



DEPARTAMENTO DE CIENCIAS BASICAS CICLO I/19

GUIA 4 DE LABORATORIO EN MATLAB PARA METODOS NUMERICOS SOBRE ECUACIONES DIFERENCIALES

Para obtener la solución exacta de ecuaciones diferenciales, Matlab emplea el comando “***dsolve***”. Este comando proporciona soluciones simbólicas de ecuaciones diferenciales ordinarias y además se utiliza para resolver sistemas de ecuaciones diferenciales. Las ecuaciones son especificadas por expresiones simbólicas conteniendo la letra “***D***” para denotar diferenciación de orden 1, o las expresiones: “***D2***”, “***D3***”, ..., etc, para denotar diferenciación de orden 2, 3, ..., etc. A continuación de la letra “***D***” se sitúa la variable dependiente (que suele ser Y) y cualquier letra no precedida por “***D***” es un candidato a variable independiente. Si no se especifica la variable independiente, por defecto Matlab emplea la variable “***t***”.

Se pueden especificar condiciones iniciales en ecuaciones adicionales, mediante la forma $y(a)=b$ ó $Dy(a)=b$, ..., etc. Si las condiciones iniciales no se especifican, las soluciones de las ecuaciones diferenciales contendrán constantes de integración C1, C2, ..., etc. Para obtener el valor numérico exacto se debe auxiliar del comando “***subs***”.

Además se podría generar programas basándose en las ecuaciones que definen los diferentes métodos para resolver aproximadamente los valores de la solución de ecuaciones diferenciales; además Matlab posee comandos de la familia “***ODE***” que nos permiten obtener la solución numérica de ecuaciones diferenciales así como de sistemas de ecuaciones diferenciales.

A continuación se presenta la sintaxis de los comandos:

dsolve('ecuación','v') Resuelve la ecuación diferencial siendo “v” la variable independiente (si no se especifica “v”, la variable independiente es “ t “)

dsolve('ecuación','condición_inicial','v') Resuelve la ecuación diferencial sujeta a la condición inicial especificada

dsolve('ecuación','condición1','condición2',...,'condiciónN','v')
Resuelve la ecuación diferencial sujeta a las condiciones iniciales especificadas.

dsolve('ecuación','condición1,condición2,...,condiciónN','v') Resuelve la ecuación diferencial sujeta a las condiciones iniciales especificadas.

dsolve('ecu1','ecu2',...,'ecuN','cond1','cond2',...,'condN','v') Resuelve el sistema diferencial sujeto a las condiciones iniciales especificadas.

dsolve('ecu1,ecu2,...,ecuN','cond1,cond2,...,condN','v') Resuelve el sistema diferencial sujeto a las condiciones iniciales especificadas.

Solver	Tipo Del Problema	Orden de la exactitud	Cuándo utilizar
ode45	No Rígido	Medio	La mayoría del tiempo. Éste debe ser el primer solver que usted debe intentar.
ode23	No Rígido	Bajo	Si usa tolerancias crudas del error o para solucionar problemas moderado rígidos.
ode113	No Rígido	Bajo a Alto	Si usa tolerancias rigurosas del error
ode15s	Rígido	Bajo a Medio	Si ode45 es lento o hay una matriz total.
ode23s	Rígido	Bajo	Si usa tolerancias crudas del error para solucionar sistemas tiesos o allí existe una matriz total constante.
ode23t	Moderadamente Rígido	Bajo	Si el problema es solamente moderado rígido
ode23tb	Rígido	Bajo	Si usa tolerancias crudas del error para solucionar sistemas rígidos o allí existe una matriz total.

Solve differential equations

Syntax

[T,Y] = solver('F',tspan,y0)

[T,Y] = solver('F',tspan,y0,options)

[T,Y] = solver('F',tspan,y0,options,p1,p2...)

[T,Y,TE,YE,IE] = solver('F',tspan,y0,options)

F	Name of the ODE file, a MATLAB function of t and y returning a column vector. All solvers can solve systems of equations in the form $y' = F(t, y)$. ode15s, ode23s, ode23t, and ode23tb can solve equations of the form $My' = F(t, y)$. Of these four solvers all but ode23s can solve equations in the form $M(t)y' = F(t, y)$
tspan	A vector specifying the interval of integration [t0 tfinal]. To obtain solutions at specific times (all increasing or all decreasing), use tspan = [t0,t1, ..., tfinal].
y0	A vector of initial conditions.
options	Optional integration argument created using the odeset function. See odeset for details.
p1,p2...	Optional parameters to be passed to F.
T,Y	Solution matrix Y, where each row corresponds to a time returned in column vector T.

Description

[T,Y] = solver('F',tspan,y0) with tspan = [t0 tfinal] integrates the system of differential equations $y' = F(t,y)$ from time t0 to tfinal with initial conditions y0. 'F' is a string containing the name of an ODE file. Function F(t,y) must return a column vector. Each row in solution array y corresponds to a time returned in column vector t. To obtain solutions at the specific times t0, t1, . . . , tfinal (all increasing or all decreasing), use tspan = [t0 t1 ... tfinal].

[T,Y] = solver('F',tspan,y0,options) solves as above with default integration parameters replaced by property values specified in options, an argument created with the odeset function (see odeset for details). Commonly used properties include a scalar relative error tolerance RelTol (1e-3 by default) and a vector of absolute error tolerances AbsTol (all components 1e-6 by default).

[T,Y] = solver('F',tspan,y0,options,p1,p2...) solves as above, passing the additional parameters p1,p2... to the M-file F, whenever it is called. Use options = [] as a place holder if no options are set.

[T,Y,TE,YE,IE] = solver('F',tspan,y0,options) with the Events property in options set to 'on', solves as above while also locating zero crossings of an event function defined in the ODE file. The ODE file must be coded so that $F(t,y,'events')$ returns appropriate information. See odefile for details. Output TE is a column vector of times at which events occur, rows of YE are the corresponding solutions, and indices in vector IE specify which event occurred.

When called with no output arguments, the solvers call the default output function odeplot to plot the solution as it is computed. An alternate method is to set the OutputFcn property to 'odeplot'. Set the OutputFcn property to 'odephas2' or 'odephas3' for two- or three-dimensional phase plane plotting. See odefile for details.

The solvers of the ODE suite can solve problems of the form $M(t, y) y' = F(t, y)$ with a mass matrix M that is nonsingular and (usually) sparse. Use odeset to set Mass to 'M', 'M(t)', or 'M(t,y)' if the ODE file F.m is coded so that $F(t,y,'mass')$ returns a constant, time-dependent, or time-and-state-dependent mass matrix, respectively. The default value of Mass is 'none'. The ode23s solver can only solve problems with a constant mass matrix M . For examples of mass matrix problems, see fem1ode, fem2ode, or batonode.

For the stiff solvers ode15s, ode23s, ode23t, and ode23tb the Jacobian matrix $\partial F / \partial y$ is critical to reliability and efficiency so there are special options. Set JConstant to 'on' if $\partial F / \partial y$ is constant. Set Vectorized to 'on' if the ODE file is coded so that $F(t,[y1\ y2\ \dots])$ returns $[F(t,y1)\ F(t,y2)\ \dots]$. Set JPattern to 'on' if $\partial F / \partial y$ is a sparse matrix and the ODE file is coded so that $F([],[],'jpattern')$ returns a sparsity pattern matrix of 1's and 0's showing the nonzeros of $\partial F / \partial y$. Set Jacobian to 'on' if the ODE file is coded so that $F(t,y,'jacobian')$ returns $\partial F / \partial y$.

