Metodi Numerici con elementi di Programmazione

(A.A. 2013-2014)

Metodi Numerici

Appunti delle lezioni:

Equazioni differenziali ordinarie (2)

Docente Vittoria Bruni

Email: vittoria.bruni@sbai.uniroma1.it

Ufficio: Via A. Scarpa,

Pal. B, I piano, Stanza n. 16

Tel. 06 49766648

Ricevimento: Giovedì 14.00-15.00

Testi consigliati:

Calcolo Numerico, L. Gori, Ed. Kappa, 2006

Esercizi di Calcolo Numerico, L. Gori-M.L. Lo Cascio, F. Pitolli, Ed. Kappa, 2007

Il materiale didattico è disponibile sul sito

http://ingaero.uniroma1.it/

nella pagina dedicata al corso Metodi Numerici con elementi di Programmazione

Equazioni differenziali ordinarie di ordine superiore al primo

Esercizio

Si consideri la seguente equazione differenziale in $z(t)$

$$z'' + 2(z')^2 + 4z = t$$

con condizioni iniziali

$$z(0) = 0, \qquad z'(0) = 1$$

e lo si risolva numericamente usando due passi del metodo di Eulero con passo di discretizzazione $h = 1/2$.

Definendo il vettore

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} z \\ z' \end{pmatrix}$$

l'equazione precedente si riduce ad un sistema di due equazioni differenziali ordinarie del primo ordine

$$\begin{cases} y_1' = y_2 \\ y_2' = t - 2y_2^2 - 4y_1 \end{cases}$$

con condizioni iniziali

$$\begin{cases} y_1(0) = 0 \\ y_2(0) = 1 \end{cases}$$

e che in forma vettoriale si scrive

$$\frac{d\mathbf{y}}{dt} = \mathbf{f}(\mathbf{y}, t) \quad \mathbf{y}(0) = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

con

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} z \\ z' \end{pmatrix}, \quad \text{e} \quad \mathbf{f} = \begin{pmatrix} y_2 \\ t - 2y_2^2 - 4y_1 \end{pmatrix}.$$

Usando il metodo di Eulero calcoliamo

$$\mathbf{y}_{j+1} = \mathbf{y}_j + h\mathbf{f}(t_j, \mathbf{y}_j)$$

cioè

$$t_0 = 0 \qquad \mathbf{y}_1 = \mathbf{y}_0 + h\mathbf{f}(t_0, \mathbf{y}_0) = \begin{pmatrix} 0 \\ 1 \end{pmatrix} + 0.5 \cdot \begin{pmatrix} 1 \\ 0 - 2 \cdot 1^2 - 4 \cdot 0 \end{pmatrix} = \begin{pmatrix} 1/2 \\ 0 \end{pmatrix}$$

$$t_1 = 0.5 \qquad \mathbf{y}_2 = \mathbf{y}_1 + h\mathbf{f}(t_1, \mathbf{y}_1) = \begin{pmatrix} 1/2 \\ 0 \end{pmatrix} + 0.5 \cdot \begin{pmatrix} 0 \\ -3/2 \end{pmatrix} = \begin{pmatrix} 1/2 \\ -3/4 \end{pmatrix}$$

```
function [T] = eulero_sistemi(x0,y0,z0,h,n_step,ffun,gfun)
% Soluzione di un sistema di due equazioni differenziali
% del primo ordine con il metodo di Eulero
%
% INPUT
% x0, y0,z0 = consizioni iniziali
% h = passo di discretizzazione
% n = numero passi
% ffun = input(funzione f(x,y,z): );
% gfun = input(funzione g(x,y,z): );
%
% OUTPUT
% T = matrice contenente [xi;yi;zi]
```

```
xi(1) = x0; yi(1) = y0; zi(1) = z0;
for i = 2:n_step+1
    x = xi(i-1);
    y = yi(i-1);
    z = zi(i-1);
    k1 = ffun(x,y,z);
    t1 = gfun(x,y,z);
    xi(i) = x + h;
    yi(i) = y + h*k1;
    zi(i) = z + h*t1;
end
figure(1)
plot(xi,yi,'r',xi,zi,'b')

T = [xi;yi;zi];
```

Dal Command window

```
>> ffun = @(t,y,z)[z];
>> gfun = @(t,y,z)[t-2*z^2-4*y];
>> [T] = eulero_sistemi(0,0,1,.5,2,ffun,gfun);
>> disp(['        ti     ','    y1(i)     ', '    y2(i)    '    ]), disp(T')
       ti        y1(i)        y2(i)
         0            0     1.0000
    0.5000       0.5000          0
    1.0000       0.5000    -0.7500
```

Usando il metodo di RK del quarto ordine con lo stesso passo di discretizzazione si ha

```
>> [T] = RK_sistemi(0,0,1,.5,2,ffun,gfun);
>> disp(['        ti     ','    y1(i)     ', '    y2(i)    '    ]), disp(T')
       ti        y1(i)        y2(i)
         0            0     1.0000
    0.5000       0.3148     0.3259
    1.0000       0.3761    -0.0401
```

