Práctica 4 – Circuitos RC

Jorge Daniel Carreón Guzmán
19120266
Ing. Mecatrónica
Morelia, México
L19120966@morelia.tecnm.mx

***Resumen*—En esta práctica se analizan circuitos en serie y paralelo con resistencias y capacitores de diferentes valores, además se hace un repaso de conceptos básicos de circuitos eléctricos**

Palabras clave — Matlab, Transformada de Laplace, ecuaciones diferenciales, variable

I. Introducción

En la presente práctica lo que se intenta es hacer una ligera introducción en el software de Matlab, utilizado lo ya visto en clase como las ecuaciones diferenciales ordinarias, lineales de primer y segundo orden con condiciones iniciales. Todo esto para adentrarnos y conocer el entorno de dicho orden. A su vez realizaremos todo lo necesario para que el software pueda resolver las ecuaciones; cosas como crear variables, graficar datos, graficar funciones, entre otras.

II. Desarrollo

Primera parte.

a) Conocer el entorno de Matlab

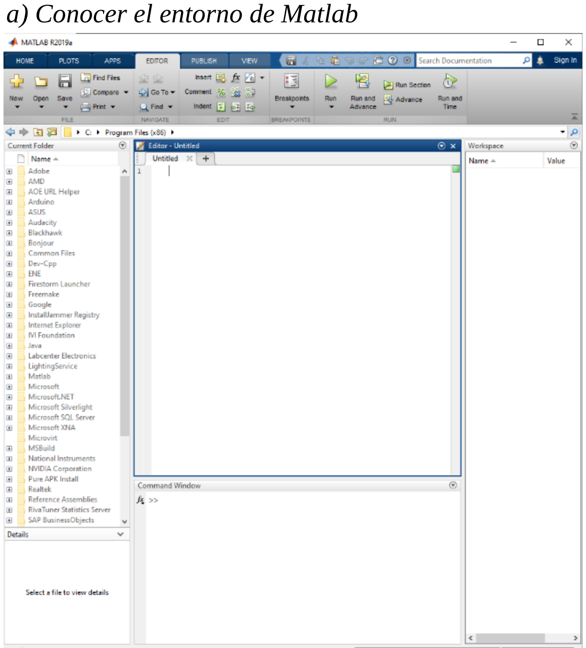

Figura 1. Entorno gráfico de Matlab

En la figura 1 podemos ver el entorno gráfico de Matlab, el cual se divide en 4 ventanas principales:

Figura 2. Command Window

Command Window es la ventana más importante dentro de Matlab, en ella, el usuario interactuará con el programa. Podrá poner comandos, como en la figura 2, una sencilla suma de números y el software regresará el resultado de dichos comandos.

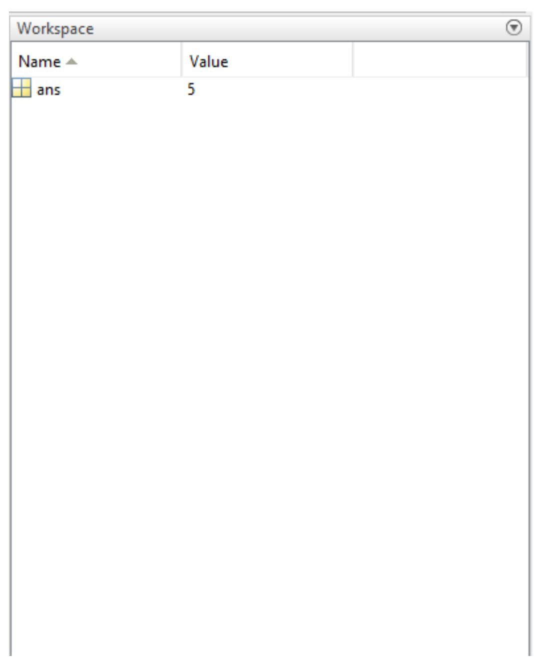

Figura 3. Workspace

La figura 3, es el espacio de trabajo, en el cual se almacenarán las variables que son ocupadas por Matlab para realzar dichas operaciones. En este caso podemos ver que existe una variable que se llama ans, con valor de 5.