If M is singular, then $M(t) * y' = F(t, y)$ is a differential algebraic equation (DAE). DAEs have solutions only when $y0$ is consistent, that is, if there is a vector $yp0$ such that $M(t0) * yp0 = f(t0, y0)$. The ode15s and ode23t solvers can solve DAEs of index 1 provided that M is not state dependent and $y0$ is sufficiently close to being consistent. If there is a mass matrix, you can use odeset to set the MassSingular property to 'yes', 'no', or 'maybe'. The default value of 'maybe' causes the solver to test whether the problem is a DAE. If it is, the solver treats $y0$ as a guess, attempts to compute consistent initial conditions that are close to $y0$, and continues to solve the problem. When solving DAEs, it is very advantageous to formulate the problem so that M is a diagonal matrix (a semi-explicit DAE). For examples of DAE problems, see hb1dae or amp1dae.

Algorithms

ode45 is based on an explicit Runge-Kutta (4,5) formula, the Dormand-Prince pair. It is a *one-step* solver - in computing $y(t_n)$, it needs only the solution at the immediately preceding time point, $y(t_{n-1})$. In general, ode45 is the best function to apply as a "first try" for most problems.

ode23 is an implementation of an explicit Runge-Kutta (2,3) pair of Bogacki and Shampine. It may be more efficient than ode45 at crude tolerances and in the presence of moderate stiffness. Like ode45, ode23 is a one-step solver.

ode113 is a variable order Adams-Bashforth-Moulton PECE solver. It may be more efficient than ode45 at stringent tolerances and when the ODE function is particularly expensive to evaluate. ode113 is a *multistep* solver - it normally needs the solutions at several preceding time points to compute the current solution.

The above algorithms are intended to solve non-stiff systems. If they appear to be unduly slow, try using one of the stiff solvers below.

ode15s is a variable order solver based on the numerical differentiation formulas, NDFs. Optionally, it uses the backward differentiation formulas, BDFs (also known as Gear's method) that are usually less efficient. Like ode113, ode15s is a multistep solver. If you suspect that a problem is stiff or if ode45 has failed or was very inefficient, try ode15s.

ode23s is based on a modified Rosenbrock formula of order 2. Because it is a one-step solver, it may be more efficient than ode15s at crude tolerances. It can solve some kinds of stiff problems for which ode15s is not effective.

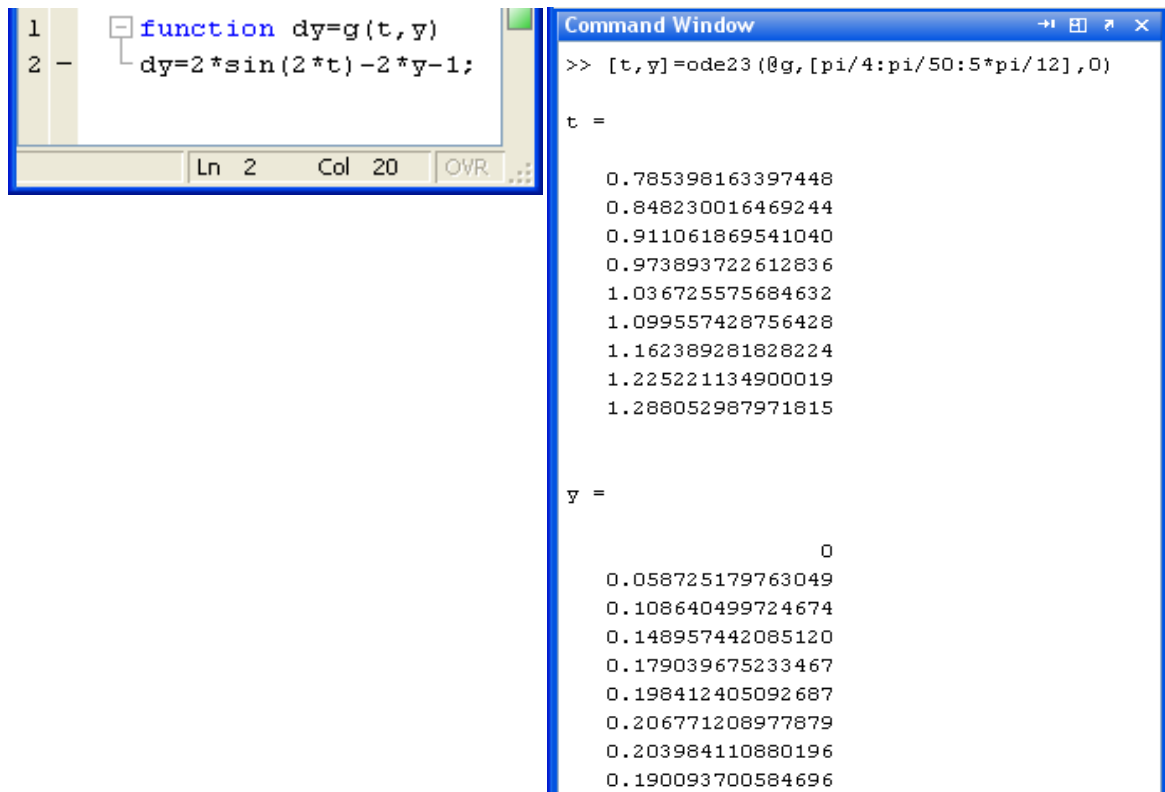
ode23t is an implementation of the trapezoidal rule using a "free" interpolant. Use this solver if the problem is only moderately stiff and you need a solution without numerical damping.

ode23tb is an implementation of TR-BDF2, an implicit Runge-Kutta formula with a first stage that is a trapezoidal rule step and a second stage that is a backward differentiation formula of order two. By construction, the same iteration matrix is used in evaluating both stages. Like ode23s, this solver may be more efficient than ode15s at crude tolerances

Ejemplo:

Obtenga la solución aproximada y exacta de la ecuación diferencial:

$$2y + y' = 2\sin(2t) - 1, \quad y\left(\frac{\pi}{4}\right) = 0, \quad \frac{\pi}{4} \leq t \leq \frac{5\pi}{12}, \quad h = \frac{\pi}{50}$$