Usando il metodo di Eulero con passo di discretizzazione $h = 0.05$:

```
>> [T] = eulero_sistemi(0,0,1,.05,20,ffun,gfun);
>> disp(['      ti    ',',   y1(i)    ', '    y2(i)   '    ]), disp(T')
      ti       y1(i)      y2(i)
         0          0     1.0000
    0.0500     0.0500     0.9000
    0.1000     0.0950     0.8115
    0.1500     0.1356     0.7316
    0.2000     0.1722     0.6585
    0.2500     0.2051     0.5907
    0.3000     0.2346     0.5273
    0.3500     0.2610     0.4676
    0.4000     0.2844     0.4110
    0.4500     0.3049     0.3572
    0.5000     0.3228     0.3060
    0.5500     0.3381     0.2571
    0.6000     0.3509     0.2104
    0.6500     0.3614     0.1657
    0.7000     0.3697     0.1232
    0.7500     0.3759     0.0827
    0.8000     0.3800     0.0444
    0.8500     0.3823     0.0082
    0.9000     0.3827    -0.0258
    0.9500     0.3814    -0.0574
    1.0000     0.3785    -0.0865
```

Mentre il metodo di RK con passo $h = 0.05$ restituisce

```
>> [T] = RK_sistemi(0,0,1,.05,20,ffun,gfun);
>> disp(['        ti     ','    y1(i)    ', '    y2(i)   '    ]), disp(T')
        ti        y1(i)       y2(i)
         0            0     1.0000
    0.0500       0.0476     0.9057
    0.1000       0.0907     0.8211
    0.1500       0.1298     0.7440
    0.2000       0.1652     0.6730
    0.2500       0.1972     0.6070
    0.3000       0.2260     0.5452
    0.3500       0.2518     0.4870
    0.4000       0.2748     0.4319
    0.4500       0.2950     0.3797
    0.5000       0.3128     0.3300
    0.5500       0.3281     0.2827
    0.6000       0.3411     0.2378
    0.6500       0.3519     0.1950
    0.7000       0.3606     0.1544
    0.7500       0.3674     0.1160
    0.8000       0.3723     0.0798
    0.8500       0.3754     0.0458
    0.9000       0.3769     0.0142
    0.9500       0.3768    -0.0149
    1.0000       0.3754    -0.0415
```

Problemi ai limiti

Problemi ai limiti

Molti **problemi applicativi** hanno a che fare con grandezze dipendenti dalla **posizione** piuttosto che dal **tempo**. In questo caso il modello è un'equazione differenziale con **condizioni imposte in più punti**.

Esempio

Una trave di lunghezza l, tesa lungo la direzione della sua lunghezza da una forza S, è soggetta a un carico trasversale $t(x)dx$. La **deflessione** della trave $w(x)$ è descritta dall'equazione differenziale **lineare** del secondo ordine

$$\boxed{\frac{d^2w}{dx^2} = \frac{S}{EI}w(x) + \frac{t(x)}{2EI}(x-l) \qquad 0 < x < l}$$

x: **distanza** dall'estremo sinistro della trave; l: **lunghezza** della trave;
$t(x)$: intensità del **carico**; E: **modulo di elasticità** ;
S: **tensione** agli estremi della trave; I: **momento principale di inerzia**

Supponendo che la trave sia **fissata** agli estremi, all'equazione differenziale devono essere associate le **condizioni al bordo**

$$\boxed{w(0) = w(l) = 0}$$

Equazioni lineari: Esempio 1

$$\begin{cases} y'' - y = 0 \\ y(0) = 0 \qquad y(b) = \beta \end{cases}$$

Integrale generale: $y(x) = \gamma_1 \sinh x + \gamma_2 \cosh x$

Condizioni al bordo $\Rightarrow$ $\gamma_1 = \dfrac{\beta}{\sinh b} \qquad \gamma_2 = 0$

Soluzione: $\boxed{y(x) = \beta \dfrac{\sinh x}{\sinh b}}$ $\Rightarrow$ La soluzione **esiste** ed è **unica** per qualunque valore di b

Equazioni lineari: Esempio 2

$$\begin{cases} y'' + y = 0 \\ y(0) = 0 \qquad y(b) = \beta \end{cases}$$

Integrale generale: $y(x) = \gamma_1 \sin x + \gamma_2 \cos x$

Condizioni al bordo $\Rightarrow$ $\gamma_1 = \dfrac{\beta}{\sin b} \qquad \gamma_2 = 0$

Soluzione: $\boxed{y(x) = \beta \dfrac{\sin x}{\sin b}}$ $\Rightarrow$ La soluzione è **unica** solo se $b \neq n\pi, \quad n \in \mathbf{Z}$

