

Seminarios con Matlab

2017

MATEMATICA AVANZADA - SEMINARIO 1

Temas de variable compleja y análisis de señales en el dominio temporal

I) Algunos cálculos básicos en variable compleja utilizando Matlab

1. a) Obtener las raíces, z_i ($i=1,2,3$), del siguiente polinomio:

$$f(z)=z^3+z^2+2z+8$$

Matlab nos permite hacer este cálculo utilizando la función 'roots'.

```
>> n=[1 1 2 8]; % Vector de coeficientes del polinomio.
>> zi=roots(n)
zi =
-2.0000
0.5000 + 1.9365i
0.5000 - 1.9365i
```

- b) Obtener las raíces del denominador del integrando del ejercicio 17. b) de la guía 1.

2. Halle la expresión en coordenadas polares de las raíces de a).

Las raíces halladas en a) están expresadas en coordenadas cartesianas. Con Matlab es muy sencillo obtener la expresión en coordenadas polares, utilizando las funciones 'abs' para obtener el módulo y 'phase' para obtener el ángulo o fase.

```
>> % Módulo y fase de zi

>> zm=abs(zi) % Devuelve los módulos de cada elemento del
               % vector zi
zm =
2.0000
2.0000
2.0000

>> zf=phase(zi) % Devuelve las fases de cada elemento del
                 % vector zi
zf =
3.1416
1.3181
-1.3181

% Tenga en cuenta que la fase está dada en radianes.
```

3. Otra aplicación de mucha utilidad es la que permite obtener una expansión en fracciones simples de un cociente de polinomios, cálculo que suele resultar trabajoso cuando se tienen 3 o más raíces en el denominador, o cuando las mismas son complejas. Por ejemplo, si deseamos escribir la siguiente función como una suma de fracciones simples:

$$F(z) = \frac{2z+1}{z^3+4z^2+13z}$$

Matlab nos permite simplificar los cálculos mediante la instrucción 'residue', que da las raíces del denominador (polos de $F(z)$, **p**), los coeficientes de las fracciones (residuo de $F(z)$ en cada punto singular, **r**) y el término directo, **k**. Las entradas que deben darse a la función 'residue' son los coeficientes del numerador y el denominador.

```
>> % Para la función dada:

>> n=[0 0 2 1]; % Coeficientes del numerador.
>> d=[1 4 13 0]; % Coeficientes del denominador.
>> [r,p,k]=residue(n,d)

r =
-0.0385 - 0.3077i
-0.0385 + 0.3077i
0.0769
p =
-2.0000 + 3.0000i
-2.0000 - 3.0000i
0
k =
[]
```

Podemos convertir los números complejos a polares:

```
rm=abs(r)
rm =
0.3101
0.3101
0.0769

>> rf=phase(r)
rf =
-1.6952
1.6952
0
```

Ahora estamos en condiciones de escribir nuestra función como suma de fracciones simples:

$$F(z) = \frac{A}{z - z_1} + \frac{B}{z - z_2} + \frac{C}{z - z_3}$$

$$F(z) = \frac{0.3101e^{-j1.6952}}{z + 2 - j3} + \frac{0.3101e^{j1.6952}}{z + 2 + j3} + \frac{0.0769}{z}$$

Para polos complejos conjugados la suma de fracciones simples puede expresarse también como:

$$F(z) = \frac{A_1 + B_1 z}{(z - z_1)(z - z_2)} + \frac{C_1}{(z - z_3)}$$

$$F(z) = \frac{0.6202[(z + 2)\cos(1.6952) - 3\sin(1.6952)]}{(z + 2)^2 + 9} + \frac{0.0769}{z}$$

a) Halle la expansión en fracciones simples de $F(z) = \frac{4z^2 - 3}{z^3 - z^2 - 5z - 3}$.

b) Halle la expansión en fracciones simples de $F(z) = \frac{z^3 + z - 8}{z^3 - 2z^2 - 5z - 6}$

Nota: utilice el 'help' de Matlab para interpretar los resultados.

II) Análisis de señales en el dominio temporal

a) Grafique la función $f_1(t) = \sin(2/t)$ y observe la existencia de infinitos máximos y mínimos.

```
>> t=[0.001:0.001:1]; % vector de tiempo
>> f1=sin(2./t); % función (observe que para dividir por un vector
% punto a punto se utiliza './'
>> plot(t,f1) % grafica (vector de abscisas, vector de ordenadas)
```

b) Calcule la función $f_2(t) = \text{sinc}(t) = \sin(t)/t$

```
>> t=[-20:0.1:20]; %vector de tiempo que incluye al cero
>> f2=sin(t)./t; % función
```

En algunas versiones de Matlab aparece un mensaje de “warning”. Esta advertencia se debe a que el vector de tiempo incluye el cero y Matlab no puede hacer la cuenta para ese valor. Para solucionar esto definimos nuevamente el vector ‘t’.

