Tema 12
Introducción a las Ecuaciones Diferenciales Ordinarias

Objetivos:

1. Definir familias de curvas planas con wxMaxima. Representación gráfica.
2. Aprender la sintaxis de ecuaciones diferenciales con wxMaxima.
3. Aprender a resolver algunas ecuaciones diferenciales de primer orden. Tipo de soluciones.
4. Aprenden a resolver algunas ecuaciones diferenciales de segundo orden. Tipo de soluciones.
5. Estudiar algunas aplicaciones de las ecuaciones diferenciales
6. Estudiar la resolución numérica de ecuaciones diferenciales.

Contenidos:

12-1. Familias de curvas planas. Ecuación diferencial de una familia de curvas.
12-2. Ecuaciones diferenciales ordinarias de orden 1. Sistemas.
12-3. Ecuaciones diferenciales ordinarias de orden 2.
12-4. Introducción a los métodos numéricos de resolución de ecuaciones diferenciales ordinarias y sistemas.

Referencias

AEM11 ALANINOS PRATS, J; EXTREMERA LIZANA, J; MUÑOZ RIVAS, P. (2011)
Cálculo con wxMaxima.

APJ11 ALANINOS PRATS, J; APARICIO DEL PRADO, C; EXTREMERA LIZANA, J; MUÑOZ RIVAS, P.; VILLENA MUÑOZ, A.R. (2011)
Prácticas de ordenador con wxMaxima.

ABD91 A. AUBANELL, A. BENSENY, A. DELSHAMS (1991)
Eines bàsiques de Càlcul Numèric

BR09 BRUZÓN GALLEGO, M. DE LOS SANTOS; RAMÍREZ LABRADOR, JOSÉ (2009)
Modelos matemáticos con Maxima

ES08 ESTELA CARBONELL, M. ROSA; SAÀ SEOANE, JOEL (2008)
Cálculo con soporte interactivo en moodle.

GV07a GONZÁLEZ VEGA, LAUREANO (2007)
Laboratorio de Matemáticas. Vol. 1: números y ecuaciones

GV07b GONZÁLEZ VEGA, LAUREANO (2007)
Laboratorio de Matemáticas. Vol. 2: límites y derivadas

JB01 JARAUTA BRAGULAT, EUSEBI (2001)
Anàlisi Matemàtica d'una variable. Fonaments i aplicacions.

RR08b RODRÍGUEZ RIOTORTO, MARIO (2008)
Curso intensivo i-MATH de software libre orientado a Ciencias e Ingeniería

SI93 George F. SIMMONS (1993)
Ecuaciones Diferenciales.

VR09 VALLEJO RODRÍGUEZ, JOSÉ ANTONIO (2009)
Cálculo diferencial con Maxima

ZI88 ZILL, Daniel G. (1988)
Ecuaciones Diferenciales con aplicaciones (2ª edición).

12-1.- Familias de curvas planas. Ecuación diferencial de una familia de curvas.

Los contenidos de este apartado se desarrollan en el archivo **Tema_12-1.wxm**.

Una familia uniparamétrica de curvas planas es un conjunto de curvas en el plano que se pueden representar en forma implícita para una ecuación del tipo:

$$F(t, y, C) = 0, \quad C \in \mathbb{R}$$

donde es C un parámetro (es decir un número cualquiera), t es la variable independiente e y es la variable dependiente. En ocasiones la ecuación anterior permite expresar la ecuación en forma explícita:

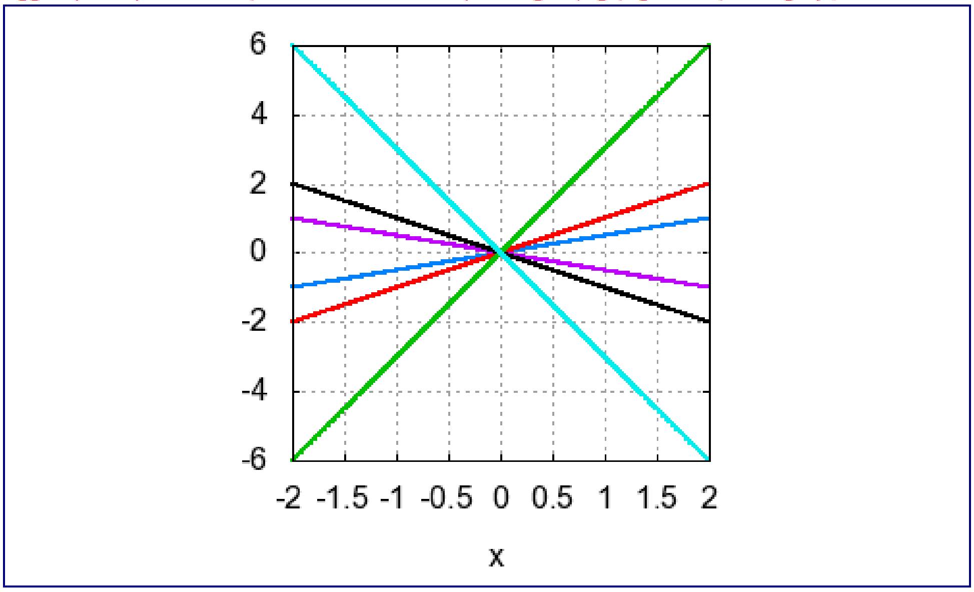
$$y = f(t, C), \quad C \in \mathbb{R}$$

Ejemplo 12.1.1. El conjunto de curvas dado por la ecuación $y = Ct,\ C \in \mathbb{R}$ es una familia uniparamétrica de curvas planas (rectas que pasan por el origen). Veamos su definición y representación gráfica con wxMaxima.

```
(%i1) f11(x):= 0.5*x$          f12(x):=x$       f13(x):=3*x$
      f14(x):=-0.5*x$          f15(x):=-x$      f16(x):=-3*x$
```

```
(%i7) wxplot2d([f11(x),f12(x),f13(x),f14(x),f15(x),f16(x)], [x,-2,2],
      [style, [lines, 2]] , [legend, ""],
       [gnuplot_preamble, "set size ratio 1; set grid;"] , [nticks, 300]  )$
```

(%t7)

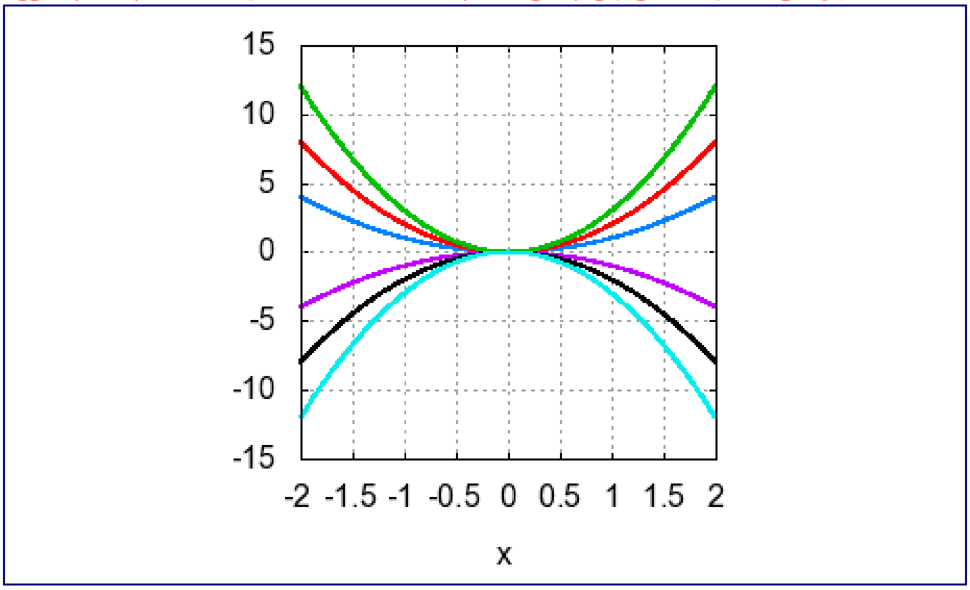


Ejemplo 12.1.2. El conjunto de curvas dado por la ecuación $y = Ct^2, C \in \mathbb{R}$ es una familia uniparamétrica de curvas planas (parábolas que pasan por el origen). Veamos su definición y representación gráfica con wxMaxima.

```
(%i1) f31(x):=x^2$         f32(x):=2*x^2$      f33(x):=3*x^2$
      f34(x):=-x^2$        f35(x):=-2*x^2$      f36(x):=-3*x^2$
```

```
(%i7) wxplot2d([f31(x),f32(x),f33(x),f34(x),f35(x),f36(x)], [x,-2,2],
      [style, [lines, 2]] , [legend, ""],
      [gnuplot_preamble, "set size ratio 1; set grid;"] , [nticks, 300]   )$
```

(%t7)

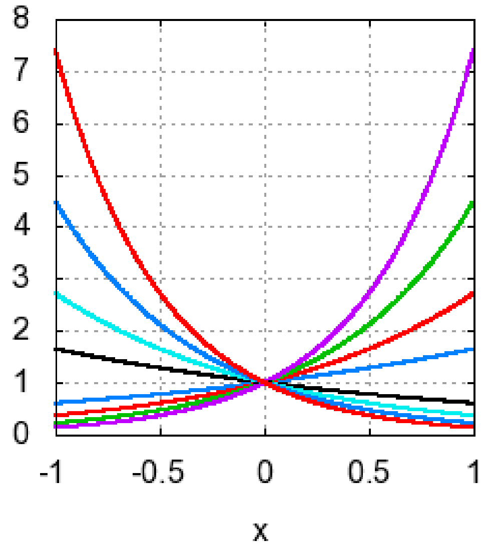


Ejemplo 12.1.3. El conjunto de curvas dado por la ecuación $y = e^{at}, a \in \mathbb{R}$ es una familia uniparamétrica de curvas planas (funciones exponenciales). Veamos su definición y representación gráfica con wxMaxima.

```
(%i1) f11(x):=exp(0.5*x)$    f12(x):=exp(x)$
      f13(x):=exp(1.5*x)$    f14(x):=exp(2*x)$
      f21(x):=exp(-0.5*x)$   f22(x):=exp(-x)$
      f23(x):=exp(-1.5*x)$   f24(x):=exp(-2*x)$
```

```
(%i9) wxplot2d([f11(x),f12(x),f13(x),f14(x),f21(x),f22(x),f23(x),f24(x)], [x,-1,1],
      [style, [lines, 2]] , [legend, ""],
      [gnuplot_preamble, "set size ratio 1; set grid;"] , [nticks, 300] )$
```

(%t9)

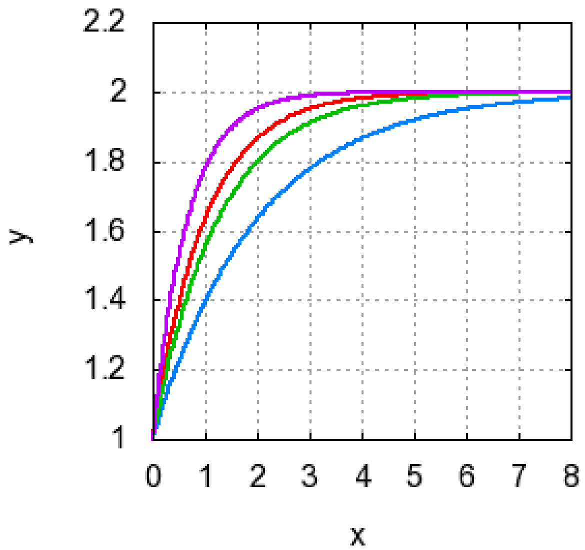


Ejemplo 12.1.4. El conjunto de curvas dado por la ecuación $y = 2 - e^{at}$, $a \in \mathbb{R}$ es una familia uniparamétrica de curvas planas (funciones exponenciales). Veamos su definición y representación gráfica con wxMaxima.

```
(%i1) f11(x):=2 - exp(- 0.5*x)$     f12(x):=2 - exp(- 1.0*x)$
      f13(x):=2 - exp(- 0.8*x)$     f14(x):=2 - exp(- 1.5*x)$
```

```
(%i5) wxplot2d([f11(x),f12(x),f13(x),f14(x)], [x, 0, 8], [y, 1.0, 2.2],
      [style, [lines, 2]] , [legend, ""],
      [gnuplot_preamble, "set size ratio 1;  set grid;"] , [nticks, 10] )$
```

(%t5)