Se $b = n\pi$ e $\beta = 0$ $\Rightarrow$ Ci sono **infinite** soluzioni $y(x) = c \sin x$

Se $b = n\pi$ e $\beta \neq 0$ $\Rightarrow$ **Non esistono** soluzioni

Problemi differenziali lineari

Consideriamo il problema differenziale **lineare**

$$\begin{cases} y'' = p(x)\, y' + q(x)\, y - r(x) & \qquad a < x < b \\ \\ y(a) = \alpha \qquad \qquad y(b) = \beta \end{cases}$$

Teorema. Se *i)* $p(x)$, $q(x)$ e $r(x)$ sono **continue** in $[a, b]$

ii) $|p(x)| \leq K$ e $q(x) > 0$ per $x \in [a, b]$

$\Rightarrow$ la **soluzione** del problema differenziale **esiste** ed è **unica**

Metodi alle differenze finite

I **metodi alle differenze finite** consistono nel sostituire ciascuna delle **derivate** che compaiono nell'equazione differenziale con una approssimazione ottenuta tramite le **differenze finite**.

Discretizzazione: si divide l'intervallo $[a, b]$ in $N+1$ **sottointervalli uguali** di ampiezza $h = \dfrac{b-a}{N+1}$ e si introducono $N+2$ **nodi equispaziati** x_i, $i = 0, 1, \ldots, N+1$

$$\boxed{x_i = a + ih, \;\; i = 0, 1, \ldots, N+1}$$

Nei punti **interni** x_i, $i = 1, \ldots, N$, l'equazione differenziale diventa

$$y''(x_i) = p(x_i)\, y'(x_i) + q(x_i)\, y(x_i) - r(x_i) \qquad\qquad i = 1, \ldots, N$$

Formula alle differenze centrate per $y'(x_i)$

Se $y \in C^3[x_{i-1}, x_{i+1}]$, per approssimare $y(x_{i+1})$ e $y(x_{i-1})$ si può sviluppare y in **serie di Taylor** con punto iniziale x_i e arrestandosi all'ordine 2:

$$y(x_{i+1}) = y(x_i + h) = y(x_i) + h\overset{\downarrow}{y'(x_i)} + \frac{1}{2!}h^2 y''(x_i) + \frac{1}{3!}h^3 y'''(\eta_i^+)$$

$$y(x_{i-1}) = y(x_i - h) = y(x_i) - hy'(x_i) + \frac{1}{2!}h^2 y''(x_i) - \frac{1}{3!}h^3 y'''(\eta_i^-)$$

$$\eta_i^+ \in (x_i, x_{i+1}) \qquad \eta_i^- \in (x_{i-1}, x_i)$$

Formula alle differenze finite centrate per $y'(x_i)$: sottraendo il secondo sviluppo dal primo e usando il teorema della media si ha

$$\boxed{y'(x_i) = \frac{y(x_{i+1}) - y(x_{i-1})}{2h} - \underbrace{\frac{h^2}{6} y^{(3)}(\eta_i)}_{\downarrow} \qquad \eta_i \in (x_{i-1}, x_{i+1})}$$

Errore di troncamento: $\tau(x_i, y(x_i); h; f) = -\frac{h^2}{6} y^{(3)}(\eta_i) = O(h^2)$ **Secondo ordine**

Formula alle differenze centrate per $y''(x_i)$

Se $y \in C^4[x_{i-1}, x_{i+1}]$, per approssimare $y(x_{i+1})$ e $y(x_{i-1})$ si può sviluppare y in **serie di Taylor** con punto iniziale x_i e arrestandosi all'ordine 4

$$y(x_{i+1}) = y(x_i + h) = y(x_i) + hy'(x_i) + \frac{1}{2!}h^2 y''(x_i) + \frac{1}{3!}h^3 y'''(x_i) + \frac{1}{4!}h^4 y^{(4)}(\xi_i^+)$$

$$y(x_{i-1}) = y(x_i - h) = y(x_i) - hy'(x_i) + \frac{1}{2!}h^2 y''(x_i) - \frac{1}{3!}h^3 y'''(x_i) + \frac{1}{4!}h^4 y^{(4)}(\xi_i^-)$$

$$\xi_i^+ \in (x_i, x_{i+1}) \qquad \xi_i^- \in (x_{i-1}, x_i)$$

Formula alle differenze finite centrate per $y''(x_i)$: sommando i due sviluppi in serie e usando il teorema della media si ha

$$\boxed{y''(x_i) = \frac{y(x_{i+1}) - 2y(x_i) + y(x_{i-1})}{h^2} - \underbrace{\frac{h^2}{12} y^{(4)}(\xi_i)} \qquad \xi_i \in (x_{i-1}, x_{i+1})}$$

Errore di troncamento: $\tau(x_i, y(x_i); h; f) = -\frac{h^2}{12} y^{(4)}(\xi_i) = O(h^2)$ **Secondo ordine**

Metodi alle differenze finite lineari

I **metodi alle differenze finite** consistono nel sostituire le **formule alle differenze finite** nell'**equazione differenziale** esatta collocata nei punti x_i, $i = 1, 2, \ldots, N$.