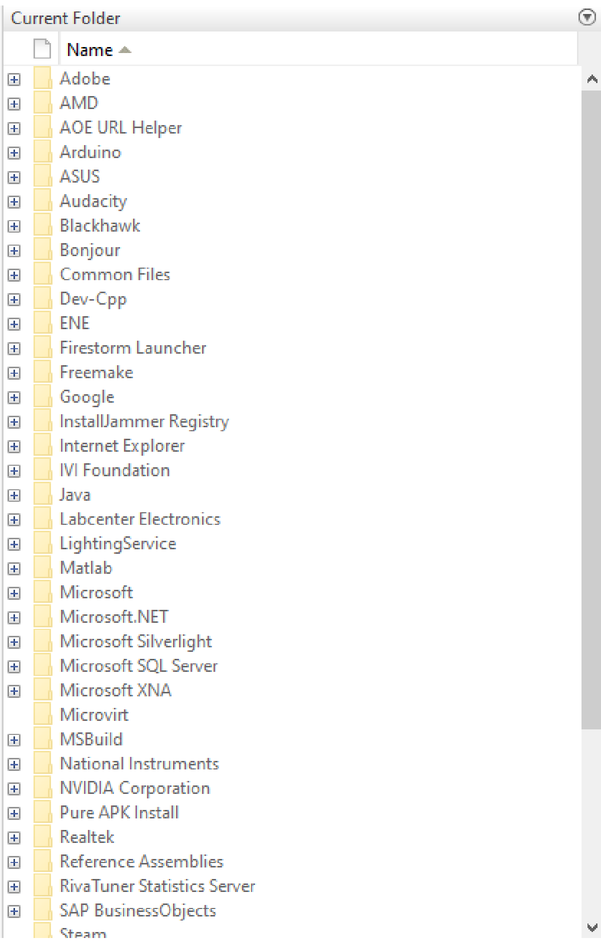

Figura 4. Current Folder

El Current Folder tiene varios trabajos. Digamos que trabaja como un explorador de Windows, entonces podremos cambiar el nombre o borrar ciertos archivos. Lo más importante es que en esta ventana tenemos acceso a todos los archivos que tiene Matlab al momento. Esto es muy importante debido a que Matlab es capaz de leer archivos de Excel, Word y hasta imágenes. Si se le hace un llamado a cierto archivo y no se encuentra en esta ventana, nos mostrará error. Hay que tener cuidado con esto para poder explotar al máximo las capacidades de Matlab.

Estas son las ventanas más importantes. Las demás las iremos conociendo mejor durante el semestre y mientras avance nuestro uso con el software.

b) Graficar datos

Pueden generarse usando el comando plot. Esta gráfica consiste de puntos conectados por líneas rectas. Los valores de la variable independiente y dependiente son dados mediante dos vectores del mismo tamaño

c) Crear variables

Para crear una variable nueva, se debe introducir el nombre de la variable en la ventana de comandos, seguido por un signo igual (=) y el valor que desea asignar a la variable.

d) Graficar funciones

Las graficas de funciones pueden generarse con el comando plot. A diferencia de los datos, estás deben tener variables asignadas para X y Y, junto con los rangos de cada una.

