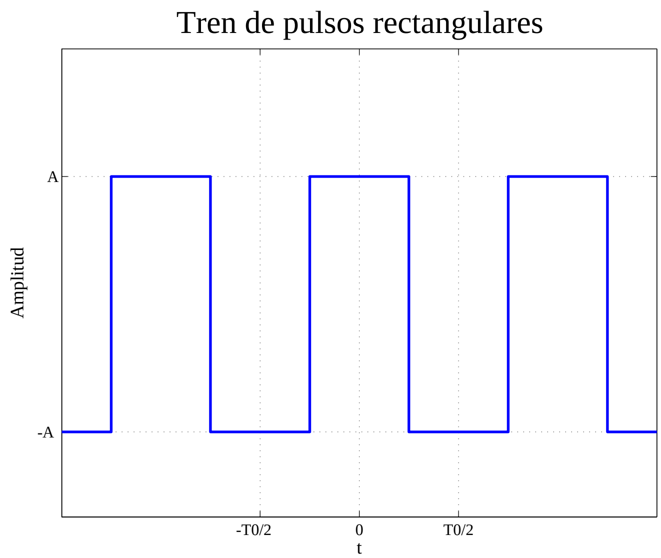


3.3 Ejercicios propuestos

Ejercicio 10

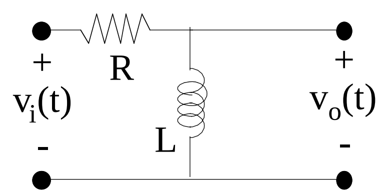
Obtener el espectro de amplitudes y fases (desarrollo trigonométrico) de la función periódica adjunta.

$$V_m = 15\pi \text{ V};\ T_0 = 4\pi \text{ ms}$$



Ejercicio 11

Obtener la tensión de salida del circuito de la figura y su espectro de amplitudes y fases.

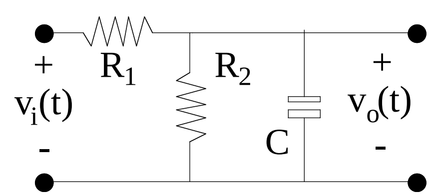


$$V_m = 15\pi \text{ V};\ T_0 = 4\pi \text{ ms};\ R = 10\ \Omega;\ L = 10 \text{ mH}$$

$$v_i(t) = \sum_k \frac{4V_m}{\pi k} \operatorname{sen}\left(\frac{\pi k}{2}\right) \cos(k\omega_0 t);\ \omega_0 = \frac{2\pi}{T_0}$$

Ejercicio 12

Obtener la energía de la tensión de salida en una resistencia de 1 Ω.



$$V_m = 30 \text{ V};\ a = 1;\ R_1 = 20\ \Omega;\ R_1 = 8\ 0\ \Omega;\ C = 0.125 \text{ F}$$

$$v_i = V_m e^{-a \times \operatorname{mod}(t)}$$