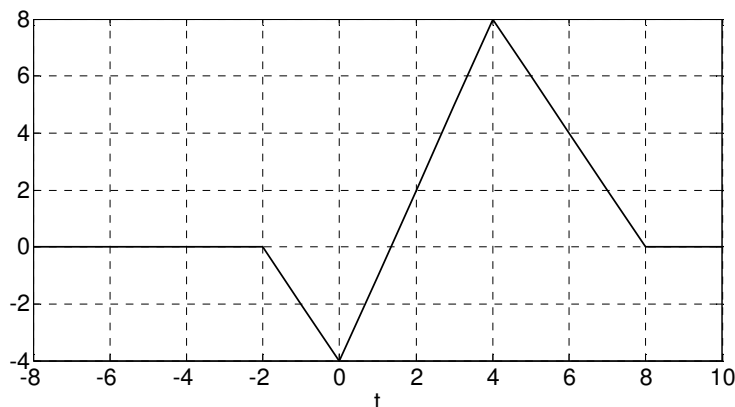
```
>> t=[-20:0.3:20]; %vector de tiempo que NO incluye al cero
>> f2=sin(t)./t; % función
```

Grafique $f_2(t)$, $f_2(2t)$, $f_2(t/2)$ y $f_2(t-10)$ en un mismo gráfico.

```
>> f2a=sin(2*t)./(2*t);
>> f2b=sin(t/2)./(t/2);
>> f2c=sin(t-10)./(t-10);
>> plot(t,f2,'k',t,f2a,'r',t,f2b,'b',t,f2c,'g') % grafica las señales
% ('k',negro,'r',rojo,'b',azul,'g',verde)
>> grid on % dibuja la grilla
```

c) Con Matlab podemos generar nuestras propias funciones y luego utilizarlas para hacer los cálculos que se deseen.

c1- Crear una función con Matlab que represente el siguiente gráfico:



Para crear una función, en la barra de Matlab seleccione ‘File’, ‘New’, ‘M_file’. Esto abrirá una nueva página del editor. Allí transcriba el siguiente texto, que corresponde a la función del gráfico. En la primera línea se declara que se trata de una función denominada ‘f3’, cuyo argumento es ‘t’. Esto debe tenerse en cuenta cuando se llama a la función, que será invocada como f3(argumento).

```
function y=f3(t)

% Calcula la expresión funcional correspondiente a cada intervalo de t.

y1=-4-2*t;
y2=-4+3*t;
y3=16-2*t;

% Une las diferentes expresiones funcionales en sus respectivos intervalos
% de validez.

y=y1.*(-2<t & t<=0)+y2.*(0<t & t<=4)+y3.*(4<t & t<=8);
```

Guarde la función con el nombre correspondiente. Para ello, en el editor vaya a ‘File’, ‘save as...’, asegúrese de que esté en el directorio o carpeta deseado y nombre a la función f3.m. Guárdela.

Interprete la última línea del programa anterior.

c2- Obtenga las siguientes funciones desplazando, comprimiendo, dilatando y cambiando la escala de f3:

- $g_1=3*f_3(t-5)$
- $g_2=0.5*f_3(3t)$
- $g_3=-2*f_3((t+1)/2)$

Grafique las cuatro funciones en el mismo gráfico y evalúe lo obtenido comparando con lo que hubiera esperado obtener. Para utilizar la función asegúrese de estar en el directorio donde la misma está guardada. (Observe ‘Current directory’)

```
>> t=[-8:0.1:15]; % Genera el vector de tiempo
>> g1=3*f3(t-5);
>> g2=0.5*f3(3*t);
>> g3=-2*f3((t+1)/2);
%% Dibujamos las tres funciones en una misma figura, junto con la
%% señal original.
>> g0=f3(t);
>> subplot(2,2,1); plot(t,g0);grid on
>> subplot(2,2,2); plot(t,g1);grid on
>> subplot(2,2,3); plot(t,g2);grid on
>> subplot(2,2,4); plot(t,g3);grid on
```

d) Repita el inciso c) para la función $f_4(t) = (1 + \sin(\pi t)) e^{-t/2} u_s(t)$.