Una familia biparamétrica de curvas planas es un conjunto de curvas en el plano que se pueden representar en forma implícita para una ecuación del tipo:

$$F(t, y, C_1, C_2) = 0, \quad C_1, C_2 \in \mathbb{R}$$

donde $C_1, C_2 \in \mathbb{R}$ son los parámetros (es decir números cualesquiera), t es la variable independiente e y es la variable dependiente. En ocasiones la ecuación anterior permite expresar la ecuación en forma explícita:

$$y = f(t, C_1, C_2), \quad C_1, C_2 \in \mathbb{R}$$

Ejemplo 12.1.5. El conjunto de curvas dado por la ecuación

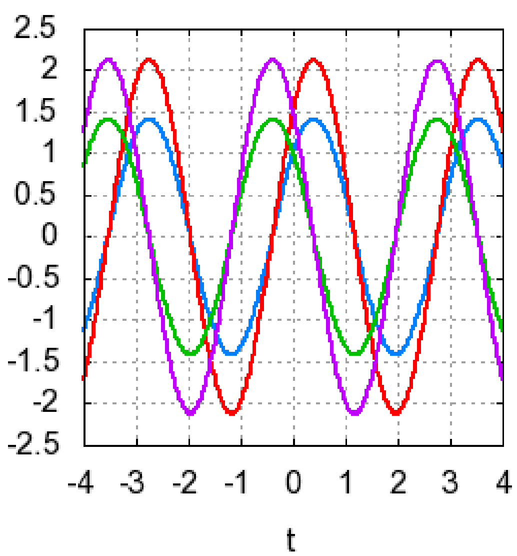
$$y = C_1 \cos(2t) + C_2 \sin(2t), \quad C_1, C_2 \in \mathbb{R}$$

es una familia biparamétrica de curvas planas (funciones trigonométricas). Veamos su definición y representación gráfica con wxMaxima.

```
(%i1) f51(t):=cos(2*t) + sin(2*t)$      f52(t):=1.5*cos(2*t) + 1.5*sin(2*t)$
      f53(t):=cos(2*t) - sin(2*t)$      f54(t):=1.5*cos(2*t) -  1.5*sin(2*t)$
```

```
(%i5) wxplot2d([f51(t),f52(t),f53(t),f54(t)], [t,-4,4],
       [style, [lines, 2]] , [legend, ""],
       [gnuplot_preamble, "set size ratio 1;  set grid;"] , [nticks, 10] )$
```

(%t5)



Ecuación diferencial de una familia de curvas planas.

Comencemos por el caso de una familia uniparamétrica. Si en la ecuación de la familia $y = Ct^2,\ C \in \mathbb{R}$ se calcula la derivada considerando que $y = y(t)$ resulta:

$$y'(t) = 2Ct$$

Ahora se puede eliminar el parámetro con las dos ecuaciones: la de la familia y su derivada, obteniéndose:

$$y'(t) = 2\left(\frac{y}{t^2}\right)t \quad \Rightarrow \quad y'(t) = \frac{2y}{t} \quad \Rightarrow \quad 2y - ty' = 0$$

La última ecuación no contiene parámetros y la cumplen todas las curvas de la familia. Esta ecuación, llamada <u>ecuación diferencial de la familia de curvas planas</u>, contiene sólo la variable, la función y la derivada de la función. Esta ecuación se llama ecuación diferencial ordinaria de primer orden (o de orden 1).

Con wxMaxima se procede de la manera siguiente. En primer lugar se escribe la dependencia funcional que se asigna a las variables de la ecuación:

```
(%i1) depends ([y] , [t]);
(%o1) [y(t)]
```

A continuación se escribe la ecuación de la familia de curvas planas y se deriva la ecuación:

```
(%i2) Eq1:  y=C*t^2;
      Eq2:  diff(Eq1,t) ;
```

(%o2) $y = t^2\,C$

(%o3) $\frac{d}{d\,t}\,y = 2\,t\,C$

A continuación se elimina el parámetro entre estas dos ecuaciones con la instrucción “eliminate”, que se aplica con la sintaxis que se ve a continuación:

```
(%i4) eliminate([Eq1, Eq2],  [C]);
```

(%o4) $\left[t\left(2\,y - t\left(\frac{d}{d\,t}\,y\right)\right)\right]$

Finalmente se puede escribir la ecuación diferencial de la familia de curvas:

```
(%i5) edo1: 2*y-t*'diff(y,t) = 0;
```

(%o5) $2\,y - t\left(\frac{d}{d\,t}\,y\right) = 0$

En el caso de una familia biparamétrica, como por ejemplo

$$y = C_1 \cos(2t) + C_2 \sin(2t), \quad C_1, C_2 \in \mathbb{R}$$

se calcula la derivada considerando que $y = y(t)$ resulta:

$$y'(t) = -2C_1 \sin(2t) + 2C_2 \cos(2t), \quad C_1, C_2 \in \mathbb{R}$$

Si se vuelve a derivar en la ecuación anterior, se obtiene:

$$y''(t) = -4C_1 \cos(2t) - 4C_2 \sin(2t), \quad C_1, C_2 \in \mathbb{R}$$

Ahora se pueden eliminar los parámetros con las tres ecuaciones: la de la familia, su derivada y la derivada segunda, obteniéndose:

$$y'' + 4y = 0$$

Esta ecuación no contiene parámetros, la cumplen todas las curvas de la familia y se llama <u>ecuación diferencial de la familia de curvas planas</u>; contiene sólo la variable, la función y/o las derivadas de primer y segundo orden de la función. Esta ecuación se llama ecuación diferencial ordinaria de segundo orden (o de orden 2).

Con wxMaxima se procede como antes, pero ahora se trata de eliminar las dos constantes. Veamos el procedimiento:

```
(%i1) depends ([y] , [t]);
(%o1) [y(t)]
(%i2) Eq1:  y = C1*cos(2*t) + C2*sin(2*t);
      Eq2:  diff(Eq1, t ) ;
      Eq3:  diff(Eq2, t ) ;
```

(%o2) $y = \sin(2\,t)\,C2 + \cos(2\,t)\,C1$

(%o3) $\frac{d}{d\,t}\,y = 2\cos(2\,t)\,C2 - 2\sin(2\,t)\,C1$

(%o4) $\frac{d^2}{d\,t^2}\,y = -4\sin(2\,t)\,C2 - 4\cos(2\,t)\,C1$

```
(%i5) eliminate([Eq1, Eq2, Eq3],  [C1,C2]);
```

(%o5) $\left[-\cos(2\,t)\left(\frac{d^2}{d\,t^2}\,y + 4\,y\right)\right]$

```
(%i6) edo2: 'diff(y,t,2)+4*y=0;
```

(%o6) $\frac{d^2}{d\,t^2}\,y + 4\,y = 0$

En relación a la ecuación diferencial, la familia de curvas planas se llama solución general de la ecuación diferencial y cada una de las curvas de la familia se llama una solución particular.

En una familia uniparamétrica, para calcular una solución particular hace falta conocer una condición sobre la curva, llamada condición inicial, expresada en general en la forma $y(0)=y_0$ o bien $y(t_0)=y_0$; esta condición permite calcular el valor del parámetro y determinar la solución particular correspondiente. Por ejemplo en la familia $y=Ct^2$ si la condición inicial es $y(2)=5$ sustituyendo esta condición en la solución general se obtiene $C=\frac{5}{4}$ y, por lo tanto, la solución particular buscada es $y=\frac{5}{4}t^2$.

Cálculos con wxMaxima:

```
(%i1) y(t):=C*t^2;
      Eq1: y(2)=5;
(%o1) y(t):=C t^2
(%o2) 4 C=5

(%i3) solve(Eq1, C);
(%o3) [C=5/4]

(%i4) C=C:5/4;
(%o4) C=5/4

(%i5) y=C*t^2;
(%o5) y=5 t^2/4
```

En una familia biparamétrica, para calcular una solución particular hay que conocer dos condiciones: una condición sobre la curva y otra sobre la derivada, llamadas condiciones iniciales, expresadas en general en la forma

$$\begin{cases} y(0)=y_0 \\ y'(0)=y'_0 \end{cases}$$

Estas condiciones permiten calcular el valor de los parámetros y determinar la solución particular correspondiente. Por ejemplo en la familia

$$y=C_1\cos(2t)+C_2\sin(2t), \quad C_1,C_2\in\mathbb{R}$$

si las condiciones iniciales son:

$$\begin{cases} y(0) = 1 \\ y'(0) = 3 \end{cases}$$

sustituyendo estas condiciones en la solución general y en la derivada se obtiene $C_1 = 1,\ C_2 = \dfrac{3}{2}$ y, por lo tanto, la solución particular buscada es $y = \cos(2t) + \dfrac{3}{2}\sin(2t)$.

Cálculos con wxMaxima:

```
(%i1) y(t):=C1*cos(2*t) + C2*sin(2*t);
      define ( Dy(t) , diff(y(t),t) ) ;
(%o1) y(t):=C1 cos(2 t)+C2 sin(2 t)
(%o2) Dy(t):=2 cos(2 t) C2-2 sin(2 t) C:
(%i3) Eq1:   y(0)=1;
      Eq2:   Dy(0)=3;
(%o3) C1=1
(%o4) 2 C2=3
(%i5) C1=C1:1;        C2=C2:3/2;
(%o5) C1=1
(%o6) C2=3/2
(%i7) 'y(t) = y(t) ;
(%o7) y(t)=3 sin(2 t)/2+cos(2 t)
```

12-2.- Ecuaciones diferenciales ordinarias de orden 1. Sistemas.

Los contenidos de este apartado se desarrollan en el archivo **Tema_12-2.wxm**.

Una ecuación diferencial ordinaria (EDO) de orden 1 o de primer orden es una ecuación de la forma $F(t, y, y') = 0$ en la que t indica una variable y y una función de t.

Resolver o integrar una EDO es determinar la familia de curvas que cumple la ecuación.

Se llama solución o integral general de una EDO de primer orden, una familia de curvas planas de la forma $\Phi(t, y, C) = 0$ que cumple la ecuación diferencial, donde C es un parámetro o constante arbitraria.

Se llama solución o integral particular de una EDO de primer orden, una curva plana que forma parte de la solución general, en la que se ha determinado el valor concreto del parámetro. Para determinar el parámetro hay que tener una información que se llama condición inicial, dada generalmente mediante $y(t_0) = y_0$.

No es nuestro objetivo desarrollar una teoría completa de EDO, sino mostrar la metodología de resolución con wxMaxima. En todo caso, se comentarán los principales tipos de EDO de orden 1, la metodología y sintaxis de cálculo de la solución general y de una solución particular y se comentarán los casos en que este proceso no se puede llevar a cabo fácilmente con wxMaxima.

Los tipos de EDO de orden 1 que se comentarán en este tema son los siguientes:

- De variables separadas (o separables)
- Lineales
- Reducibles a lineales (Bernoulli y Ricatti).

Hay otros tipos de EDO que en el contexto de un desarrollo teórico requieren desarrollar la metodología de resolución; en este caso es irrelevante ya que el programa las resuelve aplicando la misma metodología.

12.2.1.- Ecuaciones de variable separadas

Una EDO de primer orden se llama de variables separadas o separables, si es de la forma

$$P(t) + Q(y)y' = 0.$$

También se escribe a menudo en la forma

$$P(t)dt + Q(y)dy = 0.$$

La solución general de una EDO de variables separadas se obtiene calculando la integral indefinida de las funciones $P(t), Q(y)$ y se expresa mediante:

$$\int P(t)dt = -\int Q(y)dy + C$$

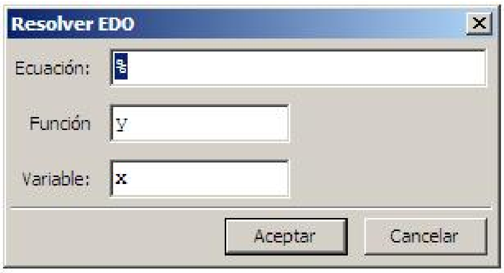
Ejemplo 12.2.1. Resolver la ecuación diferencial $2y' - y = 0$.

