

Introducción a la programación con MatLAB

Módulo 05

Cálculo simbólico. Resolución de sistemas de ecuaciones lineales.
Derivación e integración simbólica

Fernando E. Pose

Marzo 2019

IEEE
Sección Argentina



Introducción

Objetivos de esta unidad :



- Crear y manipular variables simbólicas
- Resolver expresiones y ecuaciones simbólicas
- Graficar ecuaciones simbólicas
- Introducir a la diferenciación y integración de ecuaciones simbólicas

Creación de variables simbólicas

Declaración de variable simbólica :

1 `x = sym('x')`

2 `syms x`

Ambas formas hacen al carácter 'x' igual a la variable simbólica x.

Variable simbólica utilizando otras existentes :

$$y = \frac{2 * (x + 3)^2}{x^2 + 6 * x + 9}$$

Creación de variables simbólicas

Declaración de variable simbólica :

1 `x = sym('x')`

2 `syms x`

Ambas formas hacen al carácter 'x' igual a la variable simbólica x.

Variable simbólica utilizando otras existentes :

$$y = \frac{2 * (x + 2)^2}{x^2 + 6 * x + 9}$$

Tener en cuenta

El comando **syms** permite crear múltiples variables simbólicas al mismo tiempo.

Creación de variables simbólicas

syms x

$$y = \frac{2 * (x + 2)^2}{x^2 + 6 * x + 9}$$

Workspace	
Name ▲	Value
x	1x1 sym
y	1x1 sym

```
Command Window
>> syms x
>> y = (2*(x+3)^2)/(x^2+6*x+9)

y =
|
(2*(x + 3)^2)/(x^2 + 6*x + 9)
fx >> |
```

Manipulación de expresiones y ecuaciones simbólicas

Extracción de numeradores y denominadores :

Comando

Ver comando : `[num,den] = numden(var)`

Ej. Ejecutar las siguientes líneas. Obtener conclusiones.

```
syms x
y = (2*(x+2)^2)/(x^2+6*x+9)
[num,den] = numden(y)
```

Manipulación de expresiones y ecuaciones simbólicas

Expansión de expresiones :

Comando

Ver comando : `expand(var)`

Ej. Ejecutar las siguientes líneas. Obtener conclusiones.

```
syms x
y = (2*(x+2)^2)/(x^2+6*x+9)
[num,den] = numden(y)
expand(num)
```

Manipulación de expresiones y ecuaciones simbólicas

Factorización de expresiones :

Comando

Ver comando : `factor(var)`

Ej. Ejecutar las siguientes líneas. Obtener conclusiones.

```
syms x
y = (2*(x+2)^2)/(x^2+6*x+9)
[num,den] = numden(y)
factor(num)
```


Manipulación de expresiones y ecuaciones simbólicas

Recolección de términos :

Comando

Ver comando : `collect(var)`

Ej. Ejecutar las siguientes líneas. Obtener conclusiones.

```
syms x
y = (2*(x+2)^2)/(x^2+6*x+9)
[num,den] = numden(y)
collect(num)
```

Simplificación de ecuaciones simbólicas

Simplificación de ecuación :

Comando

Ver comando : `simplify(var)`

Ej. Ejecutar las siguientes líneas. Obtener conclusiones.

```
z = sym( 'x^3-1=(x-3)*(x+3) ' )  
simplify(z)
```

Ejercicio práctico 11

- 1 Cree la variable simbólica x y verifique que se encuentra en el workspace
- 2 Cree las siguientes expresiones simbólicas :
 - $ex1 = x^2 - 1$
 - $ex2 = (x + 1)^2$
- 3 Multiplique $ex1$ por $ex2$ y llame al resultado $y1$
- 4 Divida $ex1$ entre $ex2$ y llame al resultado $y2$
- 5 Use la función **numden** para extraer el numerador y denominador de $y1$ y $y2$
- 6 Use las funciones **factor**, **expand**, **collect** y **simplify** en $y1$ e $y2$.

Resolución de expresiones y ecuaciones simbólicas

Resolución de expresiones y ecuaciones :

Comando

Ver comando : solve()

Se utilizarán dos enfoques, los mismos son :

- 1 Cuando se trata de una expresión
- 2 Cuando se trata de una ecuación
 - 1 Expresión igualada a 0
 - 2 Expresión igualada a una expresión (aplicando transformación)
 - 3 Expresión igualada a una expresión (sin transformación)

Resolución de expresiones y ecuaciones simbólicas : Caso 1

Utilización de la función solve en una expresión :

Ej. Ejecutar las siguientes líneas. Obtener conclusiones.

```
solve ( 'a*x^2+b*x+c ' )
```

Resolución de expresiones y ecuaciones simbólicas : Caso 1

Utilización de la función solve en una expresión :

Ej. Ejecutar las siguientes líneas. Obtener conclusiones.

```
solve ( 'a*x^2+b*x+c ' )
```

Importante

Cuando se usa en una expresión, la función **solve** iguala la expresión a cero y resuelve.