MATEMÁTICA AVANZADA - SEMINARIO 2

Matrices

I) Operaciones básicas con matrices utilizando Matlab.

Ingresa dos matrices de 4x4, **A** y **B**:

$$\mathbf{A} = \begin{pmatrix} 1 & 2 & 4 & -1 \\ 0 & -1 & 3 & 1 \\ 1 & 5 & 0 & 1 \\ 2 & -1 & 3 & 1 \end{pmatrix} \quad \mathbf{B} = \begin{pmatrix} 0 & 1 & 2 & 0 \\ 1 & 1 & -2 & 1 \\ -3 & 0 & 1 & 2 \\ 1 & 2 & -1 & 3 \end{pmatrix}$$

```
>> A=[1 2 4 -1;0 -1 3 1;1 5 0 1;2 -1 3 1];  
>> B=[0 1 2 0;1 1 -2 1;-3 0 1 2;1 2 -1 3];
```

Realizar con ellas las siguientes operaciones:

- Matriz inversa de **A** y matriz inversa de **B**.

```
>> Ai=inv(A)  
>> Bi=inv(B)
```

- Matriz inversa de (**A*B**).

Comparar el resultado con $\mathbf{A}^{-1}\mathbf{B}^{-1}$ y con $\mathbf{B}^{-1}\mathbf{A}^{-1}$.

- Matriz traspuesta de **A** y matriz traspuesta de **B**.

```
>> At=A'  
>> Bt=B'
```

- Matriz traspuesta de (**A*B**).

Verifique que se cumple la igualdad $(\mathbf{AB})^T = \mathbf{B}^T \mathbf{A}^T$ y que $(\mathbf{AB})^T \neq \mathbf{A}^T \mathbf{B}^T$.

- Determinante de **A** y determinante de **B**.

```
>> Da=det(A)  
>> Db=det(B)
```

- Determinante de (**AB**).

Verifique que se cumple la igualdad: $\det(\mathbf{AB}) = \det(\mathbf{A})\det(\mathbf{B})$.

- Autovalores (**L**) y autovectores (**V**) de **A** y de **B**.

```
>> eig(A) %Vector de autovalores  
% Si además se quieren obtener los autovectores:  
>> [V,L]=eig(A) % V: matriz con autovectores en las columnas  
% L: matriz diagonal de autovalores
```

Verifique que se cumple la igualdad: $\mathbf{A}^* \mathbf{V} = \mathbf{V}^* \mathbf{L}$.

```
% Puede hacerse para cada uno de los autovalores y autovectores  
>> A*V(:,1) % Producto de la matriz A por la primera columna  
% de la matriz V (Primer autovector)  
>> V(:,1)*L(1,1) % Productor del elemento (1,1) de la matriz L  
% con la primera columna de la matriz V.
```

```
% También puede comprobarse directamente  
>> A*V  
>> V*L
```

II) Propiedades de la matriz de transición de los estados $\boldsymbol{\varphi}(t)=e^{At}$.

Comprobaremos algunas propiedades de la matriz de los estados. Para ello consideremos la matriz:

$$\boldsymbol{\varphi}(t) = \begin{pmatrix} 2\frac{\sqrt{3}}{3}e^{2t}\sinh(\sqrt{3}t) + e^{2t}\cosh(\sqrt{3}t) & \frac{\sqrt{3}}{3}e^{2t}\sinh(\sqrt{3}t) \\ -\frac{\sqrt{3}}{3}e^{2t}\sinh(\sqrt{3}t) & -2\frac{\sqrt{3}}{3}e^{2t}\sinh(\sqrt{3}t) + e^{2t}\cosh(\sqrt{3}t) \end{pmatrix}$$

Esta matriz ha sido guardada como una función de Matlab: `fi.m`, cuyo argumento es t .

Para evaluar en un valor dado de $t=t_i$ simplemente debe escribirse la expresión

```
>> fi(ti)
```

- Compruebe la propiedad que establece que $\boldsymbol{\varphi}(0)=I$.

```
>> fit0=fi(0)
```

- Verifique que se cumple la propiedad $\boldsymbol{\varphi}(t_1+t_2) = \boldsymbol{\varphi}(t_1)\boldsymbol{\varphi}(t_2)$ siendo $t_1=2$ y $t_2=3$.

```
>> fit2=fi(2)
```

```
>> fit3=fi(3)
```

```
>> fit5=fi(5)
```

```
>> fit2*fit3
```

- Dado el vector de estados iniciales, $x(0)=[1;-3]$, calcule $x(t)$ para $t=2$, $t=3$ y $t=5$ utilizando la propiedad $\mathbf{x}(t)=\boldsymbol{\varphi}(t-t_0)\mathbf{x}(t_0)$.