En primer lugar hay que escribir la ecuación diferencial y asignarle una referencia:

```
(%i1) EDO_vs1: 2*diff(y(t),t) - y(t) = 0;
```

(%o1) $2\left(\frac{d}{d\,t}y(t)\right)-y(t)=0$

A continuación hay que ir al menú "Ecuaciones" y seleccionar la opción "Resolver EDO", tal como se puede ver en la figura:

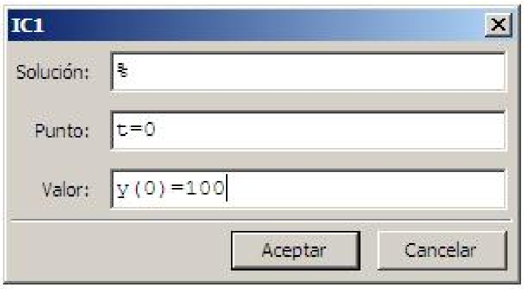


En la primera casilla hay que escribir la referencia asignada a la ecuación diferencial: en este caso es EDO_vs1. En la segunda casilla hay que poner la función con indicación de la variable independiente entre paréntesis: en este caso y(t). En la tercera casilla hay que indicar sencillamente la variable independiente: en este caso t. Cuando se aplica esta sintaxis, resulta:

```
(%i2) ode2(EDO_vs1, y(t), t);
```

(%o2) $y(t)=\%c\ \%e^{t/2}$

Por lo tanto, la solución general es la familia uniparamétrica $y(t) = Ce^{t/2}$, $C \in \mathbb{R}$. Si se quiere resolver un problema de valor inicial, es decir, determinar una solución particular, hay que ir al menú "Ecuaciones" y seleccionar la opción "Problema de valor inicial (1)", tal como se puede ver a continuación:

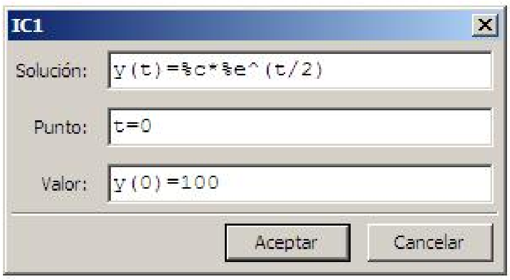


En la primera casilla hay que escribir el signo “%” si esta instrucción se aplica inmediatamente después de la anterior, ya que el programa interpreta que nos referimos a la solución general acabada de calcular. En la segunda casilla hay que poner el valor de la variable independiente: en este caso t = 0. En la tercera casilla hay que indicar el valor de la función en el punto anterior: en este caso y(0) = 100. Cuando se aplica esta sintaxis, resulta:

(%i3) ic1(% , t=0, y(0)=100);

(%o3) $y(t)=100\ \%e^{t/2}$

También se puede calcular la solución particular seleccionando la solución en el output de wxMaxima correspondiente y yendo al cuadro de diálogo anterior, donde se observará que la solución se engancha en la primera casilla. El resultado es



En wxMaxima se obtiene:

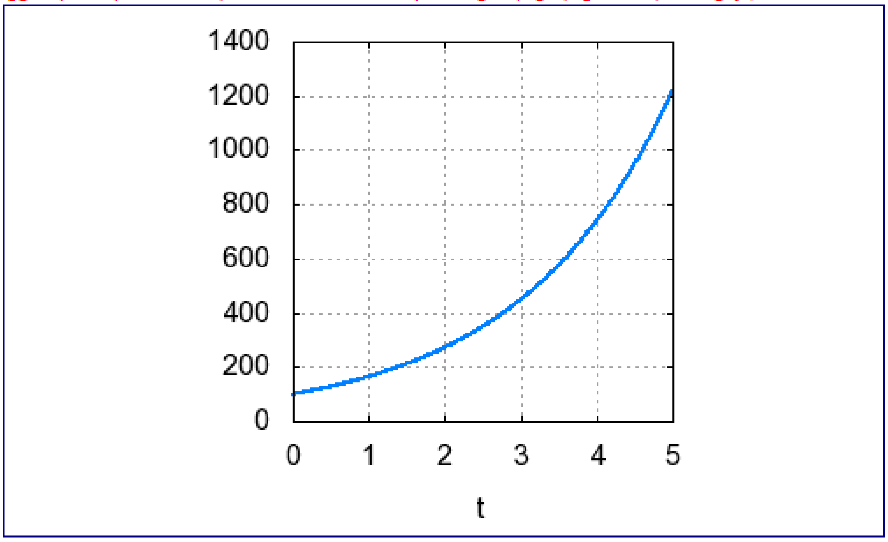
(%i4) ic1(y(t)=%c*%e^(t/2) , t=0, y(0)=100);

(%o4) $y(t)=100\ \%e^{t/2}$

Si se quiere se puede obtener una representación gráfica de esta solución particular:

```
(%i5) y(t):=100*%e^(t/2)$
      wxplot2d([y(t)], [t, 0, 5], [ylabel, ""]   , [style, [lines, 2]] ,
      [gnuplot_preamble, "set size ratio 1; set grid;"]  , [nticks, 100] )$
```

(%t6)



Ejemplo 12.2.2. Resolver la ecuación diferencial $y' - 2(100 - y) = 0$.

Veamos la resolución con wxMaxima de acuerdo con el procedimiento descrito en el ejemplo anterior:

```
(%i1) EDO_vs2:diff(y(x),x)=2*(100-y(x));
```

(%o1) $\frac{d}{d\,x}\,y(x) = 2\,(100 - y(x))$

```
(%i2) ode2(EDO_vs2, y(x), x);
```

(%o2) $y(x) = \%e^{-2\,x}\,(100\,\%e^{2\,x} + \%c)$

```
(%i3) expand(%);
```

(%o3) $y(x) = \%c\,\%e^{-2\,x} + 100$

```
(%i4) ic1( % , x=0, y(0)=50);
```

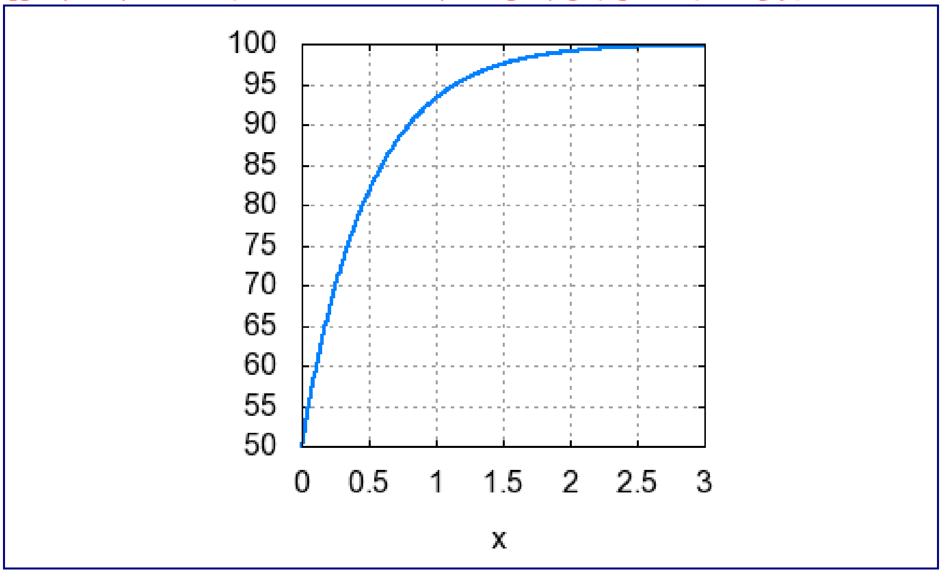
(%o4) $y(x) = \%e^{-2\,x}\,(100\,\%e^{2\,x} - 50)$

```
(%i5) expand(%);
```

(%o5) $y(x) = 100 - 50\,\%e^{-2\,x}$

```
(%i6) y(x):=100-50*%e^(-2*x)$
      wxplot2d([y(x)], [x,0, 3],   [ylabel, ""]   , [style, [lines, 2]] ,
      [gnuplot_preamble, "set size ratio 1; set grid;"]  , [nticks, 100] )$
```

(%t7)



12.2.2.- Ecuaciones diferenciales lineales

Se llaman EDO lineales de primer orden las ecuaciones de la forma:

$$y'+G(t)y=F(t)$$

donde F, G son funciones reales continuas y derivables en un cierto abierto $\Omega\subset\mathbb{R}$. Si $F(t)=0$ la ecuación se llama homogénea y si $F(t)\neq 0$ la ecuación se llama completa. Como es sabido, la solución general de la ecuación completa se obtiene sumando la solución general de la ecuación homogénea con una solución particular de la ecuación completa, es decir:

$$y(t)=y_H(t)+y_p(t)=K\,\mathrm{e}^{-\int G(t)dt}+\mathrm{e}^{-\int G(t)dt}\int F(t)e^{\int G(t)dt}dt,\ \ K\in\mathbb{R}$$

Ejemplo 12.2.3.- Consideremos la ecuación diferencial $t^2-y=ty'$; se trata de una ecuación lineal ya que se puede escribir:

$$y'+\frac{1}{t}y=t\ ;\ \ G(t)=\frac{1}{t},\ F(t)=t$$

Veamos la resolución con wxMaxima:

(%i1) EDO_lin1:   t^2 - y(t) = t * diff(y(t), t) ;

(%o1) $t^2 - y(t) = t\left(\frac{d}{d\,t}\,y(t)\right)$

(%i2) ode2(EDO_lin1, y(t), t);

(%o2) $y(t) = \dfrac{\frac{t^3}{3} + \%c}{t}$

(%i3) expand(%);

(%o3) $y(t) = \dfrac{t^2}{3} + \dfrac{\%c}{t}$

Por lo tanto, la solución general de la ecuación considerada es:

$$y(t) = C\frac{1}{t} + \frac{t^2}{3}, \;\; C \in \mathbb{R}$$

Obsérvese que la solución general se da directamente y se puede interpretar cada uno de los términos de acuerdo con la estructura de la solución general comentada antes.

Si se quiere calcular una solución particular, por ejemplo, la que cumple las condiciones iniciales t = 1, $y(1) = 5$, se obtiene:

(%i4) ic1(y(t)=t^2/3+%c/t, t=1, y(1)=5);

(%o4) $y(t) = \dfrac{t^3 + 14}{3\,t}$

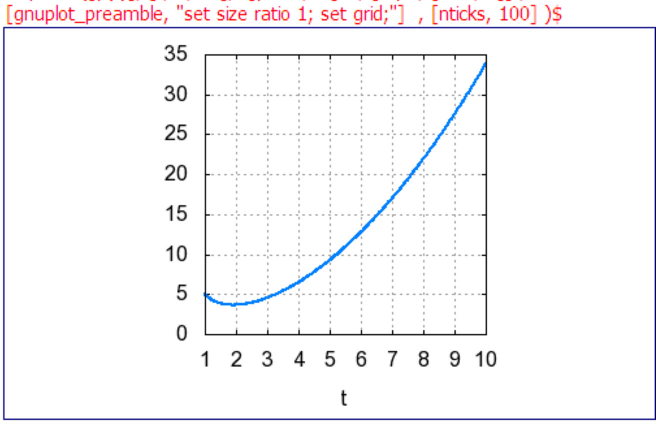
(%i5) expand(%);

(%o5) $y(t) = \dfrac{t^2}{3} + \dfrac{14}{3\,t}$

Finalmente se puede hacer una representación gráfica de esta solución particular:

```
(%i6) y(t):=t^2/3+14/(3*t)$
      wxplot2d([y(t)], [t, 1, 10],  [ylabel, ""]   , [style, [lines, 2]] ,
      [gnuplot_preamble, "set size ratio 1; set grid;"]  , [nticks, 100] )$
```

(%t7)



12.2.3.- Ecuaciones de Bernoulli

Se llaman ecuaciones de Bernoulli las ecuaciones que tienen la forma general

$$y' + G(t)y = F(t)y^n, \;\; n \in \mathbb{N}$$

Obsérvese que si $n = 0$, $n = 1$ se trata entonces de una ecuación lineal. Para resolver las ecuaciones de Bernoulli se lleva a cabo el cambio de función definido por:

$$u = y^{1-n} \quad (n \geq 2)$$

Entonces se cumple:

$$u' = (1-n)y^{-n}y' \quad \Rightarrow \quad y' = \frac{1}{1-n}y^n u'$$

Sustituyendo en la ecuación inicial resulta:

$$u' + (1-n)G(t)u = (1-n)F(t) \qquad (n \geq 2)$$

que es una ecuación lineal y que ya se sabe cómo resolver.