The image shows a MATLAB Editor window on the left and a Command Window on the right. The Editor window contains a function definition for `dy=g(t,y)` with the equation `dy=2*sin(2*t)-2*y-1;`. The Command Window shows the execution of `[t,y]=ode23(@g,[pi/4:pi/50:5*pi/12],0)`, resulting in two vectors: `t` and `y`. The `t` vector contains 11 values ranging from approximately 0.785 to 1.288. The `y` vector contains 11 values ranging from 0 to approximately 0.190.

```

1 function dy=g(t,y)
2 -   dy=2*sin(2*t)-2*y-1;

Ln 2   Col 20   OVR

Command Window

>> [t,y]=ode23(@g,[pi/4:pi/50:5*pi/12],0)

t =

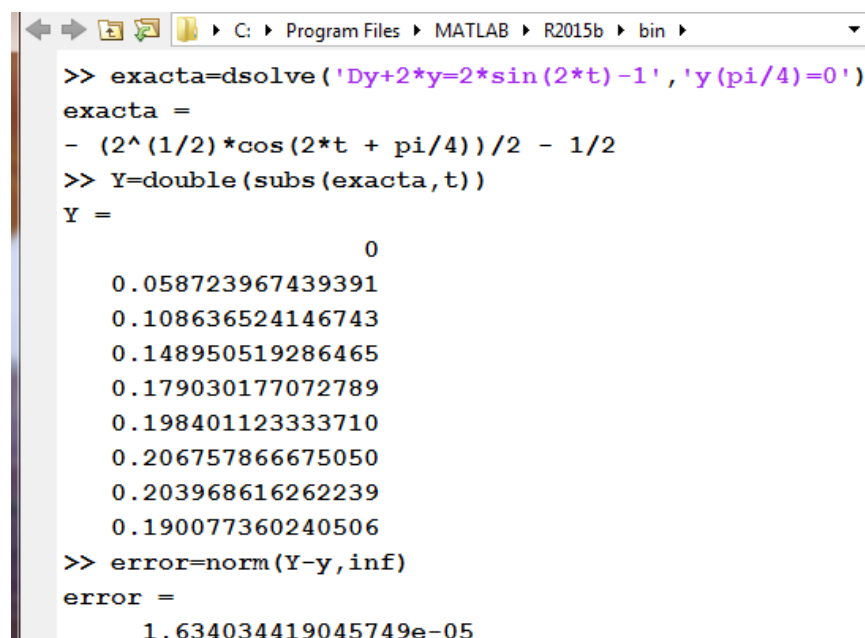
    0.785398163397448
    0.848230016469244
    0.911061869541040
    0.973893722612836
    1.036725575684632
    1.099557428756428
    1.162389281828224
    1.225221134900019
    1.288052987971815

y =

         0
    0.058725179763049
    0.108640499724674
    0.148957442085120
    0.179039675233467
    0.198412405092687
    0.206771208977879
    0.203984110880196
    0.190093700584696

```

Para obtener la solución exacta de la ecuación diferencial y luego evaluarla en el intervalo de valores y obtener el error entre el valor exacto y el aproximado se realiza lo siguiente:



The image shows a MATLAB Command Window with the following commands and results. The exact solution is calculated using `dsolve` with the equation `Dy+2*y=2*sin(2*t)-1` and the initial condition `y(pi/4)=0`. The exact solution is then converted to a double precision array using `double(subs(exacta,t))`. The error between the exact and numerical solutions is calculated using `norm(Y-y,inf)`, resulting in a value of `1.634034419045749e-05`.

```

>> exacta=dsolve('Dy+2*y=2*sin(2*t)-1','y(pi/4)=0')
exacta =
- (2^(1/2)*cos(2*t + pi/4))/2 - 1/2
>> Y=double(subs(exacta,t))
Y =

         0
    0.058723967439391
    0.108636524146743
    0.148950519286465
    0.179030177072789
    0.198401123333710
    0.206757866675050
    0.203968616262239
    0.190077360240506
>> error=norm(Y-y,inf)
error =
    1.634034419045749e-05

```

Ejemplo:

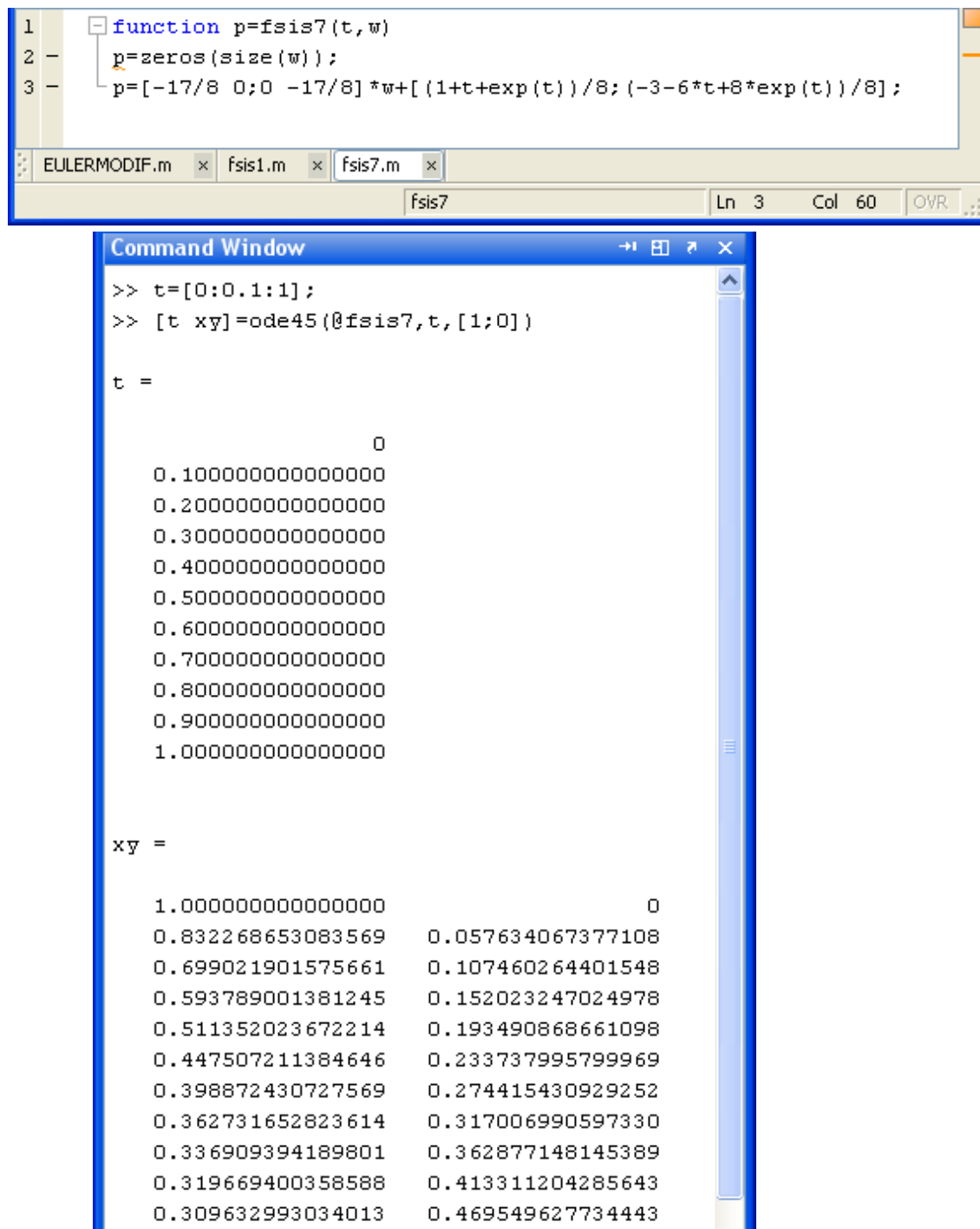
Resuelva el siguiente sistema de ecuaciones diferenciales:

$$3x' + 5x + y' - 2y = t$$

$$3x' + 6x + y' + y = e^t$$

Sujeto a las condiciones iniciales: $x(0) = 1$, $y(0) = 0$, $0 \leq t \leq 1$, $h = 0.1$

La solución aproximada se obtiene así:



The image shows a MATLAB Editor window with a function file named `fsis7.m` and a Command Window displaying the execution results.