Operatore differenziale esatto:

$$(\mathcal{L}\, y)(x_i) := -y''(x_i) + p(x_i)y'(x_i) + q(x_i)y(x_i) = r(x_i) \qquad i = 1, ..., N$$

Metodo alle differenze finite:

$$(\mathcal{L}\, y)(x_i) = -\frac{y(x_{i+1}) - 2y(x_i) + y(x_{i-1})}{h^2} + p(x_i)\frac{y(x_{i+1}) - y(x_{i-1})}{2h} +$$

$$+ q(x_i)y(x_i) - \frac{h^2}{12}[2\, p(x_i)\, y^{(3)}(\eta_i) - y^{(4)}(\xi_i)] = r(x_i)$$
$$i = 1, ..., N$$

Schema alle differenze finite lineari

Indicando con y_i l' **approssimazione** di $y(x_i)$ e trascurando l'**errore di troncamento** nella discretizzazione dell'operatore differenziale $(\mathcal{L}\, y)(x_i)$, si ottiene lo **schema alle differenze finite lineari**:

$$\begin{cases} y_0 = \alpha \qquad\qquad y_{N+1} = \beta \\ -\dfrac{y_{i+1} - 2y_i + y_{i-1}}{h^2} + p(x_i)\dfrac{y_{i+1} - y_{i-1}}{2h} + q(x_i)\, y_i = r(x_i), \\ \hfill i = 1, 2, \ldots, N \end{cases}$$

Operatore differenziale discreto:

$$(\mathcal{L}_h\{y_i\})_i = -\frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} + p(x_i)\frac{y_{i+1} - y_{i-1}}{2h} + q(x_i)\, y_i \qquad i = 1, 2, \ldots, N$$

Schema numerico:

$$(\mathcal{L}_h\{y_i\})_i = r(x_i) \qquad\qquad i = 1, 2, \ldots, N$$

Errore di troncamento /1

Definizione. L'**errore di troncamento** è definito come la differenza tra l'**operatore differenziale esatto** e l'**operatore differenziale discreto** applicato alla soluzione esatta.

Operatore differenziale esatto:

$$(\mathcal{L}\,y)(x_i) = -y''(x_i) + p(x_i)y'(x_i) + q(x_i)y(x_i) = r(x_i)$$

Operatore differenziale discreto (applicato alla soluzione esatta):

$$(\mathcal{L}_h\,\{y(x_i)\})_i = -\frac{y(x_{i+1}) - 2y(x_i) + y(x_{i-1})}{h^2} +$$

$$+\, p(x_i)\frac{y(x_{i+1}) - y(x_{i-1})}{2h} + q(x_i)y(x_i)$$

Errore di troncamento:

$$\tau(x_i, y(x_i); h; f) := (\mathcal{L}\,y)(x_i) - (\mathcal{L}_h\,\{y(x_i)\})_i$$

Errore di troncamento /2

Operatore differenziale esatto
(dello schema alle differenze finite lineare):

$$(\mathcal{L}\, y)(x_i) = -\frac{y(x_{i+1}) - 2y(x_i) + y(x_{i-1})}{h^2} + p(x_i)\frac{y(x_{i+1}) - y(x_{i-1})}{2h} +$$

$$+ q(x_i)y(x_i) - \frac{h^2}{12}[2\, p(x_i)\, y^{(3)}(\eta_i) - y^{(4)}(\xi_i)]$$

Errore di troncamento:

$$\begin{aligned} \tau(x_i, y(x_i); h; f) &= (\mathcal{L}\, y)(x_i) - (\mathcal{L}_h\{y(x_i)\})_i = \\ &= -\frac{h^2}{12}[2\, p(x_i)\, y^{(3)}(\eta_i) - y^{(4)}(\xi_i)] \end{aligned}$$

Consistenza dello schema alle differenze finite

Definizione. Uno **schema alle differenze finite** è **consistente** se

$$\lim_{h \to 0} \tau(x, y(x); h; f) = 0$$

Errore di troncamento:

$$\tau(x_i, y(x_i); h; f) = -\frac{h^2}{12}[2\, p(x_i)\, y^{(3)}(\eta_i) - y^{(4)}(\xi_i)] = O(h^2) \rightarrow 0 \text{ per } h \rightarrow 0$$

$\Rightarrow$ Lo **schema alle differenze finite lineare** è **consistente** e del **secondo ordine** .

Nota. Un metodo del secondo ordine è **esatto** per tutti i polinomi di grado ≤ 2.