Ejemplo 12.2.4. Se considera la ecuación diferencial:

$$y' + 2ty + ty^4 = 0$$

Se trata de una ecuación de Bernoulli:

$$y' + 2ty = -ty^4; \;\; F(t) = -t, \; G(t) = 2t$$

Veamos la resolución con wxMaxima:

```
(%i1) EDO_ber1: diff(y(t), t) + 2*t*y(t) + t*y(t)^4=0;
```

$$(\%o1) \; \frac{d}{d\,t}\,y(t) + t\,y(t)^4 + 2\,t\,y(t) = 0$$

```
(%i2) ode2(EDO_ber1, y(t), t);
```

$$(\%o2) \; \frac{\log(y(t)^3+2)-\log(y(t)^3)}{6} = \frac{t^2}{2} + \%c$$

```
(%i3) logcontract(%);
```

$$(\%o3) \; -\frac{\log\left(\frac{y(t)^3}{y(t)^3+2}\right)}{6} = \frac{t^2}{2} + \%c$$

Ahora expresaremos la solución en la forma más simple posible:

```
(%i4) log((y(t)^3 + 2)/(y(t)^3))=3*t^2+C1;
```

(%o4) $\log\left(\frac{y(t)^3+2}{y(t)^3}\right)=C1+3\,t^2$

```
(%i5) ((y(t)^3+2)/y(t)^3)=exp(C1+3*t^2);
```

(%o5) $\frac{y(t)^3+2}{y(t)^3}=\%e^{C1+3\,t^2}$

```
(%i6) eq1:  (y(t)^3+2)/y(t)^3=%e^(C1+3*t^2)$
      solve(eq1, y(t));
```

(%o7) $\left[y(t)=\frac{2^{1/3}\sqrt{3}\,\%i-2^{1/3}}{2\left(\%e^{C1+3\,t^2}-1\right)^{1/3}},\ y(t)=-\frac{2^{1/3}\sqrt{3}\,\%i+2^{1/3}}{2\left(\%e^{C1+3\,t^2}-1\right)^{1/3}},\ y(t)=\frac{2^{1/3}}{\left(\%e^{C1+3\,t^2}-1\right)^{1/3}}\right]$

Por lo tanto, la solución buscada es:

```
(%i8) y(t):=2^(1/3)/(%e^(C1+3*t^2)-1)^(1/3);
```

(%o8) $y(t):=\frac{2^{1/3}}{\left(\%e^{C1+3\,t^2}-1\right)^{1/3}}$

Calculemos ahora una solución particular, aplicando la condición inicial y(0) = 3. Se tendrá:

```
(%i9) y(0);
```

(%o9) $\frac{2^{1/3}}{\left(\%e^{C1}-1\right)^{1/3}}$

```
(%i10) eq2:   y(0)=1;
       solve(eq2, C1);
```

(%o10) $\frac{2^{1/3}}{\left(\%e^{C1}-1\right)^{1/3}}=1$

(%o11) $[C1=\log(3)]$

Por lo tanto, la solución particular es:

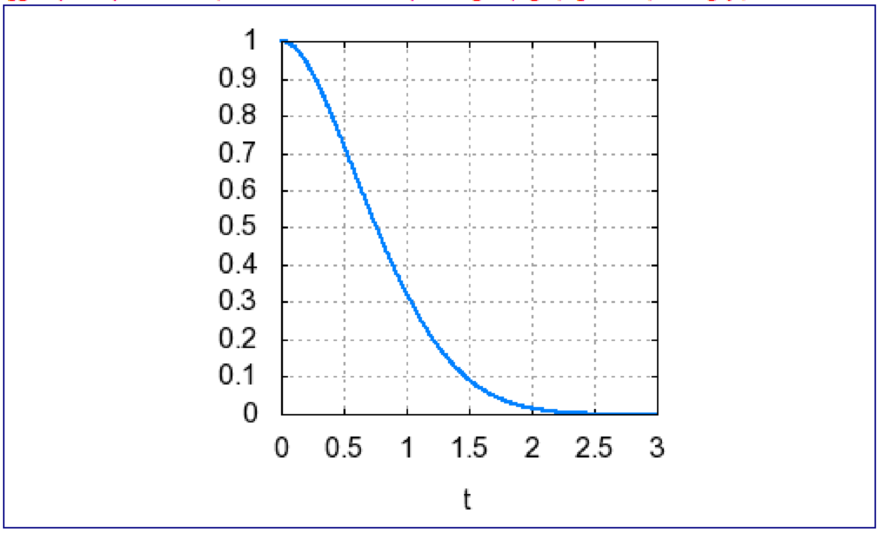
```
(%i12) yp(t):=2^(1/3)/(%e^(log(3)+3*t^2)-1)^(1/3);
```

(%o12) $yp(t):=\frac{2^{1/3}}{\left(\%e^{\log(3)+3\,t^2}-1\right)^{1/3}}$

Representación gráfica de esta solución particular:

```
(%i13) wxplot2d([yp(t)], [t, 0, 3], [ylabel, ""] , [style, [lines, 2]] ,
       [gnuplot_preamble, "set size ratio 1; set grid;"] , [nticks, 100] )$
```

(%t13)



12.2.4.- Ecuaciones de Ricatti

Se llaman ecuaciones de Ricatti las ecuaciones que tienen la forma general

$$y' = F_1(t) + F_2(t)y + F_3(t)y^2$$

en la que, las funciones $F_1(t), F_2(t), F_3(t)$ son continuas y derivables en un cierto intervalo abierto de la recta real. Para resolver una ecuación de Ricatti se lleva a cabo el cambio de función definido por:

$$y = f_1 + \frac{1}{u}; \quad y' = f'_1 - \frac{u'}{u^2}; \quad u = \frac{1}{y - f_1}$$

donde la función $f_1(t)$ es una solución particular de la ecuación de Ricatti y que hay que conocer previamente. Entonces sustituyendo en la ecuación de Ricatti resulta:

$$u' + \left(F_2(t) + 2F_3(t)f_1(t)\right)u = -F_3(t)$$

que es una ecuación lineal y que ya se sabe como resolver.

Ejemplo 12.2.5. Se considera la ecuación diferencial de Ricatti:

$$t(t-1)y' - (1-2t)y + y^2 - 2t = 0$$

Veamos como se puede resolver con wxMaxima:

```
(%i1) EDO_ric1: t*(t-1)*diff(y(t), t) - (1-2*t)*y(t) + y(t)^2 = 2*t;
```

$$(\%o1)\ (t-1)\,t\left(\frac{d}{d\,t}\,y(t)\right)+y(t)^2-(1-2\,t)\,y(t)=2\,t$$

```
(%i2) ode2(EDO_ric1, y(t), t);
(%o2) false
```

Como podemos observar, el programa no la puede resolver con esta instrucción. Aplicaremos la metodología de resolución expuesta para este tipo de ecuaciones. En primer lugar observamos que la función $f_1(t)=1$ es una solución particular de la ecuación, con lo que haremos el cambio de función definido por:

```
(%i3) y(t):=1+1/u(t);
```

$$(\%o3)\ y(t):=1+\frac{1}{u(t)}$$

Ahora calculamos la derivada y sustituyendo en la ecuación dada:

```
(%i4) diff(y(t), t);
```

$$(\%o4)\ -\frac{\frac{d}{d\,t}\,u(t)}{u(t)^2}$$

```
(%i5) EDO_ric2: t*(t-1)*(-'diff(u(t),t,1)/u(t)^2) - (1-2*t)*y(t) + y(t)^2 = 2*t;
```

$$(\%o5)\ -\frac{(t-1)\,t\left(\frac{d}{d\,t}\,u(t)\right)}{u(t)^2}-(1-2\,t)\left(\frac{1}{u(t)}+1\right)+\left(\frac{1}{u(t)}+1\right)^2=2\,t$$

```
(%i6) ratsimp(%);
```

$$(\%o6)\ -\frac{(t^2-t)\left(\frac{d}{d\,t}\,u(t)\right)-2\,t\,u(t)^2+(-2\,t-1)\,u(t)-1}{u(t)^2}=2\,t$$

Simplificamos esta ecuación:

```
(%i7) EDO_ric3:  (t^2-t)*('diff(u(t),t,1)) + (-2*t-1)*u(t) - 1 = 0;
```

$$(\%o7)\ (t^2-t)\left(\frac{d}{d\,t}\,u(t)\right)+(-2\,t-1)\,u(t)-1=0$$

Calculamos la solución general de esta ecuación:

(%i8) ode2(EDO_ric3, u(t), t);

$$(\%o8)\ u(t)=\frac{(t-1)^3\left(\%c-\frac{1}{3\,t^3-9\,t^2+9\,t-3}\right)}{t}$$

Simplificando nuevamente, se obtiene:

(%i10) u(t):=C*(t-1)^3/t - 1/(3*t);

$$(\%o10)\ u(t):=\frac{C\,(t-1)^3}{t}-\frac{1}{3\,t}$$

Por lo tanto, la solución general de la ecuación considerada es:

(%i12) 'y(t)=y(t);

$$(\%o12)\ y(t)=\frac{1}{\frac{(t-1)^3\,C}{t}-\frac{1}{3\,t}}+1$$

Podemos simplificar esta expresión:

(%i13) ratsimp(%);

$$(\%o13)\ y(t)=\frac{(3\,t^3-9\,t^2+9\,t-3)\,C+3\,t-1}{(3\,t^3-9\,t^2+9\,t-3)\,C-1}$$

(%i14) y(t):=(3*C*(t-1)^3 +3*t - 1) / (3*C*(t-1)^3 - 1);

$$(\%o14)\ y(t):=\frac{3\,C\,(t-1)^3+3\,t-1}{3\,C\,(t-1)^3-1}$$

Ahora calcularemos la solución particular tal que y(2) = 2:

(%i15) eq5: y(2)=2;
solve(eq5, C);

$$(\%o15)\ \frac{3\,C+5}{3\,C-1}=2$$

$$(\%o16)\ [C=\frac{7}{3}]$$

Por lo tanto:

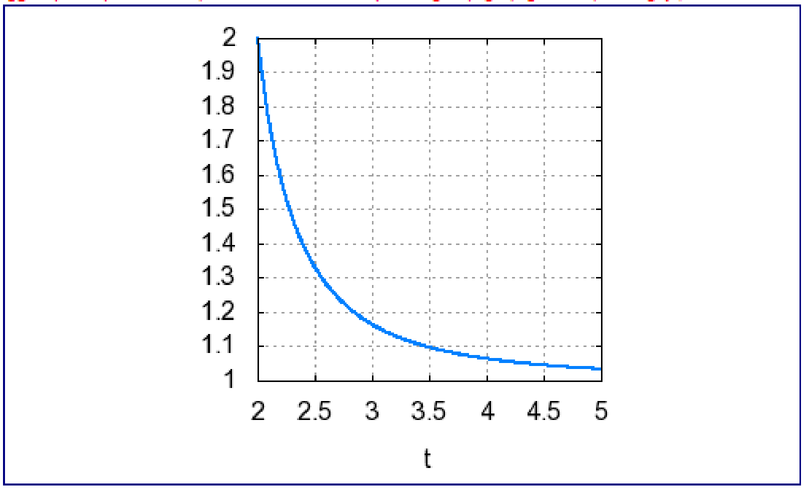
(%i17) yp(t):=(3*(7/3)*(t-1)^3+3*t-1)/(3*(7/3)*(t-1)^3-1);

$$(\%o17)\ yp(t):=\frac{3\,\frac{7}{3}\,(t-1)^3+3\,t-1}{3\,\frac{7}{3}\,(t-1)^3-1}$$

Representación gráfica:

```
(%i18) wxplot2d([yp(t)], [t, 2, 5],  [ylabel, ""]   , [style, [lines, 2]] ,
        [gnuplot_preamble, "set size ratio 1; set grid;"]  , [nticks, 100] )$
```

(%t18)



12.2.5.- Sistemas de EDO de primer orden

Un sistema de EDO de orden 1 o de EDO de primer orden es un sistema de ecuaciones diferenciales ordinarias de la forma:

$$\begin{cases} x_1' = f_1\left(t, x_1, x_2, ..., x_n\right) \\ x_2' = f_2\left(t, x_1, x_2, ..., x_n\right) \\ ..... \\ x_n' = f_n\left(t, x_1, x_2, ..., x_n\right) \end{cases}$$

Un caso particular muy importante es el de los sistemas de EDO lineales de orden 1, que son sistemas de EDO en el que las funciones del segundo miembro de cada ecuación son combinaciones lineales de las variables dependientes; estas combinaciones pueden ser de coeficientes constantes o no. Así, un sistema de EDO lineales de primer orden es:

$$\begin{cases} x_1' = a_1^1(t)x_1 + a_2^1(t)x_2 + \cdots + a_n^1(t)x_n \\ x_2' = a_1^2(t)x_1 + a_2^2(t)x_2 + \cdots + a_n^2(t)x_n \\ ..... \\ x_n' = a_1^n(t)x_1 + a_2^n(t)x_2 + \cdots + a_n^n(t)x_n \end{cases}$$

A continuación ilustraremos la metodología de resolución de sistemas de EDO de primer orden con wxMaxima.

