|  |  |  |
| --- | --- | --- |
| Facultad de Ingeniería del Ejército – Grl Div Manuel N. Savio | Universidad de la Defensa Nacional  Facultad de Ingeniería del Ejército | |
| ***Curso de MATLAB/Octave***  *Entorno de programación* | | |
|  | | |
| Ing. Gianfranco Salomone  **Año 2020** | |  |

Tabla de contenido

[1. Enfoque 3](#_Toc41330808)

[2. Matlab vs Octave 3](#_Toc41330809)

[2.1. Instalación de octave 3](#_Toc41330810)

[2.2. Source forge e instalación de paquetes en Octave 3](#_Toc41330811)

[2.3. Cargar paquete al inicio de octave 4](#_Toc41330812)

[2.4. Instalación de MATLAB (durante el periodo de aislamiento obligatorio) 4](#_Toc41330813)

[2.5. Versiones ONLINE 5](#_Toc41330814)

[3. Entorno de programación 5](#_Toc41330815)

[3.1. Operaciones básicas 5](#_Toc41330816)

[3.2. Arreglos y matrices 7](#_Toc41330817)

[3.3. Operaciones lógicas y bucles 11](#_Toc41330818)

[3.4. Gráficas 11](#_Toc41330819)

[3.5. Meshgrid 14](#_Toc41330820)

[3.6. Sistemas de ecuaciones lineales 15](#_Toc41330821)

[3.7. La paradoja de la derivada 16](#_Toc41330822)

# Enfoque

A lo largo del curso se presentarán problemas y sus soluciones analíticas; la intención del curso consiste en ser capaces de representar información conocida, de diferentes maneras. A medida que avancemos con el curso, la cantidad de instrucciones será mayor, y de mayor complejidad.

Durante el presente curso utilizaré para las demostraciones el software MATLAB R2020a. Sin embargo, lo haré de manera que el código sea accesible para quien desee utilizar Octave.

Tengamos en cuenta que, en general, los programas de código abierto “se atrasan” con respecto a los comerciales. Ello implica que la versión más actual de Octave cuenta con herramientas que tal vez son obsoletas hoy en MATLAB. A pesar de ello, a los fines prácticos de la mayoría de las aplicaciones, Octave es una herramienta más que adecuada.

# Matlab vs Octave

Octave es libre. Mayor transparencia en la generación de código. Mayor reproducibilidad. SIN GARANTÍA. Opera de manera nativa en Linux.

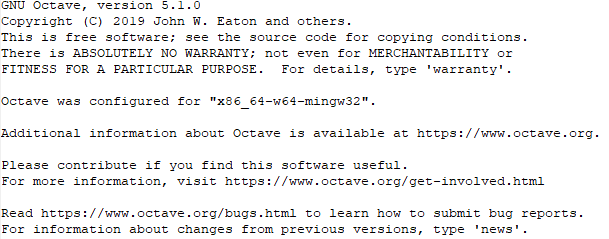


Figura 1: licencia de Octave

MATLAB da garantía, tiene entorno gráfico más amigable, es bastante más rápido para acciones iterativas (ver ejemplo de carga sumergida en campo magnético).

## Instalación de octave

Acceder a <https://www.gnu.org/software/octave/> , y buscar el instalador para el sistema operativo propio.

Para agregar funcionalidades nuevas al entorno numérico básico, existen los “packages”.

## Source forge e instalación de paquetes en Octave

***Instalación online***

Instalación de la última versión del paquete:

pkg install -forge arduino

pkg install -forge control

***Instalación offline (con el archivo [paquete].tar.gz) descargado previamente de*** <https://octave.sourceforge.io/packages.php>

Cambiar la dirección actual a la dirección en la que se encuentra el paquete:



Luego: ejecutar

pkg install [PAQUETE].tar.gz

En nuestro caso, realizaremos la Opción 1 de instalación del paquete de ***matemática simbólica***, sugerida en <https://github.com/cbm755/octsympy/wiki/Notes-on-Windows-installation>

* La instalación sólo se realiza una vez.
* Cada vez que se desea usar un paquete, se deberá cargar en el espacio de trabajo con pkg load [PAQUETE]

## Cargar paquete al inicio de octave

En la dirección (por defecto) **C:\Octave\Octave-5.1.0.0\mingw64\share\octave\site\m\startup**

Buscar el archivo ***octaverc,*** y editarlo con el block de notas:

|  |
| --- |
| ## System-wide startup file for Octave.  ##  ## This file should contain any commands that should be executed each  ## time Octave starts for every user at this site.  EXEC\_PATH (cstrcat (fullfile (OCTAVE\_HOME, 'notepad++'), pathsep, EXEC\_PATH));  EXEC\_PATH (cstrcat (fullfile (OCTAVE\_HOME, 'bin'), pathsep, EXEC\_PATH));  EDITOR (fullfile (OCTAVE\_HOME, 'notepad++', 'notepad++.exe'));  clc; % By Gian 2019 10 24  pkg load control; % By Gian 2019 10 24  pkg load arduino; % By Gian 2019 10 28 |