Stabilità dello schema alle differenze finite

Definizione. Uno **schema alle differenze finite lineare** è detto **stabile** se, data una **funzione discreta** $\{v_i\}_{i=0}^{N+1}$ definita sulla **discretizzazione** $x_i = a + ih$, $i = 0, 1, \ldots, N+1$, esiste una **costante** M tale che

$$\max_{0\le i\le N+1} |v_i| \le M \left\{ \max(|v_0|, |v_{N+1}|) + \max_{1\le i\le N} |(\mathcal{L}_h \{v_i\})_i| \right\}$$

Dalla stabilità segue la **limitazione dell'errore globale** $\{e_i\}_{i=0}^{N+1}$ con $e_i := y(x_i) - y_i$:

$$\max_{0\le i\le N+1} |e_i| \le M \max_{1\le i\le N} |(\mathcal{L}_h \{e_i\})_i|\} = M \max_{1\le i\le N} |(\mathcal{L}_h\{y(x_i)\})_i - (\mathcal{L}_h \{y_i\})_i|\} =$$

$$= M \max_{1\le i\le N} |\tau(x_i, y(x_i); h; f)|$$

Nota.
- $(\mathcal{L}_h \{y(x_i)\})_i = \underbrace{(\mathcal{L}y)(x_i)}_{=r(x_i)} - \tau(x_i, y(x_i); h; f)$
- $(\mathcal{L}_h \{y_i\})_i = r(x_i)$ (schema numerico)

Convergenza dello schema alle differenze finite

Teorema. Siano $K = \max_{a \le x \le b} |p(x)|$ e $0 < L = \min_{a \le x \le b} q(x)$.
Se $h\,K < 2 \;\Rightarrow$ lo **schema alle differenze finite lineari** è **stabile** con $M = \max(1, 1/L)$

Poiché la stabilità è soggetta alla **condizione** $h\,K < 2$ si parla di **stabilità condizionata**.

La **consistenza** e la **stabilità condizionata** implicano la **convergenza condizionata** dello schema alle differenze finite.

Sistema lineare

La soluzione approssimata si ottiene risolvendo il **sistema lineare tridiagonale** di N equazioni nelle N incognite $Y = [y_1, y_2, \cdots, y_N]^T$

$$AY = B$$

con

$$A = \begin{bmatrix} 2 + h^2 q(x_1) & -1 + \frac{h}{2} p(x_1) & 0 & \cdots & \cdots & 0 \\ -1 - \frac{h}{2} p(x_2) & 2 + h^2 q(x_2) & -1 + \frac{h}{2} p(x_2) & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & 0 & -1 - \frac{h}{2} p(x_N) & 2 + h^2 q(x_N) \end{bmatrix}$$

$$\begin{aligned} B &= \left[h^2 r(x_1) + \left(1 + \frac{h}{2} p(x_1) \right) \alpha, h^2 r(x_2), \ldots \right. \\ &\qquad \left. \ldots, h^2 r(x_{N-1}), h^2 r(x_N) + \left(1 - \frac{h}{2} p(x_N) \right) \beta \right]^T \end{aligned}$$

Se valgono le **ipotesi** dei teoremi precedenti e se, in particolare, $\boxed{h\, K \leq 2}$

$\Rightarrow$ il sistema lineare ammette un'**unica** soluzione

Esercizio

Approssimare la deflessione di una trave di acciaio lunga 120 cm con lo schema alle differenze finite lineare. Le caratteristiche della trave sono: $t(x) = qx$, $q = 100kg/m$, $E = 3.0 \cdot 10^7 kg/cm2$, $S = 1000kg$, $I = 625cm4$.

```
function [xi, yi] = ODE_diffin(a,b,N,alpha,beta,pfun,qfun,rfun)
% Function ODE_diffin: [xi, yi] = ODE_diffin(a,b,N,alpha,beta,p,q,r)
% Soluzione del problema differenziale lineare ai limiti
% y’’= p(x)y’+q(x)y-r(x)    a < x < b
% y(a)= alpha
% y(b)= beta
% con il metodo alle differenze finite lineare con passo h=(b-a)/(N+1).
% N  il numero di nodi interni
%


% Calcolo del passo di integrazione e del vettore dei nodi
%
h = (b-a)/(N+1);
xi = a+(0:N+1)*h;
```