Editor Window:

```

1 function p=fsis7(t,w)
2     p=zeros(size(w));
3     p=[-17/8 0;0 -17/8]*w+[(1+t+exp(t))/8;(-3-6*t+8*exp(t))/8];

```

Command Window:

```

>> t=[0:0.1:1];
>> [t xy]=ode45(@fsis7,t,[1;0])

t =

    0
    0.100000000000000
    0.200000000000000
    0.300000000000000
    0.400000000000000
    0.500000000000000
    0.600000000000000
    0.700000000000000
    0.800000000000000
    0.900000000000000
    1.000000000000000

xy =

    1.000000000000000    0
    0.832268653083569    0.057634067377108
    0.699021901575661    0.107460264401548
    0.593789001381245    0.152023247024978
    0.511352023672214    0.193490868661098
    0.447507211384646    0.233737995799969
    0.398872430727569    0.274415430929252
    0.362731652823614    0.317006990597330
    0.336909394189801    0.362877148145389
    0.319669400358588    0.413311204285643
    0.309632993034013    0.469549627734443

```

La solución exacta se obtiene así:

```

C:\Program Files\MATLAB\R2015b\bin
>> [X Y]=dsolve('Dx+17/8*x=(1+t+exp(t))/8','Dy+17/8*y=(-3-6*t+8*exp(t))/8','x(0)=1','y(0)=0')
X =
exp(-(17*t)/8)*((exp((17*t)/8)*(425*t + 289*exp(t) + 225))/7225 + 6711/7225)
Y =
-exp(-(17*t)/8)*((exp((17*t)/8)*(2550*t - 2312*exp(t) + 75))/7225 + 2237/7225)
>> exacta_x=double(subs(X,t));exacta_y=double(subs(Y,t));
>> exacta=[exacta_x exacta_y]
exacta =
    1.000000000000000    0
    0.832268882789914    0.057634011761911
    0.699022065759819    0.107460230800117
    0.593789116860702    0.152023230238434
    0.511352103242385    0.193490864799628
    0.447507264744140    0.233738001985329
    0.398872465196778    0.274415445064577
    0.362731673901843    0.317007011189544
    0.336909405986467    0.362877174168667
    0.319669405926289    0.413311235076887
    0.309632954804515    0.469549624551510

>> error_x=norm(exacta_x-xy(:,1));error_y=norm(exacta_y-xy(:,2));
>> error=[error_x error_y]
error =
    1.0e-06 *
    0.324807125686613    0.082561209921228
fx >>

```

Otra manera de construir el archivo m para resolver sistemas de ecuaciones diferenciales, se muestra a continuación:

Ejemplo:

Resuelva el siguiente sistema de ecuaciones diferenciales:

$$\begin{aligned} \mathbf{x}' &= -4\mathbf{x} - 2\mathbf{y} + \cos(t) + 4\sin(t) \\ \mathbf{y}' &= 3\mathbf{x} + \mathbf{y} - 3\sin(t) \end{aligned}$$

Sujeto a las condiciones iniciales: $\mathbf{x}(0) = \mathbf{0}$, $\mathbf{y}(0) = -1$, $0 \leq t \leq 0.2$, $h = 0.02$

La solución aproximada se obtiene así:

```

1 function dy=prueba1(t,xy)
2 -   dy=[-4*xy(1)-2*xy(2)+cos(t)+4*sin(t);3*xy(1)+xy(2)-3*sin(t)];

```

Otra manera que construir el archivo m es:

```

1 function P=prueba1(t,xy)
2   P=zeros(2,1);
3   P(1)=-4*xy(1)-2*xy(2)+cos(t)+4*sin(t);
4   P(2)=3*xy(1)+xy(2)-3*sin(t);

```



```
>> [t xy]=ode45(@prueba1,[0:1/50:1/5],[0;-1])
t =
    0
    0.0200000000000000
    0.0400000000000000
    0.0600000000000000
    0.0800000000000000
    0.1000000000000000
    0.1200000000000000
    0.1400000000000000
    0.1600000000000000
    0.1800000000000000
    0.2000000000000000
xy =
    0 -1.0000000000000000
    0.058817135109624 -1.019017141721234
    0.115335519819712 -1.036135624783606
    0.169652200309932 -1.051452727412958
    0.221859808901238 -1.065061461316940
    0.272046746649207 -1.077050748036591
    0.320297358672036 -1.087505588098542
    0.366692102608056 -1.096507223360906
    0.411307710472947 -1.104133292823206
    0.454217344176206 -1.110457982159960
    0.495490744864665 -1.115552167147740
```

La solución exacta se muestra a continuación:

```
>> [X Y]=dsolve('Dx=-4*x-2*y+cos(t)+4*sin(t)','Dy=3*x+y-3*sin(t)','x(0)=0','y(0)=-1')
X =
2*exp(-t) - 2*exp(-2*t) + sin(t)
Y =
2*exp(-2*t) - 3*exp(-t)
>> Exacta=[double(subs(X,t)) double(subs(Y,t))]
Exacta =
    0 -1.000000000000000
    0.058817135002197 -1.019017141615620
    0.115335519718009 -1.036135624683698
    0.169652200213627 -1.051452727318431
    0.221859808810022 -1.065061461227485
    0.272046746562784 -1.077050747951915
    0.320297358590128 -1.087505588018366
    0.366692102530397 -1.096507223284966
    0.411307710399287 -1.104133292751252
    0.454217344106306 -1.110457982091754
    0.495490744879746 -1.115552167162667
>> error=[norm(Exacta(:,1)-xy(:,1),inf) norm(Exacta(:,2)-xy(:,2),inf)]
error =
    1.0e-09 *
    0.107427219897538    0.105614850198776
```

A continuación se muestra un programa para aproximar la solución numérica de una ecuación diferencial sujeta a una condición inicial y luego se indica la manera de correr el programa:

Ejemplo:

Resuelva $y' - y = \cos(t)$, si $t \in [0, 0.45]$, $y(0) = 0$, $h = 0.05$

```

1 % este programa resuelve una ecuación diferencial empleando
2 % el método de Euler Modificado
3 % f es la función definida
4 % a y b son los extremos del intervalo
5 % ya es el valor inicial
6 % h es el tamaño del paso
7 function E=eulermodif(a,b,ya,h)
8 - warning('off','MATLAB:dispatcher:InexactCaseMatch')
9 - syms t y
10 - f=y+cos(t);
11 - m=(b-a)/h;
12 - T=a:h:b;
13 - Y(1)=ya;
14 - for j=1:m
15 -     Y(j+1)=Y(j)+0.5*h*subs(f,{t,y},{T(j),Y(j)})+0.5*h*subs(f,{t,y},{T(j+1),Y(j)+h*subs(f,{t,y},{T(j),Y(j)})});
16 - end
17 - E=[T' Y'];