Ejemplo 12.2.6.- Se trata de resolver el sistema de EDO de primer orden:

$$\begin{cases} x' + y' - x = 2t + 1 \\ 2x' + 2y' + x = t \end{cases}$$

En primer lugar se trata de escribir cada una de las EDO del sistema y asignar en cada caso una referencia:

```
(%i1) EDO_1a: 'diff(x(t), t)  - x(t) +  'diff(y(t), t) = 2*t + 1;
      EDO_1b: 2*'diff(x(t), t) + x(t) + 2*'diff(y(t), t)  = t ;
```

$$(\%o1)\ \frac{d}{d\,t}y(t)+\frac{d}{d\,t}x(t)-x(t)=2\,t+1$$

$$(\%o2)\ 2\left(\frac{d}{d\,t}y(t)\right)+2\left(\frac{d}{d\,t}x(t)\right)+x(t)=t$$

A continuación hay que escribir la instrucción que lleva a cabo el cálculo de la solución general del sistema, que en este caso es la instrucción "desolve":

```
(%i3) desolve(  [EDO_1a, EDO_1b]  , [x(t), y(t)] );
```

$$(\%o3)\ [x(t)=-t-\frac{2}{3},y(t)=\frac{t^2}{2}+\frac{4\,t}{3}+\frac{3\,y(0)+3\,x(0)+2}{3}]$$

Si se quiere calcular una solución particular, hay que ver las condiciones iniciales necesarias para hacerlo, que dependen del número de parámetros de la solución general; en este caso hay dos y la sintaxis con wxMaxima es:

```
(%i4) atvalue(x(t), t=0, -2/3);
      atvalue(y(t), t=0, 1);
```

$$(\%o4)\ -\frac{2}{3}$$

$$(\%o5)\ 1$$

Finalmente se pide al programa el cálculo de la solución particular que corresponde a estas condiciones iniciales:

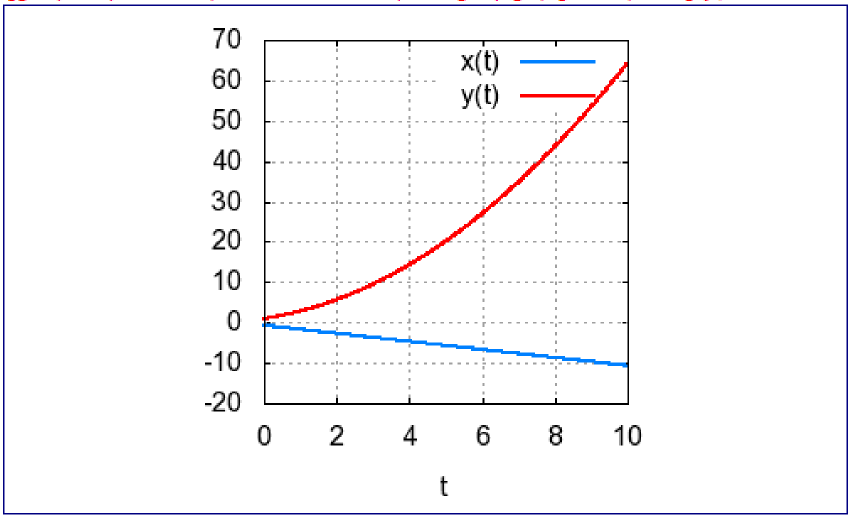
```
(%i6) desolve(  [EDO_1a, EDO_1b]  , [x(t), y(t)] );
```

$$(\%o6)\ [x(t)=-t-\frac{2}{3},y(t)=\frac{t^2}{2}+\frac{4\,t}{3}+1]$$

Como en ejemplos anteriores, ilustramos esta solución particular con una representación gráfica, que tiene como paso previo la definición de las soluciones particulares como funciones:

```
(%i7) x(t):=-t-2/3$  y(t):=t^2/2+(4*t)/3+1$
      wxplot2d([x(t), y(t)], [t, 0, 10],   [legend, "x(t)" , "y(t)"]   , [style, [lines, 2]] ,
      [gnuplot_preamble, "set size ratio 1; set grid;"]  , [nticks, 100] )$
```

(%t9)



Ejemplo 12.2.7.- Se trata de resolver el sistema de EDO de primer orden:

$$\begin{cases} x' + 2y' - 3x + 4y = 2\sin(t) \\ 2x' + y' + 2x - y = t \end{cases}$$

Escribiremos cada una de las EDO del sistema y asignaremos en cada caso una referencia:

```
(%i1) EDO_2a: 'diff(x(t), t)  -    3*x(t) + 2* 'diff(y(t), t) + 4*y(t)= 2*sin(t) ;
      EDO_2b: 2*'diff(x(t), t) + 2*x(t) +     'diff(y(t), t)  - y(t)    = t         ;
```

(%o1) $2\left(\frac{d}{d\,t}\,y(t)\right)+\frac{d}{d\,t}\,x(t)+4\,y(t)-3\,x(t)=2\,\sin(t)$

(%o2) $\frac{d}{d\,t}\,y(t)+2\left(\frac{d}{d\,t}\,x(t)\right)-y(t)+2\,x(t)=t$

A continuación calcularemos la solución general del sistema:

```
(%i3) desolve(  [EDO_2a, EDO_2b]  , [x(t), y(t)] );
```

(%o3) $[x(t)=-\frac{7\,\sin(t)}{65}-\frac{9\,\cos(t)}{65}+\frac{(45\,y(0)+60\,x(0)+243)\,\%e^{-\frac{t}{3}}}{105}-\frac{(975\,y(0)-975\,x(0)+36)\,\%e^{-5\,t}}{2275}+\frac{4\,t}{5}-\frac{54}{25},\ y(t)=\frac{18\,\sin(t)}{65}-\frac{14\,\cos(t)}{65}+\frac{(45\,y(0)+60\,x(0)+243)\,\%e^{-\frac{t}{3}}}{105}+\frac{(1300\,y(0)-1300\,x(0)+48)\,\%e^{-5\,t}}{2275}+\frac{3\,t}{5}-\frac{53}{25}]$

Para calcular una solución particular, hay que introducir ahora las condiciones iniciales necesarias par hacerlo:

```
(%i4) atvalue(x(t), t=0, 2);
      atvalue(y(t), t=0, 3);
(%o4) 2
(%o5) 3
```

Finalmente calcularemos la solución particular que corresponde a estas condiciones iniciales:

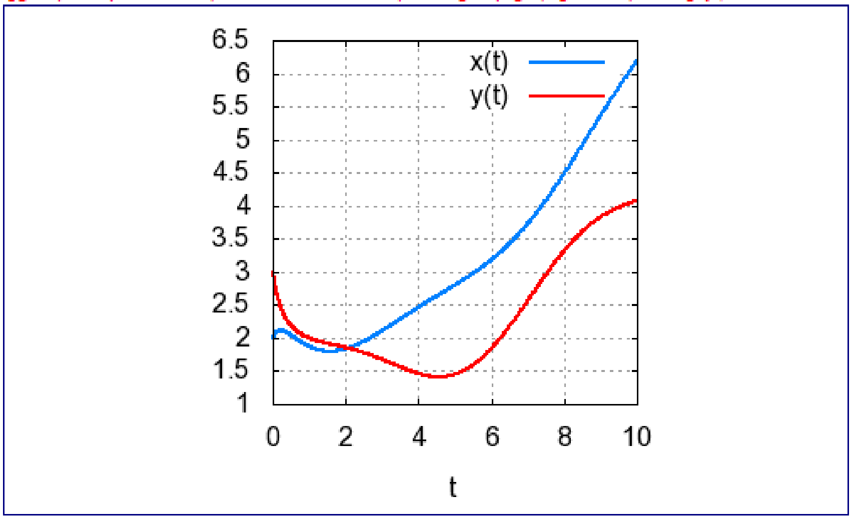
```
(%i6) desolve( [EDO_2a, EDO_2b] , [x(t), y(t)] );
```

(%o6) $\left[x(t)=-\frac{7\sin(t)}{65}-\frac{9\cos(t)}{65}+\frac{166\,\%e^{-\frac{t}{3}}}{35}-\frac{1011\,\%e^{-5t}}{2275}+\frac{4t}{5}-\frac{54}{25},\ y(t)=\frac{18\sin(t)}{65}-\frac{14\cos(t)}{65}+\frac{166\,\%e^{-\frac{t}{3}}}{35}+\frac{1348\,\%e^{-5t}}{2275}+\frac{3t}{5}-\frac{53}{25}\right]$

Como en ejemplos anteriores, ilustramos esta solución particular con una representación gráfica, que tiene como paso previo la definición de las soluciones particulares como funciones:

```
(%i9) wxplot2d([x(t), y(t)], [t, 0, 10],     [legend, "x(t)" , "y(t)"]   , [style, [lines, 2]] ,
      [gnuplot_preamble, "set size ratio 1; set grid;"]  , [nticks, 100] )$
```

(%t9)



12-3.- Ecuaciones diferenciales ordinarias de orden 2

Los contenidos de este apartado se desarrollan en el archivo **Tema_12-3.wxm**.

Una ecuación diferencial ordinaria (EDO) de orden 2 o de segundo orden es una ecuación de la forma $F(t, y, y', y'') = 0$ en la que t indica una variable y y una función de t.

Resolver o integrar una EDO es determinar la familia de curvas que cumple la ecuación.

Se llama solución general o integral general de una EDO de primer orden, una familia de curvas planas de la forma $\Phi(t, y, C_1, C_2) = 0$ que cumple la ecuación diferencial, donde C_1, C_2 son dos parámetros o constantes arbitrarias.

Se llama solución particular o integral particular de una EDO de orden 2, una curva plana que forma parte de la solución general, en la que se han determinado los valores concretos de los parámetros. Para determinar los parámetros hay que tener una información que se llama condiciones iniciales, dadas generalmente mediante

$$\begin{cases} y(t_0) = y_0 \\ y'(t_0) = y'_0 \end{cases}.$$

Como en el apartado anterior, no es nuestro objetivo desarrollar una teoría completa de EDO, sino mostrar la metodología de resolución con wxMaxima. En este caso, se verá la metodología para resolver una EDO de orden 2 y se comentará un tipo de EDO de orden 2 de importancia especial: las EDO lineales de coeficientes constantes.

Ejemplo 12.3.1.- Se quiere resolver la EDO de orden 2

$$y'' - y(y')^3 = 0$$

Veamos cómo se puede resolver con wxMaxima. En primer lugar se introduce la ecuación y se le da una referencia:

```
(%i1) EDO_1: 'diff(y, t, 2) - y*'diff(y, t)^3 = 0;
```

(%o1) $\frac{d^2}{d\,t^2}\,y - y\left(\frac{d}{d\,t}\,y\right)^3 = 0$

A continuación se aplica la instrucción para la resolución de ecuaciones diferenciales:

```
(%i2) ode2(EDO_1, y, t);
```

(%o2) $-\frac{y^3 + 6\,\%k1\,y}{6} = t + \%k2$

Esta es la solución general, que contiene dos parámetros. Para calcular una solución particular aplicaremos unas condiciones iniciales:

```
(%i3) ic2(%, t=0, y=0, 'diff(y,t)=2);
```

$$(\%o3)\ -\frac{y^3-3\,y\,(y^2+1)}{6}=t$$

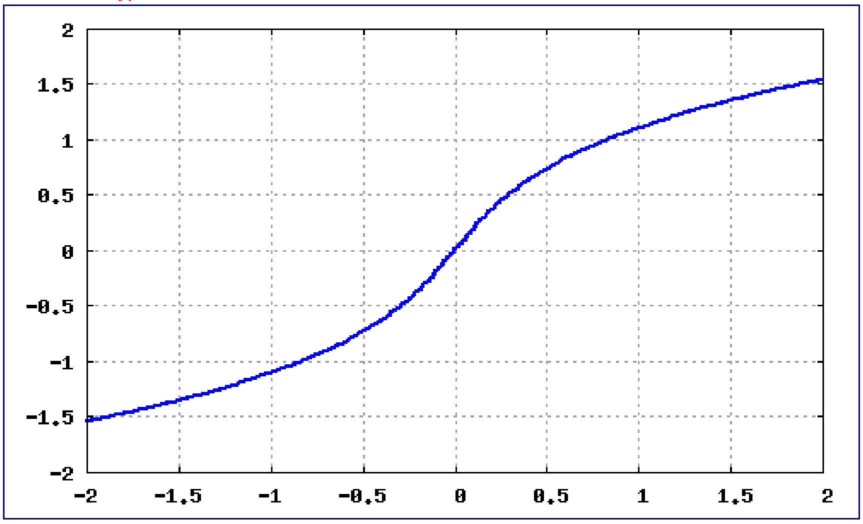
```
(%i4) ratsimp(%);
```

$$(\%o4)\ \frac{2\,y^3+3\,y}{6}=t$$

Ahora podemos obtener una representación gráfica; en este caso podemos hacerlo con la curva dada en forma implícita:

```
(%i5) load(draw)$
      wxdraw2d( grid = true,  line_type = solid,  line_width = 2, color = blue,
                implicit( (2*y^3+3*y)/6=t , t, -2.0, 2.0, y, -2,2)
               );
```

(%t6)



(%o6) [gr2d(implicit)]

Ejemplo 12.3.2.- Se quiere resolver la EDO de orden 2

$$t^2 y'' - ty' + 4y = 0$$

Veamos cómo se puede resolver con wxMaxima. En primer lugar se introduce la ecuación y se le da una referencia:

```
(%i1) EDO_2: t^2 *'diff(y, t, 2) - t * 'diff(y, t) + 4*y = 0;
```

$$(\%o1)\ t^2\left(\frac{d^2}{d\,t^2}\,y\right)-t\left(\frac{d}{d\,t}\,y\right)+4\,y=0$$

A continuación se aplica la instrucción para la resolución de ecuaciones diferenciales:

```
(%i2) ode2(EDO_2, y, t);
```