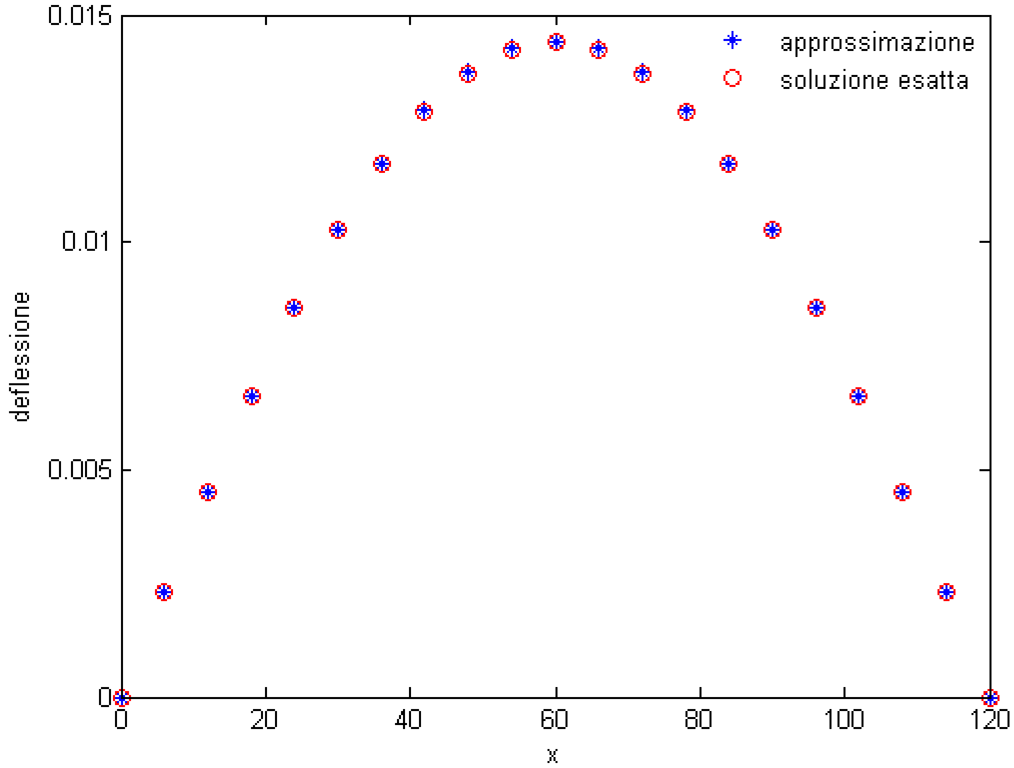
```
%
% Costruzione della matrice A
%
A_diag  = 2+h^2*qfun(xi(2:N+1)).*ones(1,N);
A_coinf = -1-h/2*pfun(xi(3:N+1)).*ones(1,N-1);
A_cosup = -1+h/2*pfun(xi(2:N)).*ones(1,N-1);
A = diag(A_diag)+diag(A_coinf,-1)+diag(A_cosup,1);
%

% Costruzione del vettore B
%
B = h^2*rfun(xi(2:N+1)).*ones(1,N);
B(1) = B(1)+alpha*(1+h/2*pfun(xi(2)));      % condizioni al bordo sn
B(N) = B(N)+beta*(1-h/2*pfun(xi(N+1)));     % condizioni al bordo dx
```

```
%
% Calcolo della soluzione approssimata
%

yi = A\B';
yi = [alpha yi' beta];

%
% Grafico della soluzione approssimata
%

figure(1)
plot(xi,yi,'b*')
```

Dal command window

```
>>L = 120;
>>S = 1000;
>>E = 3e7;
>>I = 625
>>q=100;
>>c1=S/(E*I);
>>c2=q/(2*E*I);
>>pfun = @(x)[0];
>>qfun = @(x)[c1];
>>rfun = @(x)[-c2*x.*(x-L)];
>>[xp, yp] = ODE_diffin(0,L,20-1,0,0,pfun,qfun,rfun);
>>disp([xp' yp'])
>>%Soluzione esatta
>>x = linspace(0,120,21);
>>wx=2*c2/(c1^2*(1+exp(sqrt(c1)*L)))*(exp(sqrt(c1)*x)+...
exp(-sqrt(c1)*(x-L)))-c2/c1^2*(2-c1*L*x+c1*x.^2);
>>disp([xp' yp'  wx' abs(wx'-yp')])
>>hold on, plot(x,wx,'ro')
```

```
>> disp([xp' yp' wx' abs(wx'-yp')])
  1.0e+002 *

                   0                   0                   0                   0
   0.06000000000000   0.00002298063068   0.00002292591613   0.00000005471455
   0.12000000000000   0.00004530466548   0.00004520100076   0.00000010366472
   0.18000000000000   0.00006638462726   0.00006623776862   0.00000014685864
   0.24000000000000   0.00008570215650   0.00008551786654   0.00000018428997
   0.30000000000000   0.00010280801030   0.00010259204311   0.00000021596718
   0.36000000000000   0.00011732206148   0.00011708017671   0.00000024188476
   0.42000000000000   0.00012893329792   0.00012867126148   0.00000026203644
   0.48000000000000   0.00013739982191   0.00013712338638   0.00000027643554
   0.54000000000000   0.00014254884971   0.00014226377709   0.00000028507262
   0.60000000000000   0.00014427671121   0.00014398875879   0.00000028795242
   0.66000000000000   0.00014254884971   0.00014226377709   0.00000028507262
   0.72000000000000   0.00013739982191   0.00013712338638   0.00000027643554
   0.78000000000000   0.00012893329792   0.00012867126148   0.00000026203644
   0.84000000000000   0.00011732206148   0.00011708017671   0.00000024188476
   0.90000000000000   0.00010280801030   0.00010259204311   0.00000021596718
   0.96000000000000   0.00008570215650   0.00008551786654   0.00000018428997
   1.02000000000000   0.00006638462726   0.00006623776862   0.00000014685864
   1.08000000000000   0.00004530466548   0.00004520099843   0.00000010366705
   1.14000000000000   0.00002298063068   0.00002292591846   0.00000005471222
   1.20000000000000                   0                   0                   0
```