Resolución de expresiones y ecuaciones simbólicas : Caso 1

Especificación de la variable a resolver :

Ej. Ejecutar las siguientes líneas. Obtener conclusiones.

```
solve ( 'a*x^2+b*x+c' , 'a' )
```

Importante

Matlab por defecto resuelve para la variable simbólica x.

Resolución de expresiones y ecuaciones simbólicas : Caso 2.1 ó 2.2

Transformación en una expresión :

Para el caso :

$$5 * x^2 + 6 * x + 3 = 10$$

Se podría reformular como :

$$5 * x^2 + 6 * x - 7 = 0$$

y resolver la ecuación ejecutando las siguientes líneas :

```
solve('5*x^2+6*x-7')
```


Resolución de expresiones y ecuaciones simbólicas : Caso 2.3

Sin transformación de expresión :

Para el caso :

$$5 * x^2 + 6 * x + 3 = 10$$

Se resuelve la ecuación ejecutando las siguientes líneas :

```
E2 = sym( '5*x^2+6*x+3=10' )  
solve(E2)
```

Ejercicio práctico 12

- 1 Cree las variables simbólicas x, a, b y c
- 2 Cree las siguientes expresiones simbólicas :
 - $ex1 = a * x^2 - 1$
 - $ex2 = a * x^2 + b * x + c$
 - $eq1 = a * x^2 = 1$
 - $eq2 = a * x^2 + b * x + c = 0$
- 3 Use la función solve para resolver $ex1$ y $eq1$ tanto para x como para a
- 4 Use la función solve para resolver $ex2$ y $eq2$ tanto para x como para a

Resolución de sistemas de ecuaciones

Resolver el siguiente sistemas de ecuaciones :

$$\begin{cases} 3x + 2y - z = 10 \\ -x + 3y + 2z = 5 \\ x - y - z = -1 \end{cases}$$

Resolución de sistemas de ecuaciones

- Definir las tres ecuaciones simbólicas

```
Ec1 = sym( '3*x+2*y-z=10' )  
Ec2 = sym( '-x+3*y+2*z=5' )  
Ec3 = sym( 'x-y-z=-1' )
```

Resolución de sistemas de ecuaciones

- Definir las tres ecuaciones simbólicas

```
Ec1 = sym( '3*x+2*y-z=10' )  
Ec2 = sym( '-x+3*y+2*z=5' )  
Ec3 = sym( 'x-y-z=-1' )
```

Luego utilizando la función **solve** se obtienen la solución (valores de x, y, z) :

```
[x,y,z] = solve(Ec1,Ec2,Ec3)
```

Graficación de ecuaciones simbólicas

Graficación de $y=f(x)$:

Comando

Ver comando : `ezplot()`

Ej. Ejecutar las siguientes líneas. Obtener conclusiones.

```
y = sym( 'x^2-2' )  
ezplot(y)
```

Graficación de ecuaciones simbólicas

Graficación de $y=f(x)$:

Comando

Ver comando : `ezplot()`

Ej. Ejecutar las siguientes líneas. Obtener conclusiones.

```
y = sym( 'x^2-2' )  
ezplot(y)
```

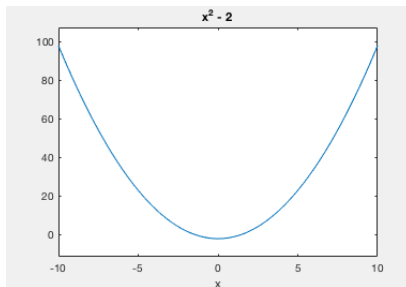
Importante

Por defecto, se grafica la función con una variación de x en el intervalo $[-2 * \pi, 2 * \pi]$

Graficación de ecuaciones simbólicas

Ej. Ejecutar las siguientes líneas. Obtener conclusiones.

```
y = sym('x^2-2')  
ezplot(y,[-10,10])
```



Graficación de ecuaciones simbólicas

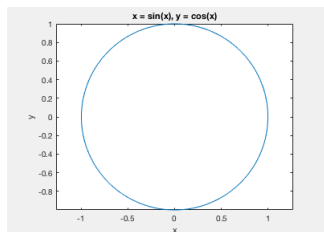
Ecuaciones paramétricas :

$$x = \text{sen}(t)$$

$$y = \cos(t)$$

Ej. Ejecutar las siguientes líneas. Obtener conclusiones.

```
ezplot('sin(x)', 'cos(x)')
```



Graficación de ecuaciones simbólicas

```
y1 = sym('sen(X)')
```

```
y2 = sym('sen(2 * X)')
```

```
y3 = sym('sen(3 * X)')
```

Ej. Ejecutar las siguientes líneas. Obtener conclusiones.

