

## Equazioni differenziali in MATLAB (1)

Data l'equazione differenziale

$$y' = f(y, t),$$

si vuole determinare una funzione  $y(t)$  tale che  $y'(t) = f(y, t)$  in un certo intervallo  $[t_o, t_f]$ .

Noto che per un certo  $t$  la funzione vale  $y(t)$ , allora la si estende ad un intorno di  $t$  stesso.

$$\frac{y(t+h) - y(t)}{h} \sim f(t, y(t)) \Rightarrow y(t+h) \sim y(t) + hf(t, y(t)),$$

dati  $h$  ed  $y(t)$  é possibile calcolare  $y(t+h)$ .

Sia adesso  $t_n = t_0 + nh$  (abbiamo suddiviso  $[t_o, t_f]$  in  $n$  sottointervalli), in generale

$$y(t_n + h) = y(t_{n+1}) \sim y(t_n) + hf(t_n, y(t_n))$$

indicando con  $y_n = y(t_n)$ , si ottiene

$$\boxed{y_{n+1} = y_n + hf(t_n, y_n)}$$

e noto il valore  $y(t_0) = y_0$  la soluzione del problema diventa unica.

### Integrazione numerica di ODE:

Oss: Il MATLAB invece di  $hf(t_n, y_n)$  viene utilizzata una funzione piú complessa che non suddivide l'intervallo temporale in nodi equispaziati, ma sceglie una successione di  $h_n$  per i quali  $\sum_n h_n = |t_f - t_0|$  e tale che l'errore commesso sia piccolo.

### Esempio1

Risolviamo il seguente problema:

$$\begin{cases} y' = e^{-(t+y)} \\ y(0) = 1 \end{cases}$$

con  $t_0 = 0, t_f = 5$ .

Di seguito in codice MATLAB che risolve il problema, la *function* che definisce la funzione da integrare é:

```
function [yp] = f(t,y)
yp = exp(-(t+y));
```

eseguendo il codice nel *WorkSpace*

```
>>[t,y] = ode23('f',[0,5],1)
>>t =
      .....
>>y =
      .....
>>plot(x,y)
```

Cambiamo adesso funzione

```
function [yp] = f(t,y)
yp = -y+sin(4*pi*t);
```

ed eseguiamo di nuovo il codice nel *WorkSpace*.

Altre routine per integrare le equazioni differenziali sono ad esempio *ode45*, *ode23t*, la sintassi ad esempio per *ode45* è la seguente

$$[tout, yout] = ode45(....., ....., ....., odeset),$$

*odeset* da le impostazione sulla precisione con la quale vogliamo lavorare, è una parola riservata nel MATLAB, all'interno di questa funzione bisogna impostare le tolleranze *Assoluta* e *Relativa* con le quali si vogliono eseguire i calcoli.

Nell'*help* in linea basta digitare *odeset*, *RelTol*, *AbsTol* per i particolari sulle routine:

```
>> help odeset
>> help RelTol
>> help AbsTol
```

L'impostazione delle tolleranze con *odeset* si esegue salvando le impostazioni in una struttura dati che chiameremo ad esempio *options*

```
>> options = odeset('AbsTol', 1.0e-10, 'RelTol', 1.0e-8);
```

In questo modo, verranno eseguiti i calcoli con una *Tolleranza Relativa* di  $10^{-10}$  ed una *Tolleranza Assoluta* di  $10^{-8}$ .

Mostriamo un esempio:

**Esempio 2:** Definiamo come sempre la funzione da integrare in un *m-file*

```
function [yp] = f(t,y)
yp = y+sin(4*pi*t);
```

ed eseguiamo l'integrazione con le tolleranze definite in precedenza:

```
>> [t,y] = ode45(@f,[0 10],0.5,options);
```

Nota: il comando @ seguito dal nome di una funzione serve per richiamare la funzione stessa (come si potrebbe fare anche usando gli apici oppure con il comando *feval*).

Se invece vogliamo eseguire i calcoli con le tolleranze scelte da MATLAB, basta sopprimere *options* da *ode45*

```
>> [t,y] = ode45(@f,[0 10],0.5);
```