```

eulermodif Ln 8 Col 52 OVR

Command Window

```

>> E=eulermodif(0,0.45,0,0.05)

E =

    0.0500000000000000    0.051218756509874
    0.1000000000000000    0.104936016248324
    0.1500000000000000    0.161152123368000
    0.2000000000000000    0.219668074932462
    0.2500000000000000    0.281085871983849
    0.3000000000000000    0.344808886221066
    0.3500000000000000    0.411042242300627
    0.4000000000000000    0.479793215780850
    0.4500000000000000    0.551071646741011

>> exacta_y=dsolve('Dy=y+cos(t)','y(0)=0')

exacta_y =

-1/2*cos(t)+1/2*sin(t)+1/2*exp(t)

>> Y=subs(exacta_y,E(:,1))

Y =

    0
    0.051250002625868
    0.105000084722225
    0.161250648632920
    0.220002755556995
    0.281258477115810
    0.345021262555868
    0.411296321600665
    0.480091022973518
    0.551415308624361

>> error=norm(Y-E(:,2),inf)

error =

    3.436618833497773e-004

```

Otra alternativa de construir el archivo M sería así:

Resuelva $t y' - 2 y = t^2$ si $t \in [1, 29/20]$, $y(1) = -2$, $h = 0.05$

```

1 % este programa resuelve una ecuación diferencial empleando
2 % el método de Euler Modificado
3 % f es la función definida entre comillas
4 % g es la ecuación diferencial entre comillas
5 % e es la condición inicial entre comillas
6 % a y b son los extremos del intervalo
7 % ya es el valor inicial
8 % h es el tamaño del paso
9 function E=eulermodif2(f,g,e,a,b,ya,h)
10 - syms t y
11 - m=(b-a)/h;
12 - T=a:h:b;
13 - F=dsolve(g,e)
14 - w(1)=ya;
15 - for j=1:m+1
16 -     w(j+1)=w(j)+0.5*h*subs(f,{t,y},{T(j),w(j)})+0.5*h*subs(f,{t,y},{T(j+1),w(j)+h*subs(f,{t,y},{T(j),w(j)})});
17 - end
18 - Y=double(subs(F,t,T));
19 - fprintf('          T          Wi+1          y(t)');
20 - fprintf('\n');
21 - E=[T'          w'          Y'];

```

```

>> E=eulermodif2(' (2*y+t^2)/t ','Dy=(2*y+t^2)/t ','y(1)=-2',1,29/20,-2,0.05)
F =
(log(t)-2)*t^2
          T          Wi+1          y(t)
E =
1.000000000000000 -2.000000000000000 -2.000000000000000
1.050000000000000 -2.151130952380953 -2.151208844003201
1.100000000000000 -2.304520330859617 -2.304674682436767
1.150000000000000 -2.459935262777407 -2.460164831208852
1.200000000000000 -2.617153248060185 -2.617456958216705
1.250000000000000 -2.775961274145300 -2.776338201071547
1.300000000000000 -2.936155039575609 -2.936604393049940
1.350000000000000 -3.097538269171989 -3.098059380259258
1.400000000000000 -3.259922106927020 -3.260514416222422
1.450000000000000 -3.423124575287480 -3.423787622600704
>> Error=norm(E(:,3)-E(:,2),inf)
Error =
6.630473132238635e-004

```

A continuación se muestra la solución de una ecuación diferencial de orden superior.

Ejemplo:

Resuelva $y'''(t) + 2y''(t) - y'(t) - 2y(t) = e^t$, sujeta a las condiciones
 iniciales: $y(0) = 1$, $y'(0) = 2$, $y''(0) = 0$ $0 \leq t \leq 0.5$, $h = 0.05$

El archivo m podrá construirse de la siguiente manera:

```

1 function p=fsis8(t,w)
2   p=zeros(size(w));
3   p=[0 1 0;0 0 1;2 1 -2]*w+[0;0;exp(t)];

```

fsis8 Ln 3 Col 39 OVR

Otra manera de construir el archivo m es:

```

File Edit Text Go Cell Tools Debug Desktop Window Help
- 1.0 + ÷ 1.1 x % % %
1 function P=fsis8(t,w)
2   P=zeros(3,1);
3   P(1)=w(2);
4   P(2)=w(3);
5   P(3)=2*w(1)+w(2)-2*w(3)+exp(t);

```

La solución será la siguiente:

```

Command Window
>> t=[0:0.05:0.5];
>> [t yuv]=ode45(@fsis8,t,[1;2;0])

t =

    0
    0.0500000000000000
    0.1000000000000000
    0.1500000000000000
    0.2000000000000000
    0.2500000000000000
    0.3000000000000000
    0.3500000000000000
    0.4000000000000000
    0.4500000000000000
    0.5000000000000000

yuv =

    1.000000000000000    2.000000000000000    0
    1.100102908146528    2.006149929469028    0.244076722778290
    1.200813800504212    2.024231243077646    0.477563107959995
    1.302716766110813    2.053759441287003    0.702237224088614
    1.406373833634899    2.094335640965305    0.919748028841181
    1.512329099636054    2.145640487297548    1.131630028771720
    1.621112567617126    2.207428770713224    1.339316657950715
    1.733243734157360    2.279524683674346    1.544152514053876
    1.849234952740841    2.361817663148154    1.747404568106366
    1.969594603326567    2.454258770128001    1.950272453423229
    2.094830092201776    2.556857563624596    2.153897926279912

```

La solución exacta será:

```

Command Window
>> exacta_y=dsolve('D3y+2*D2y-Dy-2*y=exp(t)', 'y(0)=1', 'Dy(0)=2', 'D2y(0)=0')

exacta_y =

1/6*t*exp(t)+43/36*exp(t)-4/9*exp(-2*t)+1/4*exp(-t)

>> Y=subs(exacta_y,t)

Y =

1.0000000000000000
1.100102905694803
1.200813798365949
1.302716764241379
1.406373831994753
1.512329098190139
1.621112566334333
1.733243733010049
1.849234951704414
1.969594602379086
2.094830092524348

>> error_y=norm(Y-yuv(:,1),inf)

error_y =

2.451724823160362e-009

>>

```

RESUELVA LOS SIGUIENTES EJERCICIOS:

1. Obtenga la solución **exacta y aproximada** para la ecuación diferencial: $y' + y \sec(t) = \sec(t)$, $0 \leq t \leq \frac{1}{40}$, $y(0) = 4$, con $h = \frac{1}{400}$. Para la solución aproximada emplee el comando **ode45**, el método de **Runge-Kutta-Fehlberg de cuarto orden**, el método de **Heun**.

2. Obtenga la solución **exacta y aproximada** para la ecuación diferencial: $(t^2 + 2y^2) dt = ty dy$, $-1 \leq t \leq -0.85$, $y(-1) = 1$, $h = \frac{3}{200}$.

Para la solución aproximada emplee el comando **ode23**, el método de **Euler**, el método de **Heun**.

3. Obtenga la solución **exacta y aproximada** para la ecuación diferencial: $(t + 1)y' + y = \ln(t)$, $1 \leq t \leq \frac{11}{10}$, $y(1) = 10$, con $h = \frac{1}{100}$.