0.015
0.01
0.005
0
deflessione
0
20
40
60
80
100
120
x
approssimazione
soluzione esatta

Esercizio $\begin{cases} y'' = \dfrac{2}{x}y' - \dfrac{2}{x^2}y + \dfrac{\sin(\log x)}{x}, \quad 1 < x < 2, \\ \\ y(1) = 1, \qquad\qquad y(2) = 2. \end{cases}$

Soluzione esatta:

$$y = \frac{x}{2}(4 - x) - x(1 - x)C - \frac{x^2}{2}\left(\cos(\log x) + \sin(\log x)\right)$$

$C = \cos(\log 2) + \sin(\log 2)$

Passo di Integrazione: $h = 0.1$

```
>> N = (2-1)/.1 -1
>>[xp, yp] = ODE_diffin(1,2,N,1,2,@(x)[2./x],@(x)[-2./x.^2],@(x)[-sin(log
>>x=linspace(1,2,N+2);
>>wx=x/2.*(4-x) - x.*(1-x)*(cos(log(2))+sin(log(2)))-x.^2/2.*(cos(log(x))
>> disp([xp' yp' abs(wx'-yp')])
   1.00000000000000   1.00000000000000                  0
   1.10000000000000   1.08997131797320   0.00010114433934
   1.20000000000000   1.17916606461186   0.00019022023241
   1.30000000000000   1.26868984692269   0.00026145441230
   1.40000000000000   1.35967878869232   0.00031053136854
   1.50000000000000   1.45328006475235   0.00033417783276
   1.60000000000000   1.55063839843931   0.00032988009877
   1.70000000000000   1.65288667818922   0.00029568731179
   1.80000000000000   1.76113946446343   0.00023007183610
   1.90000000000000   1.87648855084921   0.00013182838403
   2.00000000000000   2.00000000000000                  0
```

Nota. L'approssimazione può essere migliorata **riducendo** il passo h che però non può essere troppo piccolo a causa dell'**instabilità** che si produce nell'approssimare le derivate con le differenze finite.

Problemi differenziali non lineari

Consideriamo il problema differenziale del secondo ordine

$$\begin{cases} y'' = f(x, y, y') & a < x < b \\ y(a) = \alpha, \;\; y(b) = \beta & \textbf{condizioni al bordo} \end{cases}$$

dove $f(x, y, z) \in C^1([a, b] \times \mathbf{R} \times \mathbf{R})$ è una funzione **non lineare**.

Esistenza e unicità della soluzione

Teorema. Se *i)* $f(x, y, z)$ e le derivate parziali $f_y \equiv \frac{\partial f}{\partial y}$ e $f_z \equiv \frac{\partial f}{\partial z}$ sono **continue** in

$$D = \{(x, y, z) | a \leq x \leq b, -\infty < y, z < \infty\}$$

ii) $f_y(x, y, z) \geq \delta$ in D per qualche $\delta > 0$

iii) esistono due **costanti** K e L tali che

$$K = \max_{(x,y,z) \in D} f_y(x, y, z)$$

$$L = \max_{(x,y,z) \in D} |f_z(x, y, z)|$$

$\Rightarrow$ la **soluzione** del problema differenziale **esiste** ed è **unica**

Schema alle differenze finite

Sostituendo le **formule alle differenze finite centrate** nell'**equazione differenziale** $-y''(x_i) + f(x_i, y(x_i), y'(x_i)) = 0$, $i = 1, 2, \ldots, N$, si ha:

$$-\frac{y(x_{i+1}) - 2y(x_i) + y(x_{i-1})}{h^2} +$$

$$+ f\left(x_i, y(x_i), \frac{y(x_{i+1}) - y(x_{i-1})}{2h} - \frac{h^2}{6} y^{(3)}(\eta_i)\right) + \frac{h^2}{12} y^{(4)}(\xi_i) = 0$$

Lo **schema alle differenze finite** si ottiene trascurando l'**errore di troncamento** e aggiungendo le **condizioni al bordo**.