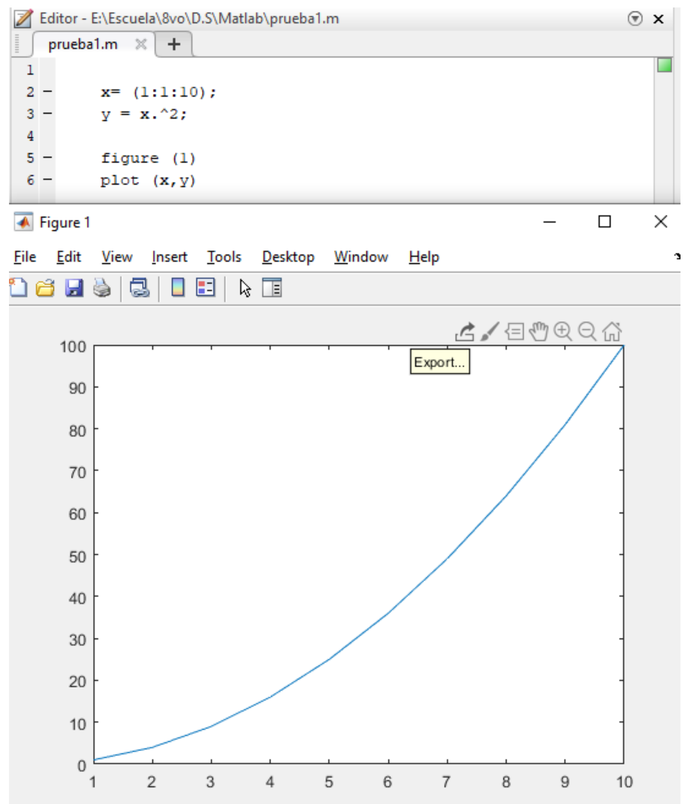

Figura 5. Graficar funciones

En x, los números que aparecen son, el valor inicial, seguido de dos puntos, luego el valor del incremento y hasta donde queremos que llegue.

En y, creamos la función. El punto que existe dentro del editor, sirve para diferenciarle a Matlab que queremos elevar al cuadrado cada elemento de x, puesto que trabaja con operaciones de matrices.

Una vez creada la función, creamos la figura, esto no es necesario, pero es útil para una mayor organización.

Finalmente usamos el comando ya mencionado, usando las variables x, y ya antes propuestas.

e) Crear variables simbólicas

Para crear estas variables, se utiliza el comando “syms”

f) Resolver funciones simbólicas

Para resolver las funciones simbólicas, simplemente es una operación que involucra una o más variables simbólicas, que son aquellas que representan una incógnita que puede tomar cualquier valor. Por tanto, no devuelve un valor numérico, sino una expresión simbólica. Veremos con un ejemplo esto más adelante.

g) Crear ecuaciones diferenciales

Una vez asignada la función syms, podemos asignar las variables dependientes e independientes. En este caso t (tiempo) sería la variable independiente. Una ecuación diferencial es una ecuación que involucra a las derivadas de una función con la propia función y/o las variables de las que depende. Para crear la derivada usaremos el comando “diff”:

```
clear all; clc;
syms y(t) s
dly=diff(y)
eqn = dly + (3*y) == 13 *(sin(2*t))
```

Figura 6. Ecuación diferencial

h) Resolver ecuaciones diferenciales

Simplemente será el comando “solve” junto con el nombre de la ecuación.

i) Conocer los comandos para realizar la transformada de Laplace y su inversa

Sus comandos son “Laplace” e “ilaplace” respectivamente antes de la función.

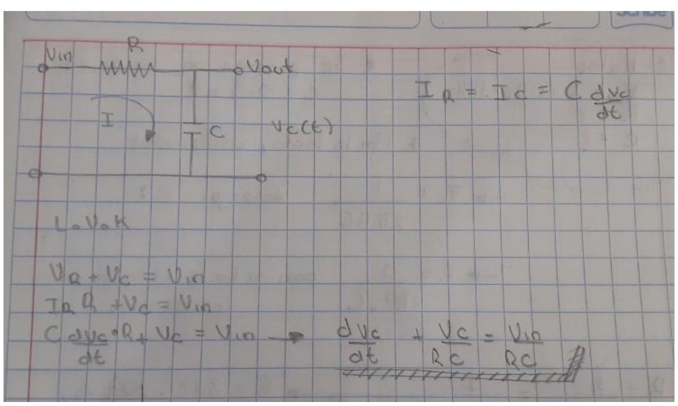

Figura 2. Desarrollo teórico de a)

j) Encontrarla transformada de Laplace de las funciones simbólicas:

