Solución Numérica de Ecuaciones Diferenciales Ordinarias usando la función ODE45 de MATLAB

Profesor:
Blanca Guillén

Universidad Nacional Experimental del Táchira

Diciembre 2016

ODE45[1] es una subrutina propia de MATLAB diseñada para aproximar la solución del problema de valor inicial:

$$\mathbf{y}'(t) = \mathbf{f}(t, \mathbf{y}(t)), \quad a \leq t \leq b$$

$$\mathbf{y}(a) = \boldsymbol{\alpha}$$

donde t es la variable independiente, $\mathbf{y}$ es un vector de variables dependientes (funciones) a determinar y $\mathbf{f}(t, \mathbf{y})$ es una función de t y $\mathbf{y}$. El problema matemático queda completamente especificado cuando se conocen el vector de funciones $\mathbf{f}(t, \mathbf{y})$ y el vector de condiciones iniciales $\boldsymbol{\alpha}$.

[1] La subrutina ODE45 implementa el método de Runge-Kutta de orden (4,5) con tamaño de paso variable propuesto por John R.Dorman y P.J. Prince, conocido como el método Dorman-Prince.

La sintaxis básica para acceder a la función ode45 es:

```
[t,y]= ode45(fname,tspan,y0)
```

donde

- `fname`: nombre del archivo donde se describe la función diferencial a resolver.
- `tspan`: vector que especifica el intervalo de integración `[t0,tf]`. Para obtener soluciones en tiempos específicos se emplea: `tspan=[t0,t1,...,tf]`
- `y0` : vector que describe las condiciones iniciales $\mathbf{y}(a)$. Es decir, `y0`$=\boldsymbol{\alpha}$.

Ejemplo

Utilice la subrutina ode45 para aproximar la solución del problema de valor inicial:

$$\begin{aligned} y'(t) &= te^{3t} - 2y(t), \qquad 0 \leq t \leq 1 \\ y(0) &= 0 \end{aligned}$$

Compare sus resultados con la solución exacta:

$$y(t) = \frac{1}{5}te^{3t} - \frac{1}{25}e^{3t} + \frac{1}{25}e^{-2t}$$

De acuerdo con la sintaxis de ode45 se requiere definir las variables de entrada:

1. `fname`: función proporcionada por el usuario en donde se define la función $f(t, y)$. Hay dos alternativas para definir esta función, mediante un archivo de función `*.m` o como una función *handle* de MATLAB. En el caso del ejemplo que estamos considerando vamos a utilizar la función *handle*:

$$f = @(t, y)\ t * exp(3 * t) - 2 * y;$$

2. `tspan=[0,1]`.
3. `y0=0`.

Ejemplo de uso - solver ode45

Ejemplo

Utilice la subrutina ode45 para aproximar la solución del problema de valor inicial:

$$\begin{aligned} y'(t) &= te^{3t} - 2y(t), \qquad 0 \leq t \leq 1 \\ y(0) &= 0 \end{aligned}$$

Compare sus resultados con la solución exacta:

$$y(t) = \frac{1}{5}te^{3t} - \frac{1}{25}e^{3t} + \frac{1}{25}e^{-2t}$$

De acuerdo con la sintaxis de ode45 se requiere definir las variables de entrada:

1. `fname`: función proporcionada por el usuario en donde se define la función $f(t, y)$. Hay dos alternativas para definir esta función, mediante un archivo de función `*.m` o como una función *handle* de MATLAB. En el caso del ejemplo que estamos considerando vamos a utilizar la función *handle*:

$$f = @(t, y)\ t * exp(3 * t) - 2 * y;$$

2. `tspan`=[0,1].
3. y0=0.