(%o2) $y = t\,(\%k1\,\sin(\sqrt{3}\,\log(t)) + \%k2\,\cos(\sqrt{3}\,\log(t)))$

Esta es la solución general, que contiene dos parámetros. Para calcular una solución particular aplicaremos unas condiciones iniciales:

```
(%i3) ic2(%, t=1, y=1, 'diff(y,t)=2);
```

(%o3) $y = t\left(\dfrac{\sin(\sqrt{3}\,\log(t))}{\sqrt{3}} + \cos(\sqrt{3}\,\log(t))\right)$

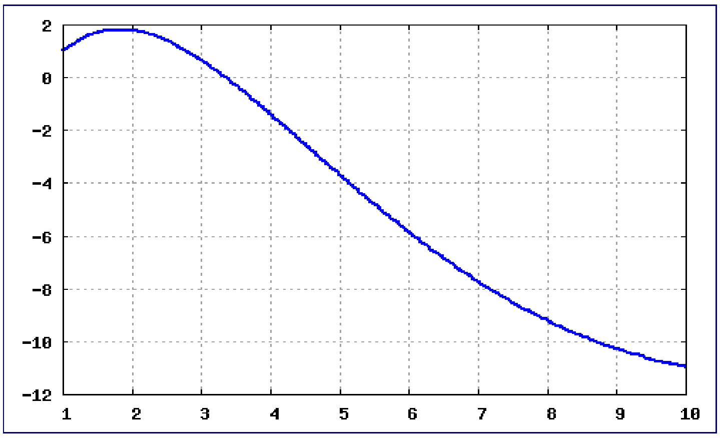
```
(%i4) expand(%);
```

(%o4) $y = \dfrac{t\,\sin(\sqrt{3}\,\log(t))}{\sqrt{3}} + t\,\cos(\sqrt{3}\,\log(t))$

Ahora podemos obtener una representación gráfica; en este caso podemos hacerlo con la curva dada en forma implícita:

```
(%i5) load(draw)$
      wxdraw2d( grid = true,  line_type = solid,  line_width = 2, color = blue,
                implicit( y=(t*sin(sqrt(3)*log(t)))/sqrt(3)+t*cos(sqrt(3)*log(t)) , t, 1, 10, y, -12,2)
              );
rat: replaced 1.732050807568877 by 5042/2911 = 1.732050841635177
   rat: replaced 1.732050807568877 by 5042/2911 = 1.732050841635177
   rat: replaced 1.732050807568877 by 5042/2911 = 1.732050841635177
```

(%t6)



(%o6) [gr2d(implicit)]

Si se prefiere hacer una gráfica con la solución dada en forma explícita primero hay que definirla:

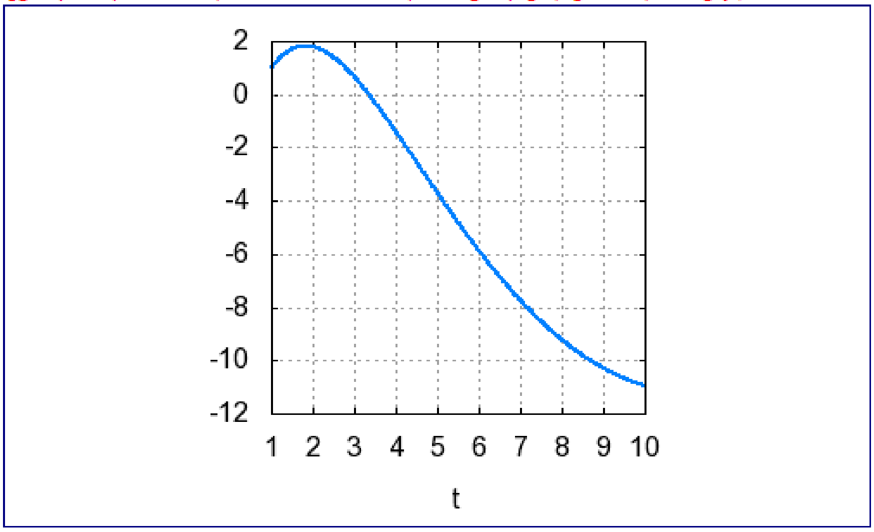
```
(%i7) y(t):=(t*sin(sqrt(3)*log(t)))/sqrt(3)+t*cos(sqrt(3)*log(t));
```

(%o7) $y(t) := \dfrac{t\,\sin(\sqrt{3}\,\log(t))}{\sqrt{3}} + t\,\cos(\sqrt{3}\,\log(t))$

Y ahora ya se puede hacer la representación gráfica:

```
(%i8) wxplot2d([y(t)], [t, 1, 10],   [ylabel, ""] , [style, [lines, 2]] ,
      [gnuplot_preamble, "set size ratio 1; set grid;"]  , [nticks, 100] )$
```

(%t8)



Ejemplo 12.3.3.- Se quiere resolver la EDO de orden 2

$$y'' - \frac{3}{t}y' + \frac{3y}{t^2} = 2t - 1$$

Introducimos la ecuación y le damos una referencia:

```
(%i1) EDO_3: 'diff(y, t, 2) - (3/t) * 'diff(y, t) + (3/t^2)*y = 2*t - 1;
```

(%o1) $\frac{d^2}{d\,t^2}\,y - \frac{3\left(\frac{d}{d\,t}\,y\right)}{t} + \frac{3\,y}{t^2} = 2\,t - 1$

A continuación se aplica la instrucción para la resolución de ecuaciones diferenciales:

```
(%i2) ode2(EDO_3, y, t);
```

(%o2) $y = \frac{2\,t^3\,\log(t) - t^3 + 2\,t^2}{2} + \%k1\,t^3 + \%k2\,t$

```
(%i3) expand(%);
```

(%o3) $y = t^3\,\log(t) + \%k1\,t^3 - \frac{t^3}{2} + t^2 + \%k2\,t$

Esta es la solución general, que contiene dos parámetros. Para calcular una solución particular aplicaremos unas condiciones iniciales:

```
(%i4) ic2(%, t=1, y=1, 'diff(y,t)=2);
```

$$(\%o4)\ y = t^3 \log(t) - \frac{t^3}{2} + t^2 + \frac{t}{2}$$

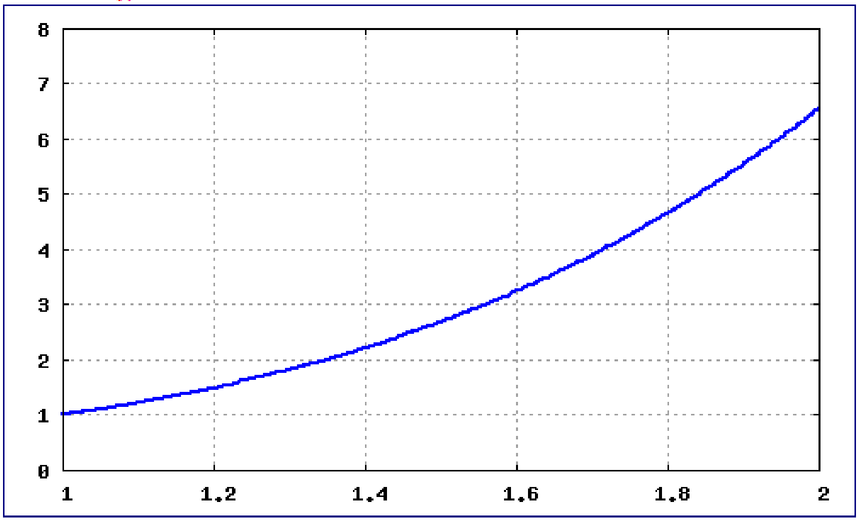
```
(%i5) expand(%);
```

$$(\%o5)\ y = t^3 \log(t) - \frac{t^3}{2} + t^2 + \frac{t}{2}$$

Ahora podemos obtener una representación gráfica; en este caso podemos hacelo con la curva dada en forma implícita:

```
(%i6) load(draw)$
      wxdraw2d( grid = true,  line_type = solid,  line_width = 2, color = blue,
                implicit( y=t^3*log(t)-t^3/2+t^2+t/2 , t, 1, 2, y, 0,8)
      );
```

(%t7)



(%o7) [gr2d(implicit)]

Se trata a continuación de estudiar un tipo concreto de EDO de orden 2: las **ecuaciones diferenciales lineales de segundo orden y de coeficientes constantes**.

Se llaman ecuaciones diferenciales lineales de segundo orden y coeficientes constantes las de la forma:

$$a_2 y'' + a_1 y' + a_0 y = F(t), \quad a_2, a_1, a_0 \in \mathbb{R},\ a_2 \neq 0$$

Si $F(t) = 0$ la ecuación se llama homogénea; si $F(t) \neq 0$ entonces la ecuación se llama completa.

Recordemos que si D indica el operador derivación, es decir, la aplicación lineal que asigna a una función derivable su función derivada, entonces la ecuación se puede escribir:

$$a_2 D^2 y + a_1 Dy + a_0 y = F(t), \quad a_2, a_1, a_0 \in \mathbb{R},\ a_2 \neq 0$$

y también:

$$\left(a_2 D^2 + a_1 D + a_0 I\right) y = L(D) y = F(t), \quad a_2, a_1, a_0 \in \mathbb{R},\ a_2 \neq 0$$

La metodología de resolución de la ecuación diferencial lineal de segundo orden y coeficientes constantes se fundamenta en el resultado que se enuncia a continuación.

Teorema.- Si $y_H(t)$ designa la solución general de la ecuación homogénea $L(D)y = 0$ y $y_P(t)$ es una solución particular de la ecuación completa $L(D)y = F(t)$, entonces la solución general de la ecuación completa es $y(t) = y_H(t) + y_P(t)$.

Por lo tanto, en virtud del teorema anterior, para resolver una ecuación diferencial lineal de segundo orden y coeficientes constantes hay que calcular por un lado la solución general de la ecuación homogénea asociada y por otro calcular una solución particular de la ecuación completa. Esto comporta desarrollar una metodología para la resolución de la ecuación homogénea y otra para calcular una solución particular de la ecuación completa; el método aplicado clásicamente se conoce como método de variación de los parámetros. Con wxMaxima no hace falta aplicar este procedimiento, ya que se obtiene directamente la solución.

Ejemplo 12.3.4.- Se trata de calcular la solución general de la ecuación diferencial

$$y'' + 4y = 12t\,.$$

Introducimos la ecuación y le damos una referencia:

```
(%i1) EDO_4:   diff(y(t), t, 2) + 4*y(t) = 12*t;
```

$$\text{(\%o1)}\ \frac{d^2}{d\,t^2}\,y(t) + 4\,y(t) = 12\,t$$

A continuación se aplica la instrucción para la resolución de ecuaciones diferenciales:

```
(%i2) ode2(EDO_4, y(t), t);
(%o2) y(t)=%k1 sin(2 t)+%k2 cos(2 t)+3 t
```

Esta es la solución general, que contiene dos parámetros. Observamos que los primeros dos términos corresponden a la solución general de la ecuación homogénea asociada; en efecto:

```
(%i3) EDO_5:   diff(y(t), t, 2) + 4*y(t) = 0;
```

$$\text{(\%o3)}\ \frac{d^2}{d\,t^2}\,y(t) + 4\,y(t) = 0$$

```
(%i4) ode2(EDO_5, y(t), t);
```

(%o4) $y(t)=\%k1\ \sin(2\,t)+\%k2\ \cos(2\,t)$

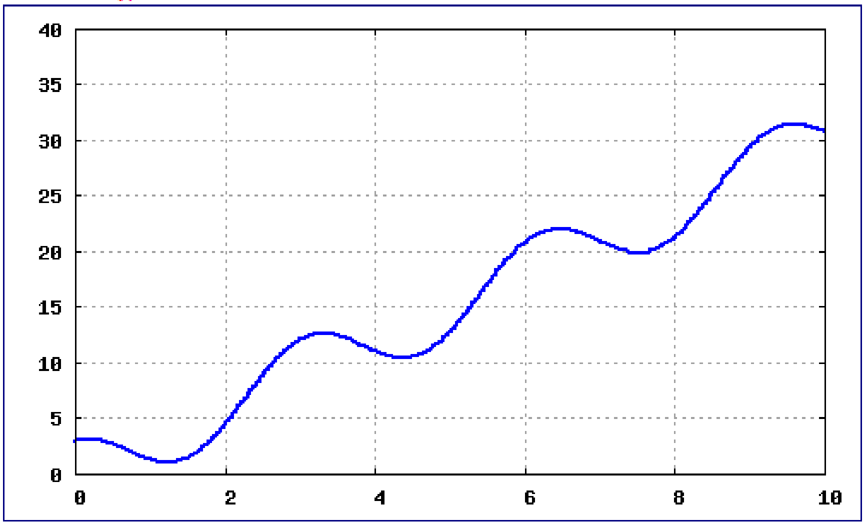
Para calcular una solución particular aplicaremos unas condiciones iniciales:

```
(%i5) ic2(  y=%k1*sin(2*t)+%k2*cos(2*t)+3*t , t=0, y=3, diff(y,t)=2);
```

(%o5) $y=-\frac{\sin(2\,t)}{2}+3\,\cos(2\,t)+3\,t$

Ahora podemos obtener una representación gráfica; en este caso podemos hacerlo con la curva dada en forma implícita:

```
(%i6) load(draw)$
      wxdraw2d( grid = true,  line_type = solid,  line_width = 2, color = blue,
                implicit( y=-sin(2*t)/2+3*cos(2*t)+3*t , t, 0, 10, y, 0,40)
              );
```

(%t7) 

(%o7) [gr2d(implicit)]

12-4.- Introducción a los métodos numéricos de resolución de ecuaciones diferenciales ordinarias y sistemas

Los contenidos de este apartado se desarrollan en el archivo **Tema_12-4.wxm**.