¿Qué observa en los valores de la solución a medida que transcurre t ? ¿Cómo se relaciona esto con la estabilidad del sistema? ¿Es un sistema estable? Justifique.

.....

MATEMÁTICA AVANZADA - SEMINARIO 3

Serie de Fourier

1) Considere nuevamente el ejercicio 14 de la guía de TP nº4, en el que se encontró que

$$V_0(t) = 12 \left[\frac{2\sqrt{2}}{\pi} + \sum_{n=1}^{\infty} \frac{4\sqrt{2}}{\pi} \frac{1}{1-4n^2} \cos(2n\pi 100t) \right]$$

a) Grafique el espectro de amplitud y fase para las cinco primeras armónicas.

¿Qué características observa en ambos espectros?

.....

b) Grafique la señal reconstruida usando las cinco primeras armónicas.

¿Qué tan buena es dicha reconstrucción?, ¿a qué se debe eso?

.....

.....

c) Calcule la potencia media de la señal y la contenida en las primeras cinco armónicas.

¿Qué porcentaje de la potencia total está contenido en el cálculo anterior?

.....

d) Repita los incisos anteriores para otros números de armónicas (por ejemplo, 3 y 10).

Solución:

De la expresión de $V_0(t)$, obtenemos que $\frac{a_0}{2} = \frac{24\sqrt{2}}{\pi}$ y $a_n = \frac{48\sqrt{2}}{\pi} \frac{1}{1-4n^2}$

y por lo tanto $c_0 = \frac{24\sqrt{2}}{\pi}$ y $c_n = \frac{24\sqrt{2}}{\pi} \frac{1}{1-4n^2}$

a) Empecemos por definir una cantidad de armónicas (supongamos $n=5$), calculemos y grafiquemos los respectivos espectros de amplitud y fase.

```
n=5 %Consideremos las cinco primeras armónicas
Cn=24/pi*sqrt(2)./(1-4*[-n:n].^2); %Calculamos los Coeficientes de Fourier
desde n=-5 a n=5
amplitud=abs(Cn); %Calculamos el espectro de amplitud
fase=phase(Cn); %Calculamos el espectro de fase
fase(1:n)=-fase(1:n); %Cambiamos por convención pi a -pi la fase de los Cn con
índice negativo
%Graficamos
figure; subplot(2,1,1); stem([-n:1:n],amplitud); title('Espectro de
Amplitud');
subplot(2,1,2); stem([-n:1:n],fase); title('Espectro de Fase');
```

b) Reconstruimos la señal para las cinco primeras armónicas

```
%Reconstruimos la señal
t=[0:.0001:.03]; % definimos nuestro vector de tiempos
clear i % eliminamos la variable i para que no se confunda con la unidad
imaginaria
for j=1:length(t)
vr(j)=sum(Cn.*exp(i*[-n:n]*2*pi*100*t(j))); %reconstruimos la señal (note el
w0 usado)
end
vo=12*sqrt(2)*abs(sin(2*pi*50.*t)); %calculamos la señal original
figure; plot(t,vr,t,vo,'r'); title('señal reconstruida (azul) vs. señal
original (rojo)');
```

c) Calculemos la potencia media de la señal realizando la integral en el tiempo. Recordemos que la señal tiene período $T=0.01$ y la potencia media de la señal se obtiene mediante

$$P_m = \frac{1}{T} \int_0^T |V_o(t)|^2 dt$$

Finalmente obtenemos la potencia contenida en las n primeras armónicas.

```
syms t; %Creación de una variable de integración de tipo simbólico
T=.01; %Definimos el período de la señal
Pm=1/T*int('(12*2^0.5*abs(sin(2*pi*50*t)))^2',t,0,T)
%La función int integra el integrando entre comillas tomando como variable de
integración t entre 0 y T
```

```
%Por el teorema de Parseval
%Pm=abs(C0)^2+2*abs(C1)^2+2*abs(C2)^2+....
C0=Cn(n+1); %Recuerde que el coeficiente C0 es la componente n+1 de Cn
Pot_n_armonicas=abs(C0)^2+2*sum(abs(Cn(1:n)).^2)
```

Comentario:

En el inciso a) los coeficientes de la serie exponencial de Fourier (línea 2 del código del inciso a)) pueden calcularse usando un lazo 'for' mediante:

```
j=0;
for k=-n:n
j=j+1;
Cn(j)=24/pi*sqrt(2)*1/(1-4*k^2);
end
```

Mientras que en b) la potencia contenida en las n armónicas también podría haberse calculado mediante:

```
C0=Cn(n+1);
Pot_n_armonicas=C0^2
for k=n:-1:1
k
Pot_n_armonicas=Pot_n_armonicas+2*abs(Cn(k))^2
end
```

donde se muestran las potencias acumuladas desde la componente de continua (armónica cero) hasta finalmente la armónica n .