**ATENCIÓN**

Si la intención es compartir los archivos .m, debemos tener en cuenta que quien los recibe puede no tener los mismos paquetes instalados. Una buena manera de recordarlo puede ser la de iniciar todos los programas con

pkg load [paquete]

## Instalación de MATLAB (durante el periodo de aislamiento obligatorio)

La Facultad de Ingeniería del Ejército cuenta con licencias educativas en su laboratorio de simulación. Dada la situación actual de aislamiento, MATLAB ha habilitado la descarga y licencia educativa para uso particular en el siguiente link (requiere crear una cuenta en MathWorks)

<https://www.mathworks.com/licensecenter/classroom/COVID-19_Access/>

Durante el presente curso utilizaremos únicamente el entorno básico de MATLAB y el toolbox de matemática simbólica, que puede descargarse de allí.

## Versiones online

Es posible acceder a versiones online tanto de MATLAB como de Octave:

<https://matlab.mathworks.com/>

<https://octave-online.net/>

En este curso nos orientaremos al uso de software descargado en la PC.

# Entorno de programación

Varía entre versiones de los programas, pero en general cuenta con las siguientes ventanas:

***Command Window (ventana de comandos)***

En esta ventana, tanto en MATLAB como en Octave, podremos realizar cualquier operación, en una línea de comandos, identificada por los símbolos de prompt: “ >> “

***Current folder / Current directory (carpeta/directorio actual)***

Está indicado como una dirección (C:\...), y es la carpeta desde se ejecutarán las funciones definidas por el usuario.

***Workspace (espacio de trabajo)***

Aquí se muestran las variables en uso, su tipo y su valor.

***Editor***

Es el entorno en que se escribirá el programa que realizará, una a una, y en orden, las operaciones deseadas.

## Operaciones básicas

Para empezar, realizaremos las operaciones básicas en la ventana de comandos.

Una práctica interactiva puede encontrarse en la página de MATLAB (requiere crear una cuenta):

<https://la.mathworks.com/learn/tutorials/matlab-onramp.html?s_cid=learn_ONRAMP_BAN>

***Suma y producto***

Es posible utilizar el programa como una simple calculadora:

>> 3 + 7

ans = 10

>> 12\*10

ans = 10

Nótese que en la ventana *Workspace* se creó una variable llamada ans, que se estableció en el valor 120.

El operador “ **;** “ (punto y coma) evita que se muestre el resultado de la operación en la ventana de comandos. Ejecute la siguiente instrucción, y observe lo que ocurre con la variable ans.

>> 4\*7;

***Trigonométricas***

Las funciones trigonométricas están definidas en radianes: sin, cos, tan, asin, acos, atan.

>> sin(30)

ans = -0.98803

>> sin(pi/6)

ans = 0.50000

Aunque también existen las funciones sind, cosd y tand, definidas en grados

>> cosd(60)

ans = 0.50000

***Potenciación y raíz cuadrada***

>> 3^3

ans = 27

>> sqrt(81)

ans = 9

***Notación científica***

>> 3e2

ans = 300

>> 3\*10^2

ans = 300

***Constantes definidas por defecto***

|  |  |
| --- | --- |
| pi |  |
| i o j | unidad imaginaria |
| eps | Precisión de operaciones de punto flotante |
| inf | Infinito |
| NaN | Not a Number |

***Formato de números***

Para expresar los valores numéricos, puede utilizarse la función format <*FORMATO>*

Ejemplos con el valor 1000/3 = 333.3333…

|  |  |  |
| --- | --- | --- |
| Argumento | Valor expresado | Comentario (MATLAB / Octave) |
| short | 333.3333 / 333.33 | 4 decimales / 5 cifras significativas |
| long | 3.333333333333333e+02 / 333.3333333333333 | 16 decimales / 16 cifras significativas |
| short e | 3.3333e+02 | Notación científica de short |
| long e | 3.333333333333333e+02 | Notación científica de long |
| bank | 333.33 | Formato bancario, 2 decimales |
| short eng | 333.3333e+000 | Formato de ingeniería short |
| long Eng | 333.333333333333e+000 /  333.333333333333314e+000 | Formato de ingeniería long |

***Redondeo***

Redondear al entero más cercano

>> round(pi)

ans = 3

Si deseamos un número redondeado a dos decimales:

>> round(pi\*100)/100

ans = 3.1400

En MATLAB, puede utilizarse la sintaxis round(X,N): redondear a N dígitos.