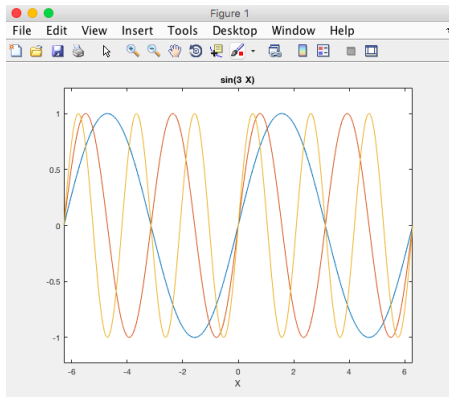
```
ezplot(y1)
```

```
hold on
```

```
ezplot(y2)
```

```
ezplot(y3)
```

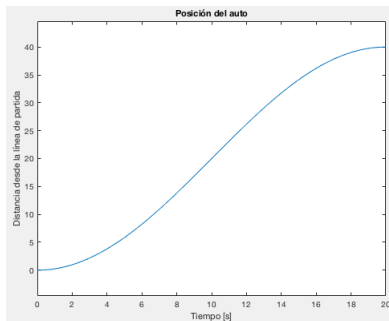
Graficación de ecuaciones simbólicas



Cálculo : Introducción a la diferenciación

Se considera un auto de carreras cuya ecuación de posición es :

$$d = 20 + 20 * \text{sen}\left(\frac{\pi * (t - 10)}{20}\right)$$



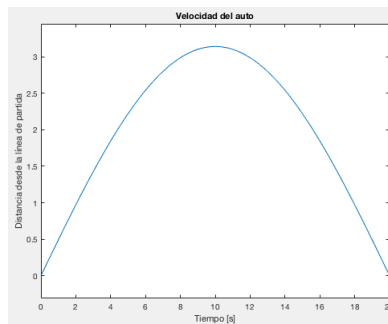
Cálculo : Introducción a la diferenciación

Sabiendo que la velocidad es la derivada de la posición y utilizando la función **diff**

Comando

Ver comando : `diff()`

Curva de velocidad



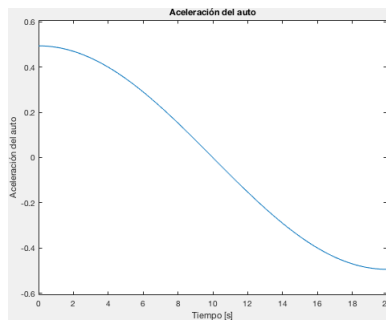
Cálculo : Introducción a la diferenciación

Sabiendo que la aceleración es la derivada de la velocidad y utilizando la función **diff**

Comando

Ver comando : `diff()`

Curva de aceleración



Cálculo : Introducción a la diferenciación

Funciones de diferenciación simbólica :

<code>diff(f)</code>	Derivada de f respecto a la variable independiente
<code>diff(f,'t')</code>	Derivada de f respecto a la variable t
<code>diff(f,n)</code>	Derivada n-ésima de f respecto a la variable independiente
<code>diff(f,'t',n)</code>	Derivada n-ésima de f respecto a la variable t

Ejercicio práctico 13

- 1 Encuentre la primera derivada con respecto a x de las siguientes expresiones :

1 $x^2 + x + 1$

2 $\text{sen}(x)$

- 2 Encuentre la primera derivada parcial con respecto a x de las siguientes expresiones :

1 $x^{0.5} - 3 * y$

2 $3 * x + 4 * y - 3 * x * y$

- 3 Encuentre la segunda derivada con respecto a x para cada una de las expresiones del problema 1 y 2.

- 4 Encuentre la primera derivada con respecto a y para las siguientes expresiones :

1 $y - 1$

2 $a * y + b * x + c * z$

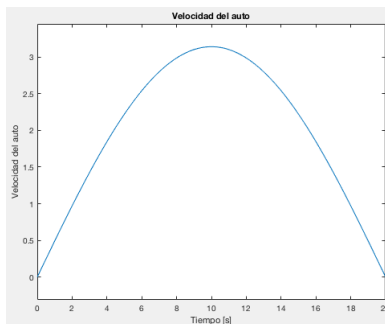
Cálculo : Introducción a la integración

Dada la curva de aceleración se procede a calcular la velocidad integrando la misma.

Comando

Ver comando : `int()`

Curva de velocidad



Cálculo : Introducción a la integración

Cálculo de integral definida :

<code>int(f)</code>	Integral de f respecto a la variable independiente
<code>int(f,'t')</code>	Integral de f respecto a la variable t
<code>int(f,a,b)</code>	Integral respecto a la variable independiente de f entre a y b

Ejercicio práctico 21

1 Integre las siguientes expresiones con respecto a x :

1 $x^2 + x + 1$

2 $\tan(X)$

2 Integre las siguientes expresiones con respecto a x :

1 $x^{0.5} - 3 * y$

2 $3 * x + 4 * y - 3 * x * y$

3 Realice una integración doble con respecto a x para cada una de las expresiones de los problemas 1 y 2.

4 Integre las siguientes expresiones con respecto a y :

1 $y - 1$

2 $a * y + b * x + c * z$

Bases de Matlab