Nota: la función 'abs' calcula el módulo y la función 'phase' la fase del vector de coeficientes C_n .

2) Grafique la serie de Fourier correspondiente a un tren de impulsos dada por

$$\sum_{n=-\infty}^{\infty} \delta(t - nT) = \frac{1}{T} \sum_{n=-\infty}^{\infty} e^{jn\omega_0 t}$$

para las primeras n armónicas (para $n=5, 20, 50$ y 1000).

a) Instrucciones para inicializar período, frecuencia y vector de tiempos.

```
T=1; %consideramos un período igual a uno
Wo=2*pi/T; %calculamos la frec. angular fundamental
t=[0:T/100:5*T]; %consideramos un vector de tiempos de 0 a 5T en instantes
% equiespaciados con intervalos T/100
clear i %para evitar confundir la unidad imaginaria con posible variable.
```

Comparando con el ejercicio 1, ¿se necesitan más o menos armónicas para obtener una buena aproximación de la señal? Justifique su respuesta.

.....
.....
.....

b) A continuación calculamos la suma de las primeras n armónicas para $n=5$, 20, 50 y 1000 y las graficamos.

```
n=5;
y=1/T*sum(exp(i*[-n:n]*Wo*t)); %calculamos la suma parcial de la serie para
n=5
figure;plot(t,y);title('la serie reconstruida con las primeras 5 armónicas');
pause

n=20;
y=1/T*sum(exp(i*[-n:n]*Wo*t));%calculamos la suma parcial de la serie para
n=20
figure;plot(t,y);title('la serie reconstruida con las primeras 20 armónicas');
pause

n=50;
y=1/T*sum(exp(i*[-n:n]*Wo*t));%calculamos la suma parcial de la serie para
n=50
figure;plot(t,y); title('la serie reconstruida con las primeras 50 armónicas');
pause

n=1000;
y=1/T*sum(exp(i*[-n:n]*Wo*t));; %calculamos la suma parcial de la serie para
n=1000
figure;plot(t,y); title('la serie reconstruida con las primeras 1000
armónicas');pause
```

Otra forma de calcular la serie:

```
%Las instrucciones anteriores para el cálculo de las sumas también se pueden
realizar con un lazo for
suma=0;
for k=-n:n
    suma=suma+exp(i*k*Wo*t);
end
```

Nota: la función exponencial es 'exp'.

En ambos casos observe qué ocurre, tanto con la reconstrucción de la señal como con el cálculo de la potencia, al aumentar el número de armónicos.

.....
.....
.....
.....

MATEMÁTICA AVANZADA - SEMINARIO 4

Transformada de Fourier

1) Visualice las relaciones entre el ancho de un pulso en el tiempo y los cruces por cero de su transformada de Fourier para el par dado por:

$$P_a(t) \rightarrow a \operatorname{sinc}\left(\frac{a\omega}{2}\right)$$

Solución

a) Instrucción para inicializar tiempos:

```
t=-10:.01:10;
```

b) Creamos el pulso en el tiempo:

```
ancho=1; %Inicializamos el ancho del pulso en 1
pulso = 1*(t>=-ancho/2 & t<=ancho/2); % Generamos el pulso
```

c) Creamos la transformada $a \operatorname{sinc}\left(\frac{a\omega}{2}\right)$:

```
w=-6*pi:.01:6*pi;
funcion_sinc=ancho*sin(ancho*w/2)./(ancho*w/2);
```

d) Graficamos pulso y transformada en una misma figura:

```
figure;
subplot(2,1,1); hold on; plot(t,pulso); axis([-10 10 0 2]);
subplot(2,1,2); hold on; plot(w,funcion_sinc);
```

Repita los pasos b), c) y d) cambiando el ancho del pulso modificando el valor de la variable *ancho* a un valor entre 1 y 10.

Observe los gráficos resultantes y extraiga conclusiones.