Ejemplo

Utilice la subrutina ode45 para aproximar la solución del problema de valor inicial:

$$\begin{aligned} y'(t) &= te^{3t} - 2y(t), \qquad 0 \leq t \leq 1 \\ y(0) &= 0 \end{aligned}$$

Compare sus resultados con la solución exacta:

$$y(t) = \frac{1}{5}te^{3t} - \frac{1}{25}e^{3t} + \frac{1}{25}e^{-2t}$$

De acuerdo con la sintaxis de ode45 se requiere definir las variables de entrada:

1. `fname`: función proporcionada por el usuario en donde se define la función $f(t, y)$. Hay dos alternativas para definir esta función, mediante un archivo de función `*.m` o como una función *handle* de MATLAB. En el caso del ejemplo que estamos considerando vamos a utilizar la función *handle*:

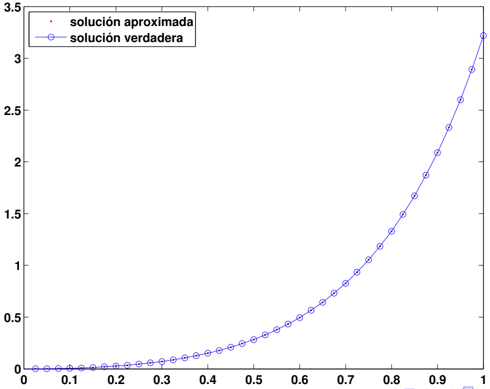
$$f = @(t, y)\ t * exp(3 * t) - 2 * y;$$

2. `tspan`=[0,1].
3. y0=0.

Llamado a la subrutina ode45:

```
>> [T,W]=ode45(f,tspan,y0);
```

Después de ejecutar la subrutina ode45 MATLAB devuelve los vectores `T` y `W` cada uno de tamaño 41×1, de tipo doble. El vector `W` es una aproximación de la solución verdadera y en los 41 puntos de red contenidos en el vector `T`. La siguiente figura muestra superpuestas las soluciones aproximada `W` (utilizando ode45) y verdadera y (proporcionada con el ejemplo).



Definición

Un sistema de m-ésimo orden de problemas de valor inicial es un conjunto de ecuaciones ecuaciones diferenciales de primer orden del tipo:

$$\begin{array}{ccc} y_1' & = & f_1(t, y_1, y_2, ..., y_m) \\ y_2' & = & f_2(t, y_1, y_2, ..., y_m) \\ . & . & . \\ . & . & . \\ . & . & . \\ y_m' & = & f_m(t, y_1, y_2, ..., y_m) \end{array}$$

para $a \leq t \leq b$, con las condiciones iniciales:

$$y_1(a) = \alpha_1, \quad y_2(a) = \alpha_2, \quad ....y_m(a) = \alpha_m,$$

Gracias a la estructura vectorizada que maneja MATLAB internamente, estos sistemas de ecuaciones pueden resolverse utilizando la subrutina ode45 sin mayores complicaciones. Sólo basta con darle estructura vectorial tanto a la función $\mathbf{f}$ como al vector de condiciones iniciales $\boldsymbol{\alpha}$.

Ejemplo

Considere el sistema de ecuaciones diferenciales de primer orden dado por

$$\begin{aligned} y_1'(t) &= 3y_1(t) + 2y_2(t) - (2t^2+1)e^{2t} \\ y_2'(t) &= 4y_1(t) + y_2(t) + (t^2+2t-4)e^{2t} \end{aligned}$$

para $0 \leq t \leq 1$. Sujeto a las condiciones iniciales

$$y_1(0) = y_2(0) = 1$$

La forma vectorizada de este ejemplo es:

$$\boldsymbol{y}(t) = \left[\begin{array}{c} y_1(t) \\ y_2(t) \end{array} \right] \qquad \boldsymbol{f}(t, \boldsymbol{y}(t)) = \left[\begin{array}{c} 3y_1(t) + 2y_2(t) - (2t^2+1)e^{2t} \\ 4y_1(t) + y_2(t) + (t^2+2t-4)e^{2t} \end{array} \right]$$

con vector de condiciones iniciales:

$$\boldsymbol{y}(0) = \boldsymbol{\alpha} = \left[\begin{array}{c} 1 \\ 1 \end{array} \right]$$

Como en el primer ejemplo, para acceder a la subrutina ode45 es preciso definir los parámetros de entrada: `fname`, `tspan` y `y0`.