1) $f(t) = e^{-3t}$ 2) $f(t) = \cos(4t)$ 3) $f(t) = e^{-t} sen(5t)$

Figura 7. funciones simbólicas

```
%% 1)
clear all
close all
syms y(t) s

eqn = exp(-3*t)
laplace (eqn,t,s)

%% 2)
clear all
close all

syms t s
f= cos(4*t)
laplace (f,t,s)

%% 3)

clear all
close all

syms y(t) s
f = exp(-t)* sin(5*t)
laplace (f,t,s)
```

Figura 8. Planteamiento en el editor

```
>> prueba1

eqn =

exp(-3*t)


ans =

1/(s + 3)


f =

cos(4*t)


ans =

s/(s^2 + 16)


f =

sin(5*t)*exp(-t)


ans =

5/((s + 1)^2 + 25)
```

Figura 9. Transformada de Laplace de funciones simbólicas propuestas

k) Crear ecuaciones diferenciales

$y'' - \frac{3}{2}y' - y = 0;$ $y(0) = 0, \quad y'(0) = \frac{5}{2}$	$y'' - y' - 6y = 0;$ $y(0) = 1, y'(0) = -1$	$y' + 3y = 13sen(2t);\ y(0) = 6$

Figura 10. Ecuaciones diferenciales a resolver en Matlab

En esta parte, para evitar volver una y otra vez a las mismas ecuaciones, decidí incluir el paso l y m dentro de este. Es decir, resolviendo las ecuaciones mediante Laplace y graficando a su vez.

```
clear all; clc;
syms y(t) s
d1y=diff(y)
d2y=diff(y,t,t)
eqn = d2y - ((3/2)*d1y) - y == 0
Fs=laplace(eqn,t,s)
syms y_LT dy_LT
Fs = subs(Fs, [laplace(y(t),t,s) subs(diff(y(t),t),t,0)],[y_LT dy_LT])
Vars = [y(0) dy_LT];
Values =[0 (5/2)];
Fs1 = subs(Fs, Vars, Values)
y_LT= solve(Fs1)
y_sol = ilaplace(y_LT)
%comp
YS1 = y_sol
D1YS1 = diff(YS1)
D2YS1 = diff(YS1,t,t)
Comp= D2YS1 - ((3/2)*D1YS1) - YS1
%grafica
fplot(YS1,[-20 20])
hold on
fplot(D1YS1,[-20 20])
hold on
fplot(D2YS1,[-20 20])
legend("original","primer derivada")
grid on
```

Figura 11. Comandos Ec. 1

```
d1y(t) =
diff(y(t), t)

d2y(t) =
diff(y(t), t, t)

eqn(t) =
diff(y(t), t, t) - (3*diff(y(t), t))/2 - y(t) == 0

Fs =
(3*y(0))/2 - (3*s*laplace(y(t), t, s))/2 - s*y(0) + s^2*laplace(y(t), t, s) - subs(diff(y(t), t), t, 0) - laplace(y(t), t, s) == 0

Fs =
(3*y(0))/2 - y_LT - dy_LT - s*y(0) - (3*s*y_LT)/2 + s^2*y_LT == 0

Fs1 =
y_LT*s^2 - (3*y_LT*s)/2 - y_LT - 5/2 == 0

y_LT =
-5/(- 2*s^2 + 3*s + 2)

y_sol =
exp(2*t) - exp(-t/2)

YS1 =
exp(2*t) - exp(-t/2)

D1YS1 =

2*exp(2*t) + exp(-t/2)/2


D2YS1 =

4*exp(2*t) - exp(-t/2)/4


Comp =

0

>>
```