Para la solución aproximada emplee el comando **ode113**, el método de **Adams-Bashforth de cuatro pasos**, el método de **Runge-Kutta-Fehlberg de quinto orden**.

4. Obtenga la solución **exacta y aproximada** para la ecuación diferencial: $dy + \tan(t)ydt = \cos^2(t)dt$, $0 \leq t \leq \frac{1}{60}$, $y(0) = -1$, $h = \frac{1}{600}$.

Para la solución aproximada emplee el comando **ode45**, el método de **Taylor de orden seis**, el método de **Euler modificado**.

5. Obtenga la solución **exacta y aproximada** para la ecuación diferencial: $y' = 1 + (t - y)^2$, $2 \leq t \leq \frac{101}{50}$, $y(2) = 1$, con $h = \frac{1}{500}$.

Para la solución aproximada emplee el comando **ode23**, el método de **Adams-Bashforth de tres pasos**, el método de **Taylor de orden cinco**.

6. Obtenga la solución **exacta y aproximada** para el siguiente sistema de ecuaciones diferenciales:

$$3x' - 5x + 3y' - 2y = \frac{7}{3}e^t - \frac{1}{3}e^{4t} + 8, \quad x(0) = 2$$

$$4x' + x + 4y' + 7y = \frac{185}{3}e^t - \frac{1}{6}e^{4t} - 28, \quad y(0) = \frac{1}{2}$$

Para los siguientes valores: $0 \leq t \leq \frac{3}{50}$, con $h = \frac{1}{100}$

Para la solución aproximada emplee el comando **ode45**, el método de **Runge-Kutta de orden cuatro**.

7. Obtenga la solución **exacta y aproximada** para el siguiente sistema de ecuaciones diferenciales:

$$x' + 3x + y' = 1$$

$$x' - x + y' - y = e^t, \quad \text{sujeto a : } x_{(0)} = 0, \quad y_{(0)} = 0$$

Para los siguientes valores: $0 \leq t \leq \frac{1}{20}$, **considere** $h = \frac{1}{200}$

Para la solución aproximada emplee el comando **ode23**, el método de **Runge-Kutta de orden cuatro**.

8. Obtenga la solución **exacta y aproximada** para el siguiente sistema de ecuaciones diferenciales:

$$4x' + 4x + y' = 2$$

$$4x' + 2x + y' - y = e^t, \quad \text{sujeto a : } x_{(0)} = 2, \quad y_{(0)} = 1$$

Para los siguientes valores: $0 \leq t \leq \frac{3}{50}$ **considere** $h = \frac{1}{200}$

Para la solución aproximada emplee el comando **ode45**, el método de **Runge-Kutta de orden cuatro**.

9. Obtenga la solución **exacta y aproximada** de cada una de las siguientes ecuaciones diferenciales:

a) $y'' - 2y' + y = te^t - t$, sujeta a $y_{(0)} = 0$, $y'_{(0)} = 0$, para los valores $0 \leq t \leq \frac{1}{10}$ considere $h = \frac{1}{80}$.

b) $y''' + 2y'' - y' - 2y = te^t$, sujeta a: $y_{(0)} = 2$, $y'_{(0)} = 1$, $y''_{(0)} = -1$, para los valores $0 \leq t \leq \frac{2}{15}$ considere $h = \frac{1}{60}$.

c) $t^3 y''' - t^2 y'' + 3ty' - 4y = 5t^3 \ln(t)$, sujeta a: $y_{(1)} = 0$, $y'_{(1)} = 1$, $y''_{(1)} = 3$ para los valores $1 \leq t \leq \frac{6}{5}$ considere $h = \frac{1}{50}$

d) $y''' + 2y'' - y' - 2y = e^t$, sujeta a: $y(0) = 1$, $y'(0) = 2$, $y''(0) = 0$,

para los valores $0 \leq t \leq \frac{3}{40}$, considere $h = \frac{1}{80}$

Para la solución aproximada emplee el comando **ode45**, el método de **Runge-Kutta de orden cuatro**.

10. Determine la carga en el capacitor y la corriente en un circuito RLC en serie, si se sabe que: $L=0.5H$, $R=20\Omega$, $C=0.02F$, $E(t)=25V$, $q(0)=0$ **Coulombs**, $i(0)=2A$, para los valores de t siguientes: $0 \leq t \leq 0.15$, con $h=\frac{1}{20}$. Para la solución aproximada emplee el comando **ode45**, el método de **Runge-Kutta de orden cuatro**, además obtenga la **solución exacta**.
11. Obtenga la solución **exacta y aproximada** para la ecuación diferencial: $y' = e^{3t+2y}$, $0 \leq t \leq \frac{2}{75}$, $y(0) = 0$, con $h = \frac{1}{300}$. Para la solución aproximada emplee el comando **ode23**, el método de **Taylor de orden cuatro**, el método de **Heun**.
12. Obtenga la solución **exacta y aproximada** para la ecuación diferencial: $(\cos(t)\sin(t) - ty^2)dt + y(1 - t^2)dy = 0$, $0 \leq t \leq \frac{3}{50}$, $y(0) = 2$, con $h = \frac{1}{200}$. Para la solución aproximada emplee el comando **ode45**, el método de **Adams-Bashforth de cinco pasos**, el método de **Runge-Kutta- Fehlberg de quinto orden**.
13. Obtenga la solución **exacta y aproximada** para la ecuación diferencial: $4t + 3y + 3(t + y^2)y' = 0$, $3 \leq t \leq \frac{123}{40}$, $y(3) = -1.57$, considere $h = \frac{1}{200}$. Para la solución aproximada emplee el comando **ode45**, el método de **Adams-Bashforth de cinco pasos**, el método de **Euler Modificado**.

14. Obtenga la solución **exacta y aproximada** para la ecuación diferencial: $\left(\frac{y}{t-1}\right) dt + (\ln(t-1) + 2y)dy = 0$, $2 \leq t \leq \frac{21}{10}$, $y_{(2)} = 4$, con $h = \frac{1}{100}$. Para la solución aproximada emplee el comando **ode45**, el método del **Punto Medio**, el método de **Runge-Kutta- Fehlberg de quinto orden**.
15. Obtenga la solución **exacta y aproximada** para la ecuación diferencial: $(y^2 + 3ty)dt = (4t^2 + ty)dy$, $1 \leq t \leq \frac{13}{10}$, $y_{(1)} = 1$, con $h = \frac{1}{50}$. Para la solución aproximada emplee el comando **ode113**, el método de **Runge-Kutta de orden cuatro**, el método del **Euler**.
16. Obtenga la solución **exacta y aproximada** para la ecuación diferencial: $y' = \frac{2y}{t} + \left(\frac{2y}{t}\right)^2$, $1 \leq t \leq \frac{41}{40}$, $y_{(1)} = 2$, con $h = \frac{1}{200}$. Para la solución aproximada emplee el comando **ode45**, el método de **Taylor de orden cinco**, el método del **Punto Medio**.
17. Obtenga la solución **exacta y aproximada** para la ecuación diferencial: $y' = \frac{ty + y^2}{t^2}$, $1 \leq t \leq \frac{31}{30}$, $y_{(1)} = 2$, con $h = \frac{1}{300}$. Para la solución aproximada emplee el comando **ode23**, el método de **Euler Modificado**, el método de **Heun**.
18. Dada la siguiente ecuación diferencial:
- $$y' - e^y \operatorname{sen}(t) = 0, \quad 0 \leq t \leq \frac{3}{20}, \quad y_{(0)} = 0, \quad \text{con } h = \frac{1}{80}$$
- a) Emplee el **Método de Taylor de orden seis** y el comando **ode23**, para obtener la solución aproximada en los valores de "t".
- b) Además aproxime el valor de $y_{(0.035)}$, mediante **Neville**.