1. `fname`: En el caso de sistemas de ecuaciones diferenciales la función vectorial $\mathbf{f}(t, \mathbf{y})$ debe definirse mediante un archivo de función `*.m`. Esto es:

```
function f=sistema1(t,y)

f=zeros(2,1);
f(1)=3*y(1)+2*y(2)-(2*t^2+1)*exp(2*t);
f(2)=4*y(1)+y(2)+(t^2+2*t-4)*exp(2*t);
```

2. `tspan=[0,1]`.
3. `y0=[1;1]`.
4. La solución aproximada del sistema se obtiene con el comando: `[T,W]=ode45(@sistema1,tspan,y0);`

Como en el primer ejemplo, para acceder a la subrutina ode45 es preciso definir los parámetros de entrada: `fname`, `tspan` y `y0`.

1. `fname`: En el caso de sistemas de ecuaciones diferenciales la función vectorial $\mathbf{f}(t, \mathbf{y})$ debe definirse mediante un archivo de función `*.m`. Esto es:

   ```
   function f=sistema1(t,y)

   f=zeros(2,1);
   f(1)=3*y(1)+2*y(2)-(2*t^2+1)*exp(2*t);
   f(2)=4*y(1)+y(2)+(t^2+2*t-4)*exp(2*t);
   ```

2. `tspan=[0,1]`.
3. `y0=[1;1]`.
4. La solución aproximada del sistema se obtiene con el comando: `[T,W]=ode45(@sistema1,tspan,y0);`

Ejemplo de uso - ode45 para sistemas de ecuaciones

Como en el primer ejemplo, para acceder a la subrutina ode45 es preciso definir los parámetros de entrada: `fname`, `tspan` y `y0`.

1. `fname`: En el caso de sistemas de ecuaciones diferenciales la función vectorial $\mathbf{f}(t, \mathbf{y})$ debe definirse mediante un archivo de función `*.m`. Esto es:

   ```
   function f=sistema1(t,y)

   f=zeros(2,1);
   f(1)=3*y(1)+2*y(2)-(2*t^2+1)*exp(2*t);
   f(2)=4*y(1)+y(2)+(t^2+2*t-4)*exp(2*t);
   ```

2. `tspan=[0,1]`.
3. `y0=[1;1]`.
4. La solución aproximada del sistema se obtiene con el comando: `[T,W]=ode45(@sistema1,tspan,y0);`

Como en el primer ejemplo, para acceder a la subrutina ode45 es preciso definir los parámetros de entrada: `fname`, `tspan` y `y0`.

1. `fname`: En el caso de sistemas de ecuaciones diferenciales la función vectorial $\mathbf{f}(t, \mathbf{y})$ debe definirse mediante un archivo de función `*.m`. Esto es:

   ```
   function f=sistema1(t,y)

   f=zeros(2,1);
   f(1)=3*y(1)+2*y(2)-(2*t^2+1)*exp(2*t);
   f(2)=4*y(1)+y(2)+(t^2+2*t-4)*exp(2*t);
   ```

2. `tspan=[0,1]`.
3. `y0=[1;1]`.
4. La solución aproximada del sistema se obtiene con el comando: `[T,W]=ode45(@sistema1,tspan,y0);`

La ecuación diferencial de 2do orden:

$$y''(t) = f(t, y(t), y'(t)), \quad a \leq t \leq b$$

con condiciones iniciales:

$$y(a) = \alpha_1, \qquad y'(a) = \alpha_2$$

puede ser reducida a un sistema de ecuaciones diferenciales de 1er orden mediante el cambio de variable:

$$y_1(t) = y(t), \qquad y_2(t) = y'(t)$$

En efecto, al aplicar el cambio de variable y derivar a las nuevas funciones y_1 y y_2 resulta el sistema de ecuaciones diferenciales de primer orden:

$$\begin{aligned} y_1' &= y' = y_2 \\ y_2' &= y'' = f(t, y_1, y_2) \end{aligned}$$

con condiciones iniciales:

$$\begin{aligned} y_1(a) &= y(a) = \alpha_1 \\ y_2(a) &= y'(a) = \alpha_2 \end{aligned}$$