Figura 12. Resultado command window Ec. 1

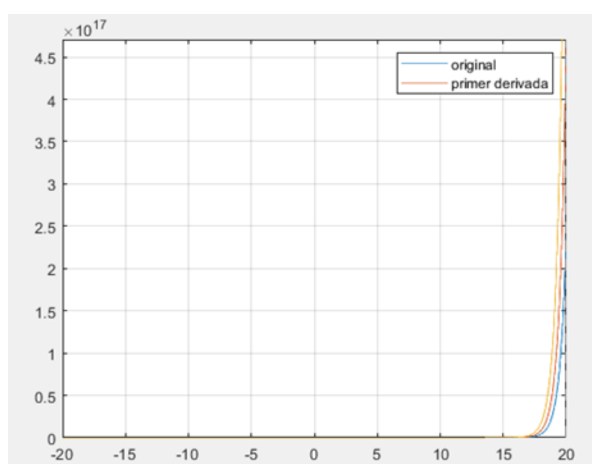

Figura 13. Gráfica Ec. 1

```
clear all; clc;
syms y(t) s
d1y=diff(y)
d2y=diff(y,t,t)
eqn = d2y - d1y - ((6)*y) == 0
Fs=laplace(eqn,t,s)
syms y_LT dy_LT
Fs = subs(Fs, [laplace(y(t),t,s) subs(diff(y(t),t),t,0)],[y_LT dy_LT])
Vars = [y(0) dy_LT];
Values =[1 -1];
Fs1 = subs(Fs, Vars, Values)
y_LT= solve(Fs1)
y_sol = ilaplace(y_LT)
%comp
YS1 = y_sol
D1YS1 = diff(YS1)
D2YS1 = diff(YS1,t,t)
Comp= D2YS1 - D1YS1 - ((6)*YS1)
%grafica
fplot(YS1,[-20 20])
hold on
fplot(D1YS1,[-20 20])
hold on
fplot(D2YS1,[-20 20])
legend("original","primer derivada")
grid on
```

Figura 14. Comandos Ec. 2

```
d1y(t) =

diff(y(t), t)


d2y(t) =

diff(y(t), t, t)


eqn(t) =

diff(y(t), t, t) - diff(y(t), t) - 6*y(t) == 0


Fs =

y(0) - s*laplace(y(t), t, s) - s*y(0) + s^2*laplace(y(t), t, s) - subs(diff(y(t), t), t, 0) - 6*laplace(y(t), t, s) == 0


Fs =

y(0) - 6*y_LT - dy_LT - s*y(0) - s*y_LT + s^2*y_LT == 0


Fs1 =

s^2*y_LT - 6*y_LT - s*y_LT - s + 2 == 0


y_LT =

-(s - 2)/(- s^2 + s + 6)


y_sol =

(4*exp(-2*t))/5 + exp(3*t)/5

YS1 =

(4*exp(-2*t))/5 + exp(3*t)/5


D1YS1 =

(3*exp(3*t))/5 - (8*exp(-2*t))/5


D2YS1 =

(16*exp(-2*t))/5 + (9*exp(3*t))/5


Comp =

0

>> |
```