***Documentación***

Octave:

<https://octave.sourceforge.io/docs.php>

MATLAB

<https://la.mathworks.com/help/index.html>

Acceso rápido en la librería pre-instalada:

>> help *FUNCIÓN*

## Arreglos y matrices

Variable: contiene un dato que puede cambiar. No necesitan ser declaradas previamente.

>> a = sin(pi/6)

a = 0.50000

***Arreglo unidimensional (o array)***

Conjunto de elementos, cada uno identificado por un número de índice (index number). Algunas veces se denomina *vector* (como en Física), o *n-upla*. En el caso de expresar un vector Físico, deberá realizarse en forma de terna ordenada.

Generación de un arreglo cualquiera:

>> A = [-10 12 -25.3 47]; % Genera un arreglo “fila”

>> B = [-10;12;-25.3;47]; % Genera un arreglo “columna”

>> A(2)

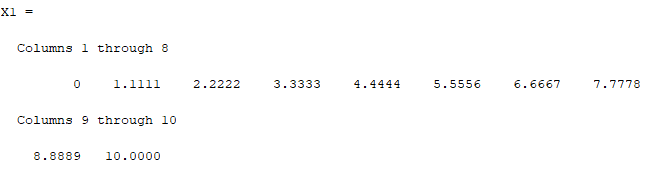
ans = 12

|  |  |
| --- | --- |
|  |  |
| Figura : generación de arreglos | |

El acceso al segundo elemento, se realiza escribiendo el número de elemento entre paréntesis.

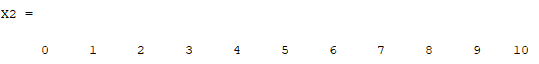
*Generación de arreglos unidimensionales igualmente espaciados:*

>> X1 = linspace(0,10,10)



Genera un arreglo que empieza en 0, termina en 10, y tiene 10 elementos (incrementos de 1.1111 en este caso)

>> X2 = 0 : 1 : 10;



Genera un arreglo que empieza en 0, termina en 10, y tiene incrementos de 1 unidad (en este caso tiene 11 elementos)

*Acceso a los últimos elementos del arreglo:*

>> X1(end) % Último elemento

ans = 10

>> X1(end-1) % Penúltimo elemento

ans = 8.8889

*(¿Por qué no es 9?)*

***Matrices (o arreglos de dos dimensiones)***

Arreglo cuyos elementos son descriptos por un par ordenado (fila,columna).

>> A = [1 0 3 ; 12 -1 7.8]

A =

1.00000 0.00000 3.00000

12.00000 -1.00000 7.80000

Para acceder a un elemento, podemos utilizar la notación de fila y columna:

>> A(2,1)

ans = 12

|  |
| --- |
|  |
| Figura : generación de una matriz. Cada elemento tiene su posición definida por el par (fila,columna) |

O bien considerar que cada elemento tiene un número de orden, de arriba hacia abajo, de izquierda a derecha:

|  |
| --- |
|  |
| Figura : matriz con número de elementos |

Siendo así, el comando >>A(4) debería dar como resultado -1.

*Acceso al último elemento de la columna 2*

>> A(end,2)

ans = -1

*Acceso a todos los elementos de la fila 2*

>> A(2,:)

ans =

12.0000 -1.0000 7.8000

*Acceso a todos los elementos de la fila 2, desde la columna 2*

>> A(2,2:end)

ans =

-1.0000 7.8000

*Transponer matriz*

>> A = A’

A =

1.00000 12.00000

0.00000 -1.00000

3.00000 7.80000

***Indexación lógica, y la función find***

>> R = rand(5) % Devuelve matriz 5x5 de números rand entre 0 y 1

R =

0.173852 0.057379 0.864244 0.951889 0.719982

0.628604 0.886837 0.878686 0.210134 0.505841

0.756125 0.047174 0.545495 0.972152 0.347260

0.714363 0.175233 0.783370 0.780233 0.801506

0.160450 0.881472 0.279142 0.479073 0.291683

>> R(R>0.5) = 87 % A todos los elementos >0.5, se les asigna valor 87

R =

0.173852 0.057379 87.000000 87.000000 87.000000

87.000000 87.000000 87.000000 0.210134 87.000000

87.000000 0.047174 87.000000 87.000000 0.347260

87.000000 0.175233 87.000000 87.000000 87.000000

0.160450 87.000000 0.279142 0.479073 0.291683

***ATENCIÓN:*** al realizar comparaciones “==” entre números de punto flotante

Instrucción *find* devuelve los ***números de índice*** para los que se cumple la condición especificada

>> t = 0:.5:2\*pi;

>> x = sin(t);

>> find(x<0)

ans =

8 9 10 11 12 13

>> x(ans)

ans =

-0.35078 -0.75680 -0.97753 -0.95892 -0.70554 -0.27942

***Creación de matrices interesantes***

Matriz de unos: ones

Matriz de ceros: zeros

Matriz identidad: eye

## Operaciones lógicas y bucles

Los operadores lógicos (if, &&, ||) y bucles (for, while) funcionan de igual modo que en lenguaje C. Ejemplos de ellas iremos viendo a medida que avance el curso.