A pesar del gran número de métodos que permiten el cálculo de la solución de una ecuación diferencial ordinaria o de un sistema de ecuaciones diferenciales ordinarias, en muchos casos el cálculo de la solución analítica puede resultar inabordable o, incluso, imposible por la no existencia de solución analítica. En estos casos se puede recurrir a métodos aproximados, los más importantes de los cuales son los métodos numéricos. La potencia de cálculo de programas como wxMaxima proporciona una herramienta muy útil para resolver estas situaciones. En este apartado se describe la metodología y la sintaxis para resolver EDO mediante métodos numéricos, en particular el método programado en wxMaxima que se conoce como método de Runge-Kutta.

La metodología consiste esencialmente en dividir el intervalo $[a,b]$ en el que se quiere calcular la solución mediante un conjunto de puntos $P=\{t_0,t_1,...,t_n\}$ llamado <u>partición</u> en el que $h=t_{k+1}-t_k$ es constante y fijado por el usuario. Esto da lugar a un conjunto de intervalos $[t_k,t_{k+1}]$ en el extremo de los que se aproxima el valor de la función solución particular de la ecuación diferencial. No entraremos en detalles sobre la precisión del método y el cálculo de la cota del error cometido; en este Curso se trata sencillamente de ilustrar la aplicación del método de Runge-Kutta de cuarto orden implementado en wxMaxima, cosa que haremos mediante los ejemplos que siguen.

Ejemplo 12.4.1.- Se trata de calcular la solución particular de la ecuación diferencial

$$t(t-1)y'-(1-2t)y+y^2=2t\,.$$

con la condición inicial $y(0)=2$. Observad que se trata de la ecuación de Ricatti que se ha estudiado en el Ejemplo 12.2.5. Introducimos la ecuación y le damos una referencia:

```
(%i1) EDO: t*(t-1)*diff(y(t), t) - (1-2*t)*y(t) + y(t)^2 = 2*t;
```

$$(\%o1)\ (t-1)\,t\left(\frac{d}{d\,t}y(t)\right)+y(t)^2-(1-2\,t)\,y(t)=2\,t$$

Como se recordará, wxMaxima no puede calcular la solución general de esta ecuación diferencial con la instrucción al efecto, aunque sabemos que esta solución analítica existe. Así, recordemos que la respuesta de wxMaxima es:

```
(%i2) ode2(EDO, y(t), t);
(%o2) false
```

Nos proponemos pues calcular la solución particular de la EDO con el método de Runge-Kutta, para lo cual primero en primer lugar obtendremos de forma explícita la derivada de la función:

```
(%i3) EDO_1b : (t-1)*t*Dy(t)+y(t)^2-(1-2*t)*y(t)=2*t;
```

(%o3) $y(t)^2+(2\,t-1)\,y(t)+(t-1)\,t\,Dy(t)=2\,t$

```
(%i4) solve( [EDO_1b] , [Dy(t)] );
```

(%o4) $\left[Dy(t)=-\dfrac{y(t)^2+(2\,t-1)\,y(t)-2\,t}{t^2-t}\right]$

Veamos ahora la sintaxis del método. Antes limitaremos el número de cifras decimales de las aproximaciones numéricas de la solución:

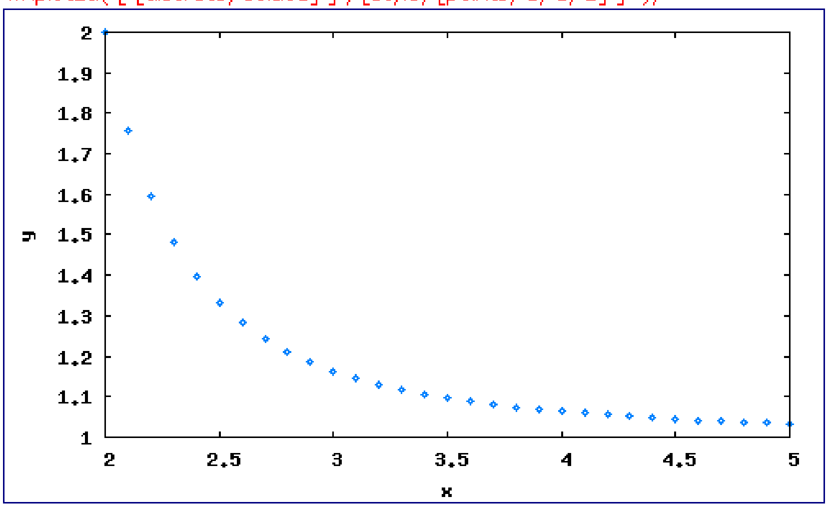
```
(%i5) fpprintprec : 5$
```

```
(%i6) soluc1: rk( [-(y^2+(2*t-1)*y-2*t)/(t^2-t) ], [y], [2] , [t, 2, 5, 0.1] );
(%o6) [[2,2],[2.1,1.7575],[2.2,1.5948],[2.3,1.4799],[2.4,1.3954],[2.5,1.3315],[2.6,1.2819],[2.7,
1.2426],[2.8,1.2109],[2.9,1.1851],[3.0,1.1636],[3.1,1.1457],[3.2,1.1306],[3.3,1.1176],[3.4,1.1065],
3.5,1.0969],[3.6,1.0885],[3.7,1.0812],[3.8,1.0747],[3.9,1.0689],[4.0,1.0638],[4.1,1.0593],[4.2,1.05
],[4.3,1.0515],[4.4,1.0482],[4.5,1.0451],[4.6,1.0424],[4.7,1.0399],[4.8,1.0376],[4.9,1.0355],[5.0,
1.0336]]
```

La solución se da en forma de lista con los pares de valores que corresponden a la variable independiente y la variable dependiente. Precisamente en la instrucción anterior se asigna un nombre a esta lista y esto permite ahora obtener una representación gráfica:

```
(%i6) wxplot2d( [ [discrete, soluc1] ] , [style, [points, 1, 1, 2] ]  );
```

(%t6)

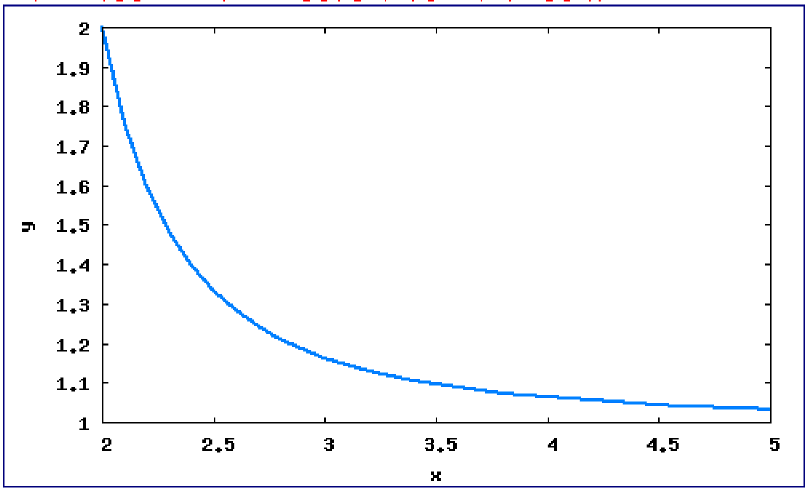


(%o6)

Si se quiere se puede cambiar el formato de la representación gráfica:

```
(%i7) wxplot2d( [ [discrete, soluc1] ] , [style, [lines, 2, 1 ] ]  );
```

(%t7)



(%o7)

Ejemplo 12.4.2.- Se trata de calcular la solución particular de la ecuación diferencial

$$(100 - t^6)y' - \cos(t) = 0\,.$$

con la condición inicial $y(0) = 0.01$.

Introducimos la ecuación y le damos una referencia:

```
(%i1) EDO: (100 - t^6)*diff(y(t), t) - cos(t) = 0;
```

(%o1) $(100 - t^6)\left(\frac{d}{d\,t}y(t)\right) - \cos(t) = 0$

Para obtener la solución establecemos ahora una precisión numérica de los resultados de wxMaxima:

```
(%i2) fpprintprec : 5 $
```

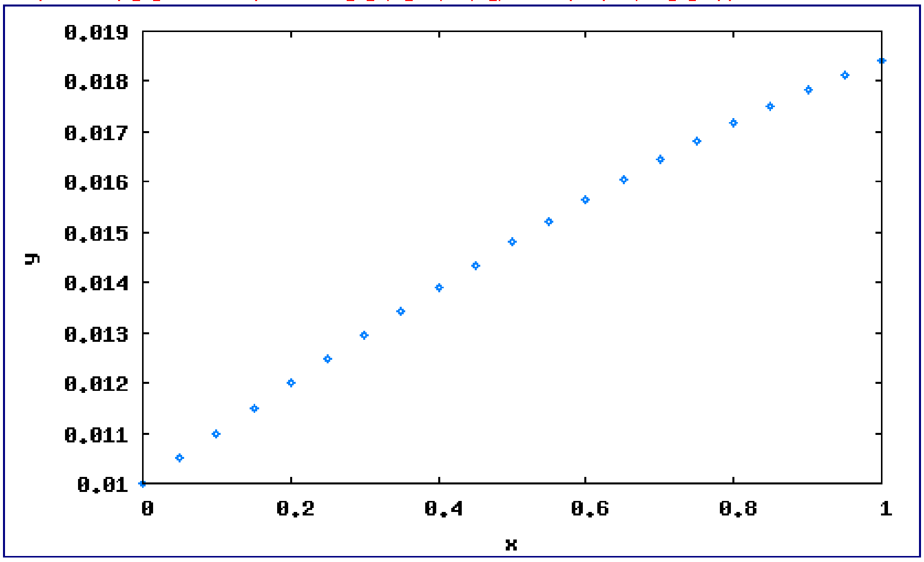
Ahora calcularemos la solución particular con el método de Runge-Kutta, para lo cual hay que tener de forma explícita la derivada de la función, que en este caso es inmediata. Veamos el resultado:

```
(%i4) soluc1: rk( [ cos(t)/(100 - t^6) ], [y], [0.01] , [t, 0, 1,  0.05] );
(%o4) [[0,0.01],[0.05,0.0105],[0.1,0.011],[0.15,0.0115],[0.2,0.012],[0.25,0.0125],[0.3,0.013],[0.35
,0.0134],[0.4,0.0139],[0.45,0.0143],[0.5,0.0148],[0.55,0.0152],[0.6,0.0156],[0.65,0.0161],[0.7,0.016
],[0.75,0.0168],[0.8,0.0172],[0.85,0.0175],[0.9,0.0178],[0.95,0.0181],[1.0,0.0184]]
```

La solución se da en forma de lista con los pares de valores que corresponden a la variable independiente y la variable dependiente. Ahora podemos obtener una representación gráfica:

```
(%i5) wxplot2d( [ [discrete, soluc1] ] , [style, [points, 1, 1, 2] ]  );
```

(%t5)



(%o5)

Ejemplo 12.4.3.- Se trata de calcular la solución particular del sistema de ecuaciones diferenciales:

$$\begin{cases} x_1' = -0.2x_1 + 0.3x_2 \\ x_2' = 0.1x_1 - 0.4x_2 \end{cases}.$$

con las condiciones iniciales $x_1(0) = 1, x_2(0) = 3$.