Figura 15. Resultado command window. Ec. 2

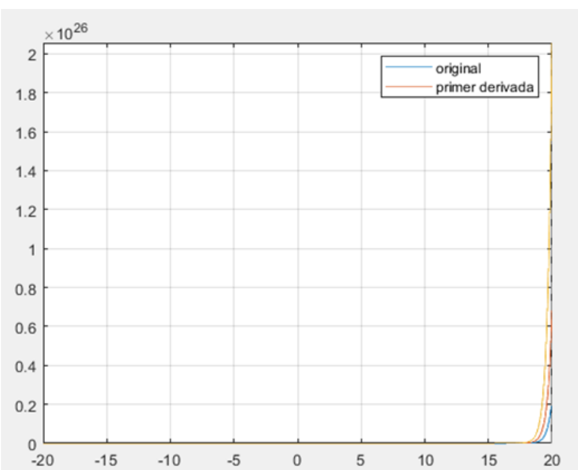

Figura 16. Gráfica Ec. 2

```
clear all; clc;
syms y(t) s
d1y=diff(y)
eqn = d1y + (3*y) == 13 *(sin(2*t))
Fs=laplace(eqn,t,s)
syms y_LT dy_LT
Fs = subs(Fs, [laplace(y(t),t,s) subs(diff(y(t),t),t,0)],[y_LT dy_LT])
Vars = [y(0)];
Values =[6];
Fs1 = subs(Fs, Vars, Values)
y_LT= solve(Fs1)
y_sol = ilaplace(y_LT)

%comprobasion
YS1 = y_sol
D1YS1 = diff(YS1)
Comp=D1YS1 +((3)*YS1)

%graficar
fplot(YS1,[-5 5])
hold on
fplot(D1YS1,[-5 5])
hold on
legend("original","primer derivada")
grid on
```

Figura 17. Comandos Ec. 3

```
d1y(t) =

diff(y(t), t)


eqn(t) =

3*y(t) + diff(y(t), t) == 13*sin(2*t)


Fs =

s*laplace(y(t), t, s) - y(0) + 3*laplace(y(t), t, s) == 26/(s^2 + 4)


Fs =

3*y_LT - y(0) + s*y_LT == 26/(s^2 + 4)


Fs1 =

3*y_LT + s*y_LT - 6 == 26/(s^2 + 4)


y_LT =

(26/(s^2 + 4) + 6)/(s + 3)


y_sol =

8*exp(-3*t) - 2*cos(2*t) + 3*sin(2*t)


YS1 =

8*exp(-3*t) - 2*cos(2*t) + 3*sin(2*t)


D1YS1 =

6*cos(2*t) - 24*exp(-3*t) + 4*sin(2*t)
```

Figura 18. Resultado command window Ec. 3

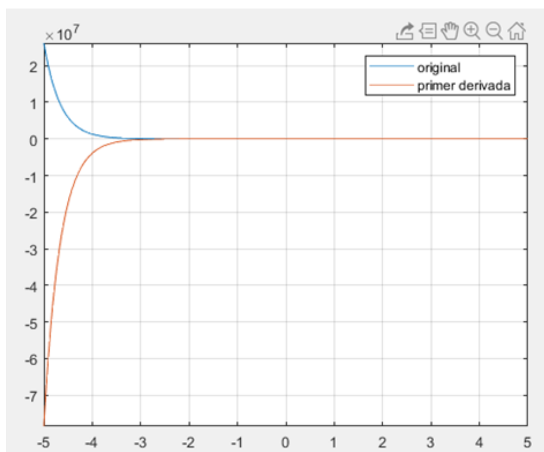

Figura 19. Gráfica Ec. 3

Segunda parte.

a) *Comprobar a mano los resultados de las soluciones de ecuaciones diferenciales mediante la transformada de Laplace*