## Gráficas

Para realizar una serie de operaciones en secuencia, y sobre todo si la serie de operaciones está en cantidad creciente, se utilizan “Scripts”, que tienen extensión .m. Para crear un nuevo script, en MATLAB se hace click en **Home\New Script**, y en Octave se sigue la ruta **File\New\New Script**. En ambos casos, puede utilizarse el acceso directo **ctrl + N**.

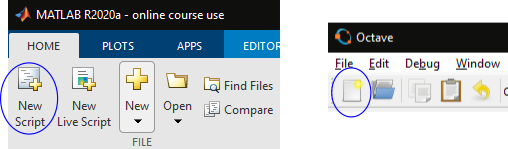


Figura 5: generación de un script

***Actividad 1:*** generar la gráfica de las funciones , y entre y

***Actividad 2:*** para la gráficas del seno, generar una figura independiente, especificar el tamaño de letra de los ejes, los nombres de las etiquetas del eje x, y las ubicaciones de las mismas. (Esta acción no es necesaria para el uso básico del software, pero es útil cuando se desea generar imágenes para publicación)



|  |
| --- |
| % Definir valor de tf (tiempo final de simulación  figure(1)  plot(t,x)    % Nombre de los ejes y título  ylabel('x (m)')  xlabel('t (s)')  title('Gráfica de la función seno')    % Nombre de los ejes y título  xlim([0 6\*pi]);  xticks([0:pi:tf])  xticklabels({'0','\pi','2\pi','3\pi','4\pi','5\pi','6\pi'})    ax = gca; % Handle de "current axes"  set(ax,'FontSize',12); |

***Actividad 3:*** realizar cada una de las gráficas de la *Actividad 1* en diferentes ejes, dentro de la misma figura. *Función* subplot

***Actividad 4:*** realizar la gráfica de una función de la forma

## Sistemas de ecuaciones lineales

El sistema de ecuaciones lineales

Puede escribirse como

Para despejar, debemos *pre-multiplicar* por la matriz inversa, miembro a miembro:

Esta operación se realiza con el símbolo “ \ “

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

>> x = A\B

x =

2

1

*Puede utilizarse también la función* linsolveo bien multiplicar por la inversa x = A^-1\*B

## La paradoja de la derivada

Basado en <https://www.youtube.com/watch?v=9vKqVkMQHKk&t=563s>

Introducción a Live Script (.mlx)

Supongamos que la posición de un móvil puede describirse según

La velocidad media entre dos instantes de tiempo puede describirse como

¿Qué significaría la velocidad “instantánea”? Este término no tiene mucho sentido, porque para determinar la velocidad, se necesitan DOS instantes de tiempo. En este sentido, a medida que , la velocidad media es cada vez una mejor aproximación al comportamiento del vehículo. En tal caso:

Supongamos que . Este es un valor finito. Luego, reemplazando en el caso actual:

Como vemos, si fuera menor, tanto más se aproximaría la velocidad a al valor

***Tarea:*** determinar aproximaciones para velocidad de un móvil, a medida que se hace más pequeño

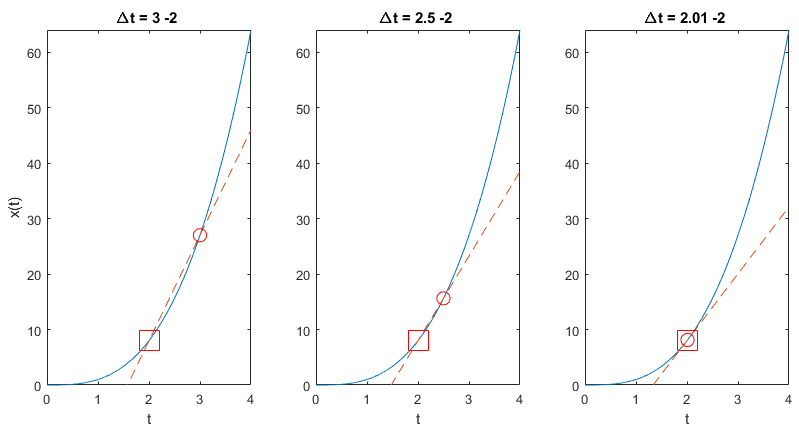


Figura 3.5: aproximación numérica de una recta tangente

¿Qué ocurre para ?

En , … ¿el vehículo no se mueve? OJO Sigue siendo una aproximación…