¿Qué relación observa entre el ancho del pulso en el tiempo y el ancho de los lóbulos de la sinc en el dominio transformado?

.....
.....
.....

Verifique que también es posible crear el pulso en el inciso b) empleando la función predefinida *Heaviside*.

```
pulso=heaviside(t+ancho/2*ones(size(t)))-heaviside(t-ancho/2*ones(size(t)));
%equivalente a realizar Pa(t)=u(t+a/2)-u(t-a/2) donde a es el ancho del pulso
```

2) Considere ahora el ejercicio 34 de la guía de transformada de Fourier.

Dada $f(t) = \sin(\omega_0 t) P_a(t)$, una señal limitada en el tiempo, y su correspondiente transformada de Fourier

$$F(\omega) = -\frac{ja}{2} \left\{ \text{sinc} \left[\frac{a}{2} (\omega - \omega_0) \right] - \text{sinc} \left[\frac{a}{2} (\omega + \omega_0) \right] \right\}, \text{ grafique } f(t) \text{ y el espectro de amplitud } |F(\omega)|$$

considerando $\omega_0 \gg \frac{2\pi}{a}$. (Tome por ejemplo $\omega_0=10$ y $a=5$)

Solución:

```
clear i % borra la variable i para no confundir con la unidad imaginaria
wo=10; %define la frecuencia de la senoidal que forma f(t)
a=5; %define el ancho del pulso de f(t)
t=[-1.5*a:.01:1.5*a]; %define el vector de tiempos t
w=[-2*wo:.01:2*wo]; %define el vector de frecuencias w
f=sin(wo*t).*(t>-a & t<a); %calcula la f(t)
F=abs(i/2*(a*sin(a*(w-wo)/2)./(a*(w-wo)/2)-a*sin(a*(w+wo)/2)./(a*(w+wo)/2)));
%calcula el espectro de amplitud de F(w)
figure; subplot(2,1,1); plot(t,f); subplot(2,1,2); plot(w,F); %grafica la
función correspondiente y su espectro de amplitud en un mismo gráfico
```

Repita el ejercicio al aumentar y disminuir el ancho a de la señal en el tiempo y extraiga conclusiones.

¿Qué ocurre si no se cumple la condición $\omega_0 \gg \frac{2\pi}{a}$?

.....
.....
.....

Transformada de Laplace. Variable de estado en el dominio transformado

1) La herramienta 'simulink' de Matlab permite simular y analizar modelos de diversos tipos, entre ellos, modelos en variables de estados. En primer lugar, observaremos el comportamiento de dos sistemas representados por modelos en variables de estado siguiendo las instrucciones dadas a continuación:

a) Abra la librería de simulink haciendo click en el ícono correspondiente o escribiendo simulink en el editor de comandos de MATLAB.

b) Haga click en abrir y cargue el archivo estados.mdl.

c) Haga doble click sobre cada modelo de estado y observe las matrices A, B, C, D de cada uno. ¿Cuál de ellos tiene cómo salidas a las variables de estado?

.....

d) Corra la simulación mediante el botón 'play' y haga doble click sobre los osciloscopios para ver las formas de onda. Identifique la parte transitoria y la parte permanente en la señal de salida.

¿Cuánto dura el transitorio aproximadamente?.....

2) Utilizando MATLAB vamos a obtener la respuesta analítica de un sistema representado por un modelo en variables de estado:

a) Ingrese las matrices $A = [-1 \ 0 \ 0; 1 \ -2 \ 0; 2 \ 2 \ -3]$; $B = [1; 0; 0]$; $C = [0 \ 0 \ 1]$; $D = 1$;

b) Obtenga la matriz de transición de los estados mediante las siguientes instrucciones:

```
>> syms s % Definimos la variable simbólica s.
>> I=eye(3); % Matriz identidad de 3x3
>> fis=inv(s*I-A) % Obtenemos transformada de la matriz de transición del
modelo.
>> fit=ilaplace(fis) % Matriz de transición.
```

Suponiendo que en $t=0$ los estados son $X_0 = [0;1;1]$, calcule la respuesta libre de los estados. Esto puede hacerse en el dominio temporal o en el dominio transformado.

```
% Cálculo en el dominio temporal
>> xt=fit*x0

% Cálculo en el dominio transformado
>> xs=fis*x0
>> xt=ilaplace(xs)
```

¿Las respuestas obtenidas por ambos caminos son iguales?

.....

¿Cómo calcularía la solución libre del sistema? Escríbala.

.....
.....
.....
.....