Se y_i è l' **approssimazione** di $y(x_i)$ si ha

$$\begin{cases} y_0 = \alpha \qquad y_{N+1} = \beta \\ (\mathcal{L}_h\{y_i\})_i = -\dfrac{y_{i+1} - 2y_i + y_{i-1}}{h^2} + f\left(x_i, y_i, \dfrac{y_{i+1} - y_{i-1}}{2h}\right) = 0 \quad i = 1, \ldots, N \end{cases}$$

Schema alle differenze finite non lineare

Si tratta di risolvere il **sistema non lineare** di N equazioni nelle N incognite $y_1, y_2, \ldots, y_N$

$$
\begin{cases}
2y_1 - y_2 + h^2 f\left(x_1, y_1, \dfrac{y_2 - \alpha}{2h}\right) - \alpha = 0 \\[2ex]
-y_1 + 2y_2 - y_3 + h^2 f\left(x_2, y_2, \dfrac{y_3 - y_1}{2h}\right) = 0 \\[2ex]
\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
-y_{N-2} + 2y_{N-1} - y_N + h^2 f\left(x_{N-1}, y_{N-1}, \dfrac{y_N - y_{N-2}}{2h}\right) = 0 \\[2ex]
-y_{N-1} + 2y_N + h^2 f\left(x_N, y_N, \dfrac{\beta - y_{N-1}}{2h}\right) - \beta = 0
\end{cases}
$$

Se valgono le **ipotesi** dei teoremi precedenti e, in particolare, $\boxed{hK \leq 2}$
$\Rightarrow$ il sistema ammette un'**unica soluzione**.

Soluzione del sistema con il metodo di Newton

La soluzione $[y_1, y_2, \ldots, y_N]^T$ del **sistema non lineare** può essere approssimata con il **metodo di Newton**: si genera una **successione di approssimazioni** $\{[y_1^{(k)}, y_2^{(k)}, \ldots, y_N^{(k)}]^T\}$ che **converge** alla soluzione se l'**approssimazione iniziale** $\{[y_1^{(0)}, y_2^{(0)}, \ldots, y_N^{(0)}]^T\}$ è abbastanza **vicina** alla soluzione e la **matrice Jacobiana** $J(y_1, \ldots, y_N)$ del sistema è **regolare**.

Matrice Jacobiana: $J(y_1, \ldots, y_N)$ è **tridiagonale**

$$[J(y_1, \ldots, y_N)]_{ij} = \begin{cases} -1 + \dfrac{h}{2} f_{y'}\left(x_i, y_i, \dfrac{y_{i+1} - y_{i-1}}{2h}\right) & j = i+1, \quad i = 1, \ldots, N-1 \\[2ex] 2 + h^2 f_y\left(x_i, y_i, \dfrac{y_{i+1} - y_{i-1}}{2h}\right) & j = i, \qquad\quad i = 1, \ldots, N \\[2ex] -1 - \dfrac{h}{2} f_{y'}\left(x_i, y_i, \dfrac{y_{i+1} - y_{i-1}}{2h}\right) & j = i-1, \quad i = 2, \ldots, N \end{cases}$$

con $y_0 = \alpha$ e $y_{N+1} = \beta$.

Esempio $\begin{cases} y'' = \frac{1}{8}(32 + 2x^3 - yy'), \quad 1 \le x \le 3 \\ y(1) = 17, y(3) = \dfrac{43}{3} \end{cases}$ $\Rightarrow \; y(x) = x^2 + \dfrac{16}{x}$

Passo di integrazione: $h = 0.1$

Criterio di arresto:

$$\max_{1 \le i \le N} |y_i^{(k+1)} - y_i^{(k)}| \le 10^{-8}$$

x_i	y_i	$y(x_i)$	$\lvert y(x_i) - y_i \rvert$
1.0	17.000000	17.000000	
1.1	15.754503	15.755455	9.520×10^{-4}
1.2	14.771740	14.773333	1.594×10^{-3}
1.3	13.995677	13.997692	2.015×10^{-3}
1.4	13.386297	13.388571	2.275×10^{-3}
1.5	12.914252	12.916667	2.414×10^{-3}
1.6	12.557538	12.560000	2.462×10^{-3}
1.7	12.299326	12.301765	2.438×10^{-3}
1.8	12.126529	12.128889	2.360×10^{-3}
1.9	12.028814	12.031053	2.239×10^{-3}
2.0	11.997915	12.000000	2.085×10^{-3}
2.1	12.027142	12.029048	1.905×10^{-3}
2.2	12.111020	12.112727	1.707×10^{-3}
2.3	12.245025	12.246522	1.497×10^{-3}
2.4	12.425388	12.426667	1.278×10^{-3}
2.5	12.648944	12.650000	1.056×10^{-3}
2.6	12.913013	12.913846	8.335×10^{-3}
2.7	13.215312	13.215926	6.142×10^{-4}
2.8	13.553885	13.554286	4.006×10^{-4}
2.9	13.927046	13.927241	1.953×10^{-4}
3.0	14.333333	14.333333	

Riferimenti bibliografici

L. Gori, *Calcolo Numerico*: §9.13 (escluso metodo shooting)