Introducimos las ecuaciones del sistema:

```
(%i4) edo1:  'diff(x1, t)=   -0.2*x1 +  0.3*x2;
      edo2:  'diff(x2, t)=   0.1*x1 - 0.4*x2;
```

(%o4) $\frac{d}{d\,t}x1=0.3\,x2-0.2\,x1$

(%o5) $\frac{d}{d\,t}x2=0.1\,x1-0.4\,x2$

Ahora introducimos las condiciones iniciales, los puntos inicial y final del intervalo en el que se quiere obtener la solución, el paso o amplitud de los intervalos en que se divide el intervalo anterior y se calcula el número de intervalos:

```
(%i6) x1_0:1;          x2_0:3;
      t_ini=t0:0.0;    t_fin=t1:10.0;        Pas=p:0.5;     n_int=n_int:(t1 - t0)/0.5;
(%o6) 1
 (%o7) 3
 (%o8) t_ini = 0.0
 (%o9) t_fin = 10.0
(%o10) Pas = 0.5
(%o11) n_int = 20.0
```

Para obtener la solución establecemos ahora una precisión numérica de los resultados de wxMaxima:

```
(%i2) fpprintprec : 5 $
```

Ahora calcularemos la solución particular con el método de Runge-Kutta, para lo cual hay que tener de forma explícita la derivada de cada una de las funciones, que en este caso es la forma en que se ha expresado el sistema. Veamos el resultado:

```
(%i13) solucio_LV: rk([ - 0.2*x1 + 0.3*x2 , 0.1*x1 - 0.4*x2], [x1, x2], [x1_0, x2_0], [t, t0, t1, p]);
(%o13) [[0.0,1,3],[0.5,1.2961,2.5088],[1.0,1.5014,2.1179],[1.5,1.6374,1.8055],[2.0,1.7204,1.5545],[
2.5,1.7634,1.3518],[3.0,1.7762,1.1871],[3.5,1.7665,1.0523],[4.0,1.7403,0.941],[4.5,1.7021,0.848],[5.
1.6554,0.771],[5.5,1.603,0.705],[6.0,1.5468,0.648],[6.5,1.4886,0.6],[7.0,1.4294,0.557],[7.5,1.3701,
0.519],[8.0,1.3113,0.486],[8.5,1.2537,0.456],[9.0,1.1975,0.429],[9.5,1.1429,0.404],[10.0,1.0902,0.38
]]
```

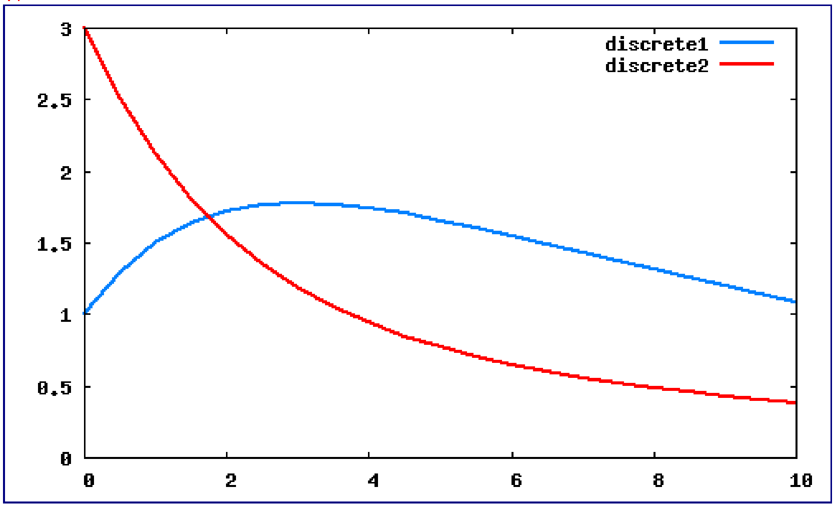
La solución se da en forma de lista con los ternas de valores que corresponden a la variable independiente y a cada una de las variables dependientes. Esto obliga a hacer algunas operaciones para obtener las listas correspondientes a cada una de las variables. Los detalles de la sintaxis se justificarán en el Apéndice dedicado a Álgebra Lineal.

```
(%i14) M1:matrix(
        [0.0,1,3],[0.5,1.2961,2.5088],[1.0,1.5014,2.1179],[1.5,1.6374,1.8055],[2.0,1.7204,1.5545],
        [2.5,1.7634,1.3518],[3.0,1.7762,1.1871],[3.5,1.7665,1.0523],[4.0,1.7403,0.941],
        [4.5,1.7021,0.848],[5.0,1.6554,0.771],[5.5,1.603,0.705],[6.0,1.5468,0.648],[6.5,1.4886,0.6],
        [7.0,1.4294,0.557],[7.5,1.3701,0.519],[8.0,1.3113,0.486],[8.5,1.2537,0.456],
        [9.0,1.1975,0.429],[9.5,1.1429,0.404],[10.0,1.0902,0.381]
        )$
(%i15) M1T:transpose(M1)$
        T:first(M1T)$      Y1:second(M1T)$      Y2:third(M1T)$
(%i19) List1:makelist([T[n],Y1[n]],  n, 1, 1 + n_int)$
        List2:makelist([T[n],Y2[n]],  n, 1, 1 + n_int)$
```

Ahora se puede obtener la representación gráfica de cada una de las listas, es decir, de cada una de las soluciones del sistema:

```
(%i21) wxplot2d(
       [ [discrete, List1] , [discrete, List2] ] ,
       [style, [lines, 2, 1] , [lines, 2, 2] , [lines, 2, 3] ]
       );
```

(%t21)



(%o21)

Veamos a continuación un clásico de los sistemas de EDO: las ecuaciones de Lotka-Volterra que expresan la relación entre el número de individuos de un ecosistema predador y presa. En este caso N1 indica el número de presas (liebres) y N2 indica el número de predadores (linces). Los valores de los parámetros de las ecuaciones se obtienen a partir de datos experimentales por regresión multivariante.

Ejemplo 12.4.4.- Se trata de calcular la solución particular del sistema de ecuaciones diferenciales:

$$\begin{cases} N_1' = a_1 N_1 - a_2 N_1 N_2 \\ N_2' = -b_1 N_1 + b_2 N_1 N_2 \end{cases}$$

con las condiciones iniciales $N_1(0) = 30, N_2(0) = 6$.

Introducimos las ecuaciones del sistema:

```
(%i4) edo1:'diff(N1, t)=   a1*N1 -  a2*N1*N2;
      edo2:'diff(N2, t)= - b1*N2 + b2*N1*N2;
```

(%o4) $\frac{d}{d\,t} N1 = a1\, N1 - a2\, N1\, N2$

(%o5) $\frac{d}{d\,t} N2 = b2\, N1\, N2 - b1\, N2$

Ahora introducimos las condiciones iniciales y los parámetros de las ecuaciones diferenciales:

```
(%i6) N1_0:30;        N2_0:6;
       a1:0.5654;     a2:0.0245;     b1:0.8267;     b2:0.0219;
(%o6) 30
 (%o7) 6
 (%o8) 0.565
 (%o9) 0.0245
(%o10) 0.827
(%o11) 0.0219
```

Ahora introducimos los puntos inicial y final del intervalo en el que se quiere obtener la solución, el paso o amplitud de los intervalos en que se divide el intervalo anterior y se calcula el número de intervalos:

```
(%i12) t_ini=t0:0.0;    t_fin=t1:20.0;     Pas=p:0.5;    n_int=n_int:(t1 - t0)/0.5;
(%o12) t_ini=0.0
 (%o13) t_fin=20.0
 (%o14) Pas=0.5
 (%o15) n_int=40.0
```

Para obtener la solución establecemos ahora una precisión numérica de los resultados de wxMaxima:

```
(%i2) fpprintprec : 5 $
```

Ahora calcularemos la solución particular con el método de Runge-Kutta, para lo cual hay que tener de forma explícita la derivada de cada una de las funciones, que en este caso es la forma en que se ha expresado el sistema. Veamos el resultado:

```
(%i17) solucio_LV: rk([ a1*N1 -  a2*N1*N2, - b1*N2 + b2*N1*N2], [N1, N2], [N1_0, N2_0], [t, t0, t1, p]);
(%o17) [[0.0,30,6],[0.5,37.062,5.721],[1.0,45.805,5.9473],[1.5,56.232,6.8679],[2.0,67.82,8.9548],[
 2.5,78.782,13.243],[3.0,84.883,21.613],[3.5,79.734,35.646],[4.0,61.802,51.611],[4.5,40.965,59.755],
 5.0,26.386,56.753],[5.5,18.405,47.734],[6.0,14.477,37.697],[6.5,12.794,28.905],[7.0,12.455,21.932],
 7.5,13.065,16.669],[8.0,14.488,12.811],[8.5,16.727,10.046],[9.0,19.871,8.1114],[9.5,24.073,6.8174],
 10.0,29.533,6.0399],[10.5,36.474,5.7259],[11.0,45.087,5.9101],[11.5,55.396,6.7669],[12.0,66.937,
 8.7396],[12.5,78.055,12.808],[13.0,84.734,20.793],[13.5,80.615,34.424],[14.0,63.416,50.564],[14.5,
 42.366,59.56],[15.0,27.223,57.265],[15.5,18.835,48.491],[16.0,14.678,38.423],[16.5,12.867,29.504],[
 17.0,12.445,22.393],[17.5,12.991,17.011],[18.0,14.355,13.06],[18.5,16.529,10.222],[19.0,19.601,8.23
 ],[19.5,23.717,6.8958],[20.0,29.075,6.0823]]
```

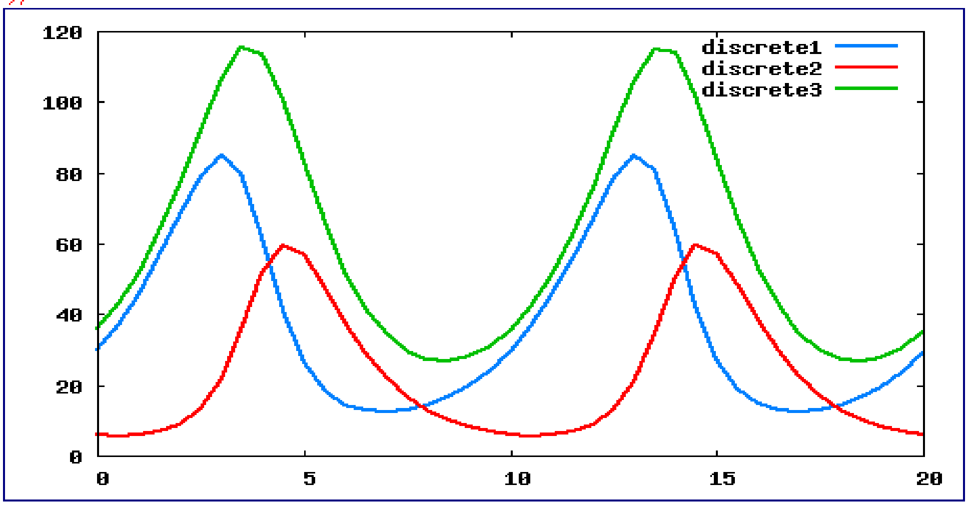
La solución se da en forma de lista con las ternas de valores que corresponden a la variable independiente y a cada una de las variables dependientes. Esto obliga a hacer las operaciones que se han comentado en el ejemplo anterior:

```
(%i3) M1:matrix(
      [0.0,30,6],[0.5,37.062,5.721],[1.0,45.805,5.9473],[1.5,56.232,6.8679],[2.0,67.82,8.9548],
      [2.5,78.782,13.243],[3.0,84.883,21.613],[3.5,79.734,35.646],[4.0,61.802,51.611],[4.5,40.965,59.755],
      [5.0,26.386,56.753],[5.5,18.405,47.734],[6.0,14.477,37.697],[6.5,12.794,28.905],[7.0,12.455,21.932],
      [7.5,13.065,16.669],[8.0,14.488,12.811],[8.5,16.727,10.046],[9.0,19.871,8.1114],[9.5,24.073,6.8174],
      [10.0,29.533,6.0399],[10.5,36.474,5.7259],[11.0,45.087,5.9101],[11.5,55.396,6.7669],[12.0,66.937,8.7396],
      [12.5,78.055,12.808],[13.0,84.734,20.793],[13.5,80.615,34.424],[14.0,63.416,50.564],[14.5,42.366,59.56],
      [15.0,27.223,57.265],[15.5,18.835,48.491],[16.0,14.678,38.423],[16.5,12.867,29.504],[17.0,12.445,22.393],
      [17.5,12.991,17.011],[18.0,14.355,13.06],[18.5,16.529,10.222],[19.0,19.601,8.2328],[19.5,23.717,6.8958],
      [20.0,29.075,6.0823]
      )$
(%i19) M1T:transpose(M1)$
       T:first(M1T)$     Y1:second(M1T)$     Y2:third(M1T)$     M:Y1+Y2$
(%i24) List1:makelist([T[n],Y1[n]],  n, 1, 1 + n_int)$
       List2:makelist([T[n],Y2[n]],  n, 1, 1 + n_int)$
       List3:makelist([T[n],M[n]],  n, 1, 1 + n_int)$
```

Ahora se puede obtener la representación gráfica de cada una de las listas, es decir, de cada una de las soluciones del sistema:

```
(%i27) wxplot2d(
       [ [discrete, List1] , [discrete, List2] , [discrete, List3] ] ,
       [style, [lines, 2, 1] , [lines, 2, 2] , [lines, 2, 3] ]
       );
```

(%t27)



(%o27)