Risoluzione numerica di equazioni differenziali: il metodo di Eulero

Il metodo piú semplice di integrazione numerica per equazioni differenziali é quello di *Eulero*. Consideriamo l'equazione del prim'ordine non omogenea ($\ldots = f(t)$) a coefficienti costanti ($a =$ costante):

$$y'(t) + ay(t) = f(t) \qquad \text{o} \qquad y'(t) = -ay(t) + f(t) \tag{1}$$

Partendo da un qualche tempo, t_0, il valore di $y(t_0 + h)$ puó essere approssimato dal valore di $y(t_0)$ piú il passo temporale moltiplicato per la pendenza della funzione, che é la derivata di $y(t)$ (Nota: questa é semplicemente una espansione al primo ordine di Taylor) :

$$y(t_0 + h) \approx y(t_0) + h \cdot y'(t_0)$$

chiamiamo questo valore approssimato $y^*(t)$

$$y^*(t_0 + h) = y(t_0) + h \cdot y'(t_0) \tag{2}$$

Quindi, se siamo in grado di calcolare il valore di $y'(t)$ al tempo t_0 (attraverso l'equazione 1), allora si puó generare una approssimazione per il valore di y al tempo $t_0 + h$ utilizzando l'equazione 2. Si puó quindi utilizzare questo nuovo valore di $y(t_0)$ per trovare $y'(t)$ (in t_0) e ripetere. Questo procedimento genera una soluzione approssimata ed é indicato come metodo di **Eulero esplicito**.

Con queste premesse il metodo di Eulero per equazioni differenziali del primo ordine puó essere schematizzato come segue:

1. Dato un tempo t_0, si sceglie un valore di h, e la condizione iniziale $y(t_0)$.
2. Da $y(t_0)$ si calcola la derivata di $y(t)$ per $t = t_0$ attraverso l'equazione 1. Chiamiamo questo valore k_1.
$$k_1 = y'(t_0)$$
3. Da questo si determina un valore approssimato per $y * (t_0 + h)$ utilizzando l'equazione 2.
4. Sia $t_0 = t_0 + h$, e $y(t_0) = y^*(t_0 + h)$.
5. Ripetere i passi da 2 a 4 fino ad un tempo finale T.

Esempio: Consideriamo il problema

$$\begin{cases} y'(t) & = -2y(t) + 4e^{-4t} \\ y(0) & = 1 \end{cases}$$

la cui soluzione esatta é

$$y(t) = \frac{5e^{-2t} - 3e^{-4t}}{2}$$

Usiamo $h = 0.1$ per determinare $y^*(t_0 + h) = y^*(t_0)$ (da $t_0 = 0$) calcolando la derivata:

$$\begin{aligned} y'(0) &= k_1 = 3e^{-4t_0} - 2y(t_0) \\ &= 3e^{-4 \cdot 0} - 2y(0) \\ &= 3 \cdot 1 - 2 \cdot 1 \\ &= 1 \end{aligned}$$

e quindi troviamo la soluzione approssimata dopo il passo in tempo:

$$\begin{aligned} y^*(t_0 + h) &= y(t_0) + h \cdot k_1 \\ y^*(0.1) &= y(0) + 0.1 \cdot 1 \\ &= 1 + 0.1 = 1.1 \end{aligned}$$

Per determinare l'approssimazione successiva consideriamo $t_0 = 0.1$ e ripetiamo calcolando la derivata utilizzando l'approssimazione per $y(t_0)$

$$\begin{aligned} y'(t_0+h) &= k_1 = 3e^{-4(t_0+h)} - 2y(t_0+h) \\ &= 3e^{-4 \cdot 0.1} - 2y(0.1) \\ &= 3 \cdot 0.67 - 2 \cdot 1.1 \\ &= -0.189 \end{aligned}$$

e quindi troviamo di nuovo la soluzione approssimata dopo il secondo passo in tempo:

$$\begin{aligned} y^*(t_0+h) &= y(t_0) + h \cdot k_1 \\ y^*(0.2) &= y(0.1) + 0.1 \cdot (-0.189) \\ &= 1.1 - 0.0189 = 1.08 \end{aligned}$$

la procedura si ripete fino a che $t = T$, con T un valore scelto.

Di seguito lo script il Matlab che implementa la procedura di Eulero.

```
% Script che mostra l'integrazione di Eulero per un problema
% del primo ordine con Matlab.

% Il problema da rsolvere e':

%y'(t)+2*y(t)=3*exp(-4*t)

% Nota: questo problema ha soluzione esatta nota definita come
% y(t)=2.5*exp(-2*t)-1.5*exp(-4*t)

h=0.1;    % h e' il passo temporale.
t=0:h:4;  % inizializzazione dell'intervallo temporale.

ystar(1)=1.0; % condizione iniziale.

for i=1:length(t)-1,       % cicolo "for".
   k1=3*exp(-4*t(i))-2*ystar(i);      % calcolo della derivata;
   ystar(i+1)=ystar(i)+h*k1;          % stima del nuovo valore di y;
end

% soluzione esatta
y=2.5*exp(-2*t)-1.5*exp(-4*t);

% Plot delle soluzioni esatta ed approssimata.
plot(t,ystar,'b--',t,y,'r-');
legend('Approximate','Exact');
title('Euler Approximation, h=0.01');
xlabel('Time');
ylabel('y*(t), y(t)');

%ciclo che stampa i valori numerici delle soluzioni
for i=1:length(t)
   disp(sprintf('t=%5.3f,  y(t)=%6.4f,  y*(t)=%6.4f',t(i),y(i),ystar(i)));
end
```

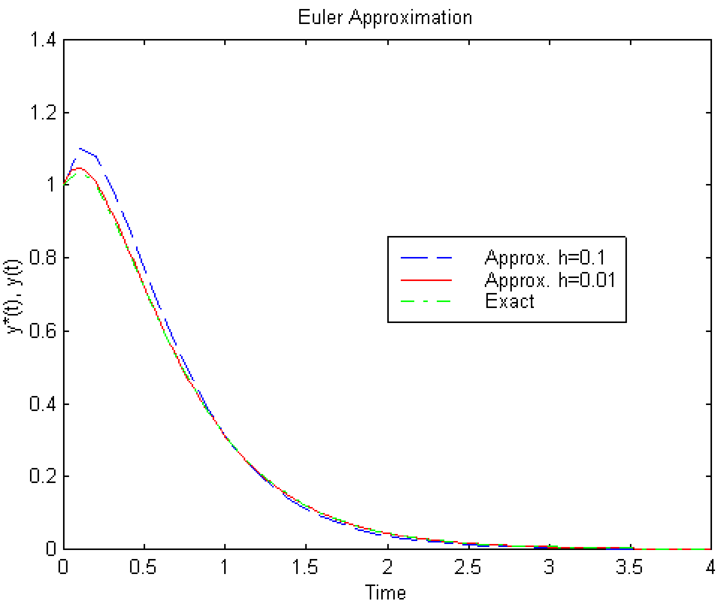

Figure 1: Output grafico delle soluzioni del problema in Matlab