19. Dada la siguiente ecuación diferencial:

$$t^2 dy + (y^2 + ty)dt = 0, \quad 1 \leq t \leq \frac{49}{45}, \quad y(1) = 1, \quad \text{con } h = \frac{1}{90}$$

- a) Determine la solución aproximada para los valores de “t”, empleando el **Método de Runge-Kutta de orden cuatro** y el comando **ode45**.
- b) Además obtenga la solución exacta en los valores de “t”.

20. Dada la siguiente ecuación diferencial:

$$(1 + \ln t + y/t)dt = (1 - \ln t)dy, \quad 1 \leq t \leq \frac{41}{40}, \quad y(1) = 2, \quad \text{con } h = \frac{1}{400}$$

- a) Emplee el **Método de Adams-Bashforth de cuatro pasos**, para obtener la solución aproximada en los valores de “t”, los valores iniciales obténgalos mediante la solución exacta. Además emplee el comando **ode113**.
- b) Obtenga la solución exacta y evalúela en los valores de “t”.

21. Dada la siguiente ecuación diferencial:

$$dy - y \tan(t)dt = 2e^t dt, \quad 0 \leq t \leq \frac{1}{25}, \quad y(0) = 2, \quad \text{con } h = \frac{1}{200}$$

- a) Emplee el **Método de Runge-Kutta-Fehlberg de cuarto orden**, para obtener la solución aproximada en los valores de “t”. Además emplee el comando **ode45**.
- b) Obtenga la solución exacta y evalúela en cada valor de “t”.

22. Obtenga la solución **exacta y aproximada** para el siguiente sistema diferencial:

$$5x' + x + 5y' + 2y = 2e^{-2t} + e^{-t} + \sin(t) + 5\cos(t), \quad x(0) = 0$$

$$4x' - 3x + 4y' - y = 4e^{-2t} + e^{-t} - 3\sin(t) + 4\cos(t), \quad y(0) = -1$$

Para los siguientes valores $0 \leq t \leq \frac{3}{10}$, con $h = \frac{1}{40}$

Para la solución aproximada emplee el comando **ode45**, el método de **Runge-Kutta de orden cuatro**.

23. Emplee el **Método de Extrapolación**, con una tolerancia 10^{-12} , para obtener la solución aproximada de la siguiente ecuación diferencial:

$$(t^2 + 4)dy - tydt = 0, \quad 0 \leq t \leq \frac{1}{40}, \quad y(0) = 3, \quad \text{con } h = \frac{1}{200}$$

Además emplee el comando **ode45** y también obtenga la solución exacta.

24. Emplee el **Método de Extrapolación**, con una tolerancia 10^{-9} , para obtener la solución aproximada de la siguiente ecuación diferencial:

$$dy + y \tan(t)dt = \cos^2(t)dt, \quad 0 \leq t \leq \frac{2}{25}, \quad y(0) = -1, \quad \text{con } h = \frac{1}{100}$$

Además emplee el comando **ode23** y también obtenga la solución exacta.

25. Emplee el **Método de Extrapolación**, con una tolerancia 10^{-12} , para obtener la solución aproximada de la siguiente ecuación diferencial:

$$dy - 4t^2dt = 4dt, \quad \frac{\pi}{4} \leq t \leq \frac{3\pi}{10}, \quad y\left(\frac{\pi}{4}\right) = 1, \quad \text{con } h = \frac{\pi}{100}$$

Además emplee el comando **ode45** y también obtenga la solución exacta.

26. Un acumulador de **12V** se conecta a un circuito en serie LR, con una inductancia de **0.5H** y una resistencia de **10Ω**. Si la corriente inicial es cero, determine la corriente en el inductor para los siguientes valores de t : $0 \leq t \leq \frac{2}{5}$, considere $h = \frac{1}{20}$. Para la solución aproximada emplee el comando **ode113**, el método de **Runge-Kutta de orden cuatro**, además obtenga la **solución exacta**.

27. Si se sabe que la rapidez con que se enfría un objeto es proporcional a la diferencia entre su temperatura y la del medio que lo rodea, determine la temperatura de un pastel, para los siguientes valores de t : $0 \leq t \leq 2$, considere $h = \frac{1}{5}$, se sabe que al sacarlo del horno tiene una temperatura de 300°F. Al cabo de 3 minutos, su temperatura es de 200°F, la temperatura ambiente es de 70°F. Para la solución aproximada emplee el comando **ode45**, el método de **Runge-Kutta de orden cuatro**, además obtenga la **solución exacta**.

28. Suponga que un alumno es portador del virus de la gripe y regresa a su escuela, donde hay 1000 estudiantes. Si se supone que la razón con que se propaga el virus es proporcional no sólo a la cantidad 'y' de alumnos infectados, sino también a la cantidad de alumnos no infectados, determine la cantidad de alumnos infectados al cabo de t días: $0 \leq t \leq 6$, considere $h = \frac{1}{2}$. Además se observa que al cabo de cuatro días el número de infectados es igual a 50. Para la solución aproximada emplee el comando **ode45**, el método de **Runge-Kutta de orden cuatro**, además obtenga la **solución exacta**.

29. Dada la siguiente ecuación diferencial:

$$y' = 1 + \frac{y}{t} + \left(\frac{y}{t}\right)^2, \quad 1 \leq t \leq \frac{41}{40}, \quad y(1) = 2, \quad \text{con } h = \frac{1}{200}$$

- a) Emplee el Método de **Taylor de orden cinco**, para obtener la solución aproximada en los valores de “t”, además emplee **ode45**.
- b) Además aproxime el valor de $y(\frac{511}{500})$ mediante **Hermite**.

30 Dada la siguiente ecuación diferencial:

$$y' + 2y = 2\text{sen}(2t) - 1, \quad \frac{\pi}{4} \leq t \leq \frac{11\pi}{40}, \quad y(\frac{\pi}{4}) = 0, \quad \text{con } h = \frac{\pi}{160}$$

- a) Emplee el Método de **Extrapolación**, con una tolerancia 10^{-12} , para obtener la solución aproximada en los valores de “t”, además emplee **ode45**.
- b) Además obtenga la solución exacta para los valores de “t”.

31 Dada la siguiente ecuación diferencial:

$$(t + ye^{y/t})dt - te^{y/t} dy = 0, \quad 1 \leq t \leq \frac{43}{40}, \quad y(1) = 0, \quad \text{con } h = \frac{1}{80}$$

- a) Emplee el Método de **Runge-Kutta-Fehlberg de quinto orden**, para obtener la solución aproximada en cada uno de los valores de “t”, además emplee **ode45**.
- b) Obtenga la solución exacta y evalúela en cada valor de “t”.