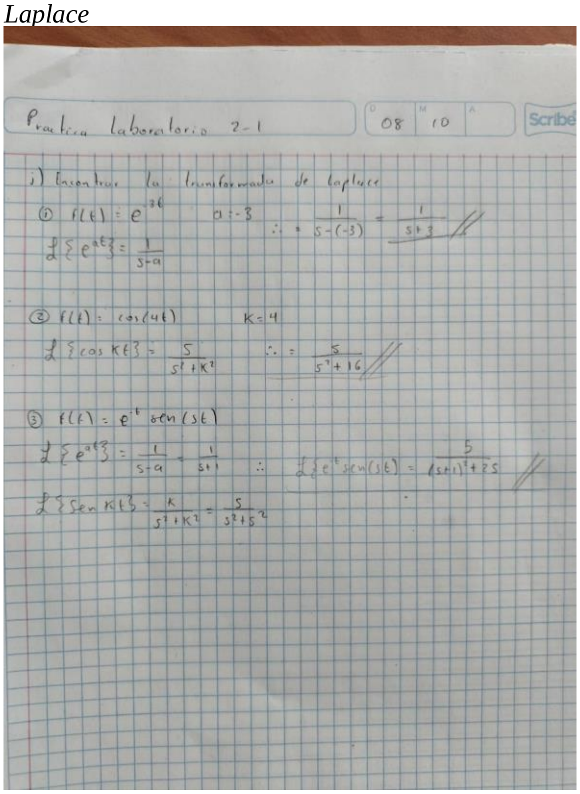

Figura 20. Ejercicios a mano (1-3) parte j)