Ejemplo

Considere la ecuación diferencial de segundo orden:

$$y''(t) = 2y'(t) - 2y(t) + e^{2t}\sin(t), \qquad 0 \leq t \leq 1$$

con condiciones iniciales

$$y(0) = -0.4, \quad y'(0) = -0.6$$

Al aplicar el cambio de variable recomendado $y_1 = y$ y $y_2 = y'$ resulta el sistema de ecuaciones diferenciales de primer orden:

$$\begin{aligned} y_1' &= y' = y_2 \\ y_2' &= y'' = 2y_2 - 2y_1 + e^{2t}\sin(t) \end{aligned}$$

con condiciones iniciales:

$$\begin{aligned} y_1(0) &= y(0) = -0.4 \\ y_2(0) &= y'(0) = -0.6 \end{aligned}$$

Ejemplo

Considere la ecuación diferencial de segundo orden:

$$y''(t) = 2y'(t) - 2y(t) + e^{2t}\sin(t), \qquad 0 \leq t \leq 1$$

con condiciones iniciales

$$y(0) = -0.4, \quad y'(0) = -0.6$$

La forma vectorizada de este sistema de ecuaciones es:

$$\boldsymbol{Y}(t) = \begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} \qquad \boldsymbol{F}(t, \boldsymbol{Y}(t)) = \begin{bmatrix} y_2(t) \\ 2y_2(t) - 2y_1(t) + e^{2t}\sin(t) \end{bmatrix}$$

con vector de condiciones iniciales:

$$\boldsymbol{Y}(0) = \boldsymbol{\alpha} = \begin{bmatrix} -0.4 \\ -0.6 \end{bmatrix}$$

Habiendo reducido el problema a un sistema de ecuaciones de 1er orden, se procede como antes. Esto es,se definen los parámetros de entrada: `fname`, `tspan` y `y0`.

1. `fname`: En este caso la función vectorial $\mathbf{F}(t, \mathbf{Y})$ se define mediante la función:

```
function F=sistema2(t,y)

F=zeros(2,1);
F(1)=y(2);
F(2)=2*y(2)-2*y(1)+exp(2*t)*sin(t);
```

2. `tspan=[0,1]`.
3. `y0=[-0.4  -0.6]`.
4. La solución aproximada del sistema se obtiene con el comando: `[T,W]=ode45(@sistema2,tspan,y0);`

Ejemplo de uso - ode45 para EDO de 2do orden

Habiendo reducido el problema a un sistema de ecuaciones de 1er orden, se procede como antes. Esto es,se definen los parámetros de entrada: `fname`, `tspan` y `y0`.

1. `fname`: En este caso la función vectorial $\mathbf{F}(t, \mathbf{Y})$ se define mediante la función:

```
function F=sistema2(t,y)

F=zeros(2,1);
F(1)=y(2);
F(2)=2*y(2)-2*y(1)+exp(2*t)*sin(t);
```

2. `tspan=[0,1]`.
3. `y0=[-0.4  -0.6]`.
4. La solución aproximada del sistema se obtiene con el comando: `[T,W]=ode45(@sistema2,tspan,y0);`

Habiendo reducido el problema a un sistema de ecuaciones de 1er orden, se procede como antes. Esto es,se definen los parámetros de entrada: `fname`, `tspan` y `y0`.

1. `fname`: En este caso la función vectorial $\mathbf{F}(t, \mathbf{Y})$ se define mediante la función:

```
function F=sistema2(t,y)

F=zeros(2,1);
F(1)=y(2);
F(2)=2*y(2)-2*y(1)+exp(2*t)*sin(t);
```

2. `tspan=[0,1]`.
3. `y0=[-0.4  -0.6]`.
4. La solución aproximada del sistema se obtiene con el comando: `[T,W]=ode45(@sistema2,tspan,y0);`

Ejemplo de uso - ode45 para EDO de 2do orden

Habiendo reducido el problema a un sistema de ecuaciones de 1er orden, se procede como antes. Esto es,se definen los parámetros de entrada: `fname`, `tspan` y `y0`.

1. `fname`: En este caso la función vectorial $\mathbf{F}(t, \mathbf{Y})$ se define mediante la función:

```
function F=sistema2(t,y)

F=zeros(2,1);
F(1)=y(2);
F(2)=2*y(2)-2*y(1)+exp(2*t)*sin(t);
```

2. `tspan=[0,1]`.
3. `y0=[-0.4  -0.6]`.
4. La solución aproximada del sistema se obtiene con el comando: `[T,W]=ode45(@sistema2,tspan,y0);`