32 Dada la siguiente ecuación diferencial:

$$(2t \tan(y) + 5)dt + (t^2 \sec^2(y))dy = 0, \quad \frac{1}{2} \leq t \leq \frac{13}{25}, \quad y(\frac{1}{2}) = \frac{\pi}{4}, \quad \text{con } h = \frac{1}{500}$$

- a) Emplee el Método de **Adams-Bashforth de cinco pasos**, para obtener la solución aproximada en los valores de “t”, además emplee **ode23**.
- b) Obtenga la solución exacta y evalúela en los valores de “t”.

33 Dada la siguiente ecuación diferencial:

$$(y - te^{t/y})dy + ye^{t/y} dt = 0, \quad 0 \leq t \leq \frac{1}{50}, \quad y(0) = 1, \quad \text{con } h = \frac{1}{200}$$

- a) Emplee el **Método de Extrapolación**, con una tolerancia 10^{-12} , para obtener la solución aproximada en los valores de “t”.
- b) Además obtenga la solución exacta para los valores de “t”.

34 Dada la siguiente ecuación diferencial:

$$e^{4t^2}(y^2 + 1)dy - tydt = 0, \quad 0 \leq t \leq \frac{7}{800}, \quad y(0) = 1, \quad \text{con } h = \frac{1}{800}$$

- Emplee el **Método de Adams-Bashforth de cuatro pasos y el ode45**, para obtener la solución aproximada en los valores de “t”.
- Obtenga la solución exacta y evalúela en los valores de “t”.

35 Dada la siguiente ecuación diferencial:

$$(e^{2y} - y\cos(ty))dt + (2te^{2y} - t\cos(ty) + 2y)dy = 0, \quad \frac{\pi}{2} \leq t \leq \frac{80\ln}{1600}, \quad y\left(\frac{\pi}{2}\right) = 1, \\ \text{con } h = \frac{\pi}{6400}$$

- Emplee el **Método de Extrapolación**, con una tolerancia 10^{-12} , para obtener la solución aproximada en los valores de “t”.
- Además obtenga la solución exacta para los valores de “t”.

36 Dada la siguiente ecuación diferencial:

$$(2t \tan(y) + 5)dt + (t^2 \sec^2(y))dy = 0, \quad \frac{1}{2} \leq t \leq \frac{51}{100}, \quad y(1/2) = \frac{\pi}{4}, \quad \text{con } h = \frac{1}{800}$$

- Emplee el **Método de Adams-Bashforth de tres pasos y el ode23**, para obtener la solución aproximada en los valores de “t”.
- Obtenga la solución exacta y evalúela en los valores de “t”.

37 Dado el siguiente sistema diferencial:

$$\begin{aligned} 8x' + 6x + 2y - 5y' &= 66e^t \cos(t) - 22e^t \sin(t), & x(0) &= 5 \\ 7x' - 5x - 7y + 9y' &= 7e^t \cos(t) - 64e^t \sin(t), & y(0) &= 3 \end{aligned}$$

Determine la solución aproximada para $0 \leq t \leq \frac{3}{50}$, con $h = \frac{1}{100}$

Para la solución aproximada emplee el comando **ode45**, el método de **Runge-Kutta de orden cuatro**.

38 Dado el siguiente sistema diferencial:

$$4x' - 5x + 7y + 4y' = 24e^{-2t} - 27e^{-t} + 4\cos(t) - 5\sin(t), \quad x(0) = 0$$

$$7x' - 8x - 9y + 7y' = -2e^{-2t} + 18e^{-t} + 7\cos(t) - 8\sin(t), \quad y(0) = -1$$

Determine la solución aproximada para $0 \leq t \leq \frac{2}{75}$, con $h = \frac{1}{300}$

Para la solución aproximada emplee el comando **ode45**, el método de **Runge-Kutta de orden cuatro**.

39 Dado el siguiente sistema diferencial:

$$9x' + 2x - 5y + 9y' = -120\cos(3t) - \frac{56}{3}\sin(3t), \quad x(0) = -1$$

$$7x' - 3x + 2y + 7y' = -77\cos(3t) - \frac{62}{3}\sin(3t), \quad y(0) = 2$$

Determine la solución aproximada para $0 \leq t \leq \frac{1}{50}$, con $h = \frac{1}{400}$

Para la solución aproximada emplee el comando **ode45**, el método de **Runge-Kutta de orden cuatro**.

40 Dada la siguiente ecuación diferencial:

$$t dy - y dt = t \sec\left(\frac{y}{t}\right) dt, \quad 2 \leq t \leq \frac{21}{10}, \quad y(2) = \frac{\pi}{3}, \quad \text{con } h = \frac{1}{100}$$

- Emplee el Método de **Adams-Bashforth de cuatro pasos y el ode45**, para obtener la solución aproximada en los valores de “t”, los valores iniciales obténgalos mediante la solución exacta.
- Obtenga la solución exacta y evalúela en cada valor de “t”.

41 Obtenga la solución exacta y aproximada de la siguiente ecuación diferencial:

$y^{IV} - 4y''' - 7y'' + 22y' + 24y = 6\sin 3t - 80te^{3t} + 12e^{3t}$, sujeta a las condiciones: $y(0)=12$, $y'(0)=17$, $y''(0)=26$, $y'''(0)=41$, para cada uno de los valores $0 \leq t \leq \frac{1}{120}$ considere $h = \frac{1}{600}$. Para la **solución aproximada** emplee el comando **ode45**, el método de **Runge-Kutta de orden cuatro**.

42 Obtenga la **solución exacta y aproximada** de la siguiente ecuación diferencial: $y''' - 2y'' + y' = 2 - 24e^t + 40e^{5t}$, sujeta a: $y(0) = \frac{1}{2}$, $y'(0) = \frac{5}{2}$, $y''(0) = -\frac{9}{2}$, para cada uno de los valores $0 \leq t \leq \frac{3}{20}$ considere $h = \frac{1}{40}$. Para la **solución aproximada** emplee el comando **ode23**, el método de **Runge-Kutta de orden cuatro**.

43 Emplee el Método de Extrapolación, con una tolerancia 10^{-12} , para obtener la solución aproximada de la siguiente ecuación diferencial:

$$2ye^t dy + y^2 e^t dt = t e^t dt, \quad 0 \leq t \leq \frac{1}{50}, \quad y(0) = 1, \quad \text{con } h = \frac{1}{200}$$

44 Dado el siguiente sistema diferencial:

$$9x' - 5x + 9y' + 7y = \frac{-2001}{50}e^{2t} + \frac{13696}{175}e^{7t} + 52t + \frac{342}{7}, \quad x(0) = \frac{1}{2}$$

$$4x' + 7x + 4y' + 3y = \frac{-667}{50}e^{2t} + \frac{7062}{175}e^{7t} + 4t + \frac{114}{7}, \quad y(0) = \frac{1}{5}$$

Determine la solución aproximada para $0 \leq t \leq \frac{1}{50}$, con $h = \frac{1}{200}$

Para la solución aproximada emplee el comando **ode45**, el método de **Runge-Kutta de orden cuatro**.