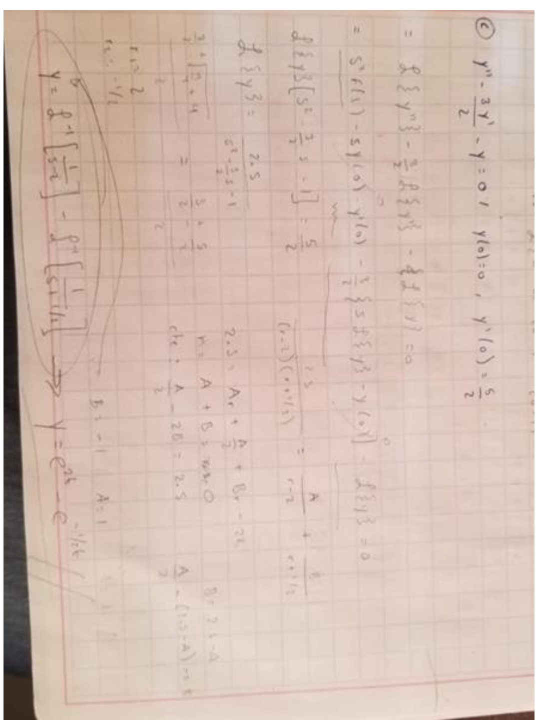

Figura 21. Ejercicioa a mano k,1

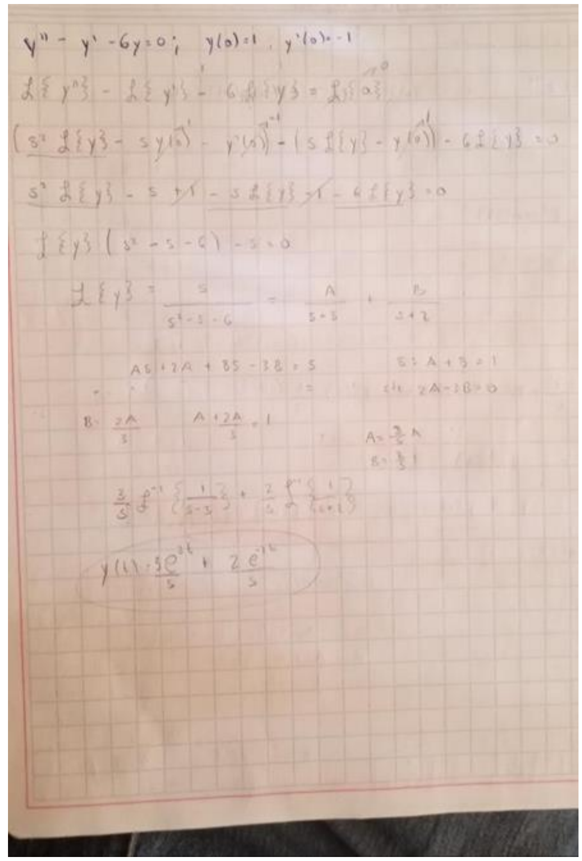

Figura 22. Ejercicio a mano k,2

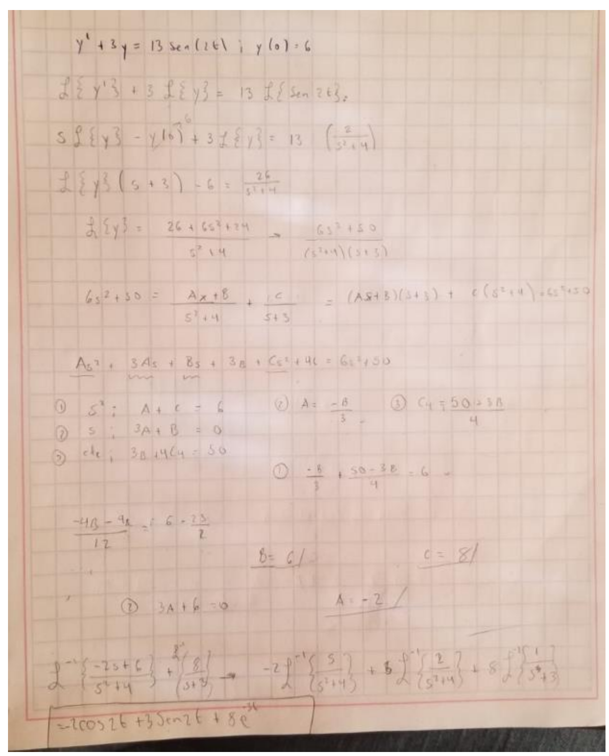

Figura 23. Ejercicio a mano k,3

Tercera parte.

a) ¿Qué es y para qué sirve Matlab? [1]

MATLAB ® es una plataforma de programación diseñado específicamente para los ingenieros y científicos, para analizar y diseñar sistemas y productos que transforman nuestro mundo. MATLAB es un lenguaje basado en matrices que permite la expresión más natural de las matemáticas computacionales.

b) ¿Qué significa variable simbólica? [2]

Una variable simbólica es un objeto en Matlab que representa una incógnita que puede tomar cualquier valor. Por tanto, una operación que involucra una o más variables simbólicas no devuelve un valor numérico, sino una expresión simbólica que involucra números, operaciones, funciones y variables simbólicas.

c) ¿Cuáles son y como se utilizan los comandos para la transformada de la place y su inversa?

Se usan como <<Laplace>> e <<ilaplace>> respectivamente en el comand window. Seguido abriremos unos paréntesis, donde irá la variable de la función, la variable dependiente (generalmente es el tiempo), y a la que queremos transformar (que tiende a ser la "s")

d) ¿Qué es y para qué sirve el comando "diff"? [3]

El comando diff(S,var) se utiliza para calcular Ia derivada de una expresión con mas de una variable simbólica. Las segundas derivadas (y otras de mayor orden) se pueden calcular mediante las sintaxis diff(S,n) o diff(S,var,n), donde n es un número positivo.

III. CONCLUSIONES

La práctica me pareció algo pesada. Fueron muchos cálculos y mucha información en tan poco tiempo que resulta difícil concretarla. Ahora que se pudo comprobar el resultado de los cálculos analíticos en libreta con MATLAB, me siento satisfecho. La transformada de Laplace nos da mucha ventaja a la hora de realizar cálculos y más utilizando un software como MATLAB. Ahora podemos resolver ecuaciones en minutos, que en el pasado nos tomaría quizá horas.

IV. REFERENCIAS

[1] Qué es MATLAB? - MATLAB y Simulink - Componentes Electrónicas. (2019, diciembre 19). Componentes Electrónicas LTDA. https://www.compelect.com.co/que-es-matlab/

[2] Cálculo simbólico — Laboratorio de Matemáticas 2010/2011. (s/f). Uam.es. Recuperado el 8 de octubre de 2022,de http://verso.mat.uam.es/~pablo.angulo/doc/laboratorio/b5s1.html

[3] Ali, A. (2022, enero 20). La función diff() en MATLAB. Delft Stack. https://www.delftstack.com/es/howto/matlab/matlab-diff-function/