CN #44, 45 / 13 maggio 2013 / C43

Es (riepilogo):

PROCEDURA: dati: x_0, t_0, t_f, F, E — errore locale max ammesso

uscita: $x_0, \dots, x_N$; $t_0, \dots, t_N$

- $k = 0$;
- finché $t_k < t_f$ ripeti:
  - SCEGLI h_k
  - CALCOLA x_{k+1} } ... nell'Es con Metodo di EULERO
  - $t_{k+1} = t_k + h_k$
  - $k = k+1$

(1) Metodo di EULERO: $EL_k \sim h_{k-1}^2$

$\Rightarrow (EL_k \leq E)$ $\boxed{h_{k-1} \sim \sqrt{E}}$

(2) se $\forall h_k \sim \sqrt{E}$, $\boxed{N \sim \frac{1}{\sqrt{E}}}$

(3) se ciascun err locale è "traslato" a t_f,

$\boxed{\max ET_k \sim NE \sim \sqrt{E}}$

M. di Eulero: di ORDINE 1

---

def: se $\forall \underset{\sim}{x}, \underset{\sim}{t}$

$$\| \hat{x} - x(\underset{\sim}{t} + h; \underset{\sim}{x}, \underset{\sim}{t}) \| \sim h^{p+1}$$

METODO di ORDINE p

$\hat{x}$: appross fornita dal metodo per $x(\underset{\sim}{t} + h; \underset{\sim}{x}, \underset{\sim}{t})$

In generale:

(1) $EL_k \sim h_{k-1}^{p+1} \Rightarrow h_{k-1} \sim E^{\frac{1}{p+1}}$

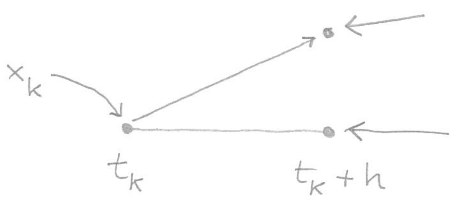
(2) $N \sim E^{-\frac{1}{p+1}}$

(3) $\boxed{\max ET_k \sim E^{1-\frac{1}{p+1}} = E^{\frac{p}{p+1}}}$

<u>Es</u>: Procedura eulero_pv ...

- <u>stima</u> dell' err locale ...

<u>ip</u>: $EL_k = Ch^2$

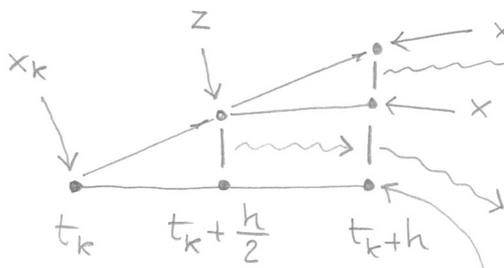


$xx1 = x_k + h\,F(t_k, x_k)$

$x(t_k+h;\, x_k, t_k)$

1° step

$\Rightarrow \| xx1 - x(t_k+h;\, x_k, t_k) \| = Ch^2$



$xx2 \rightsquigarrow c\,\frac{h^2}{4}$

$x(t_k+h;\, z, t_k+\frac{h}{2})$

$c\,\frac{h^2}{4}$ (<u>ip</u>: err locale a $t_k + \frac{h}{2}$ "trasla" inalterato a $t_k + h$)

$\Rightarrow \| xx2 - x(t_k+h;\, x_k, t_k) \| \simeq c\,\frac{h^2}{2}$

2° step

<u>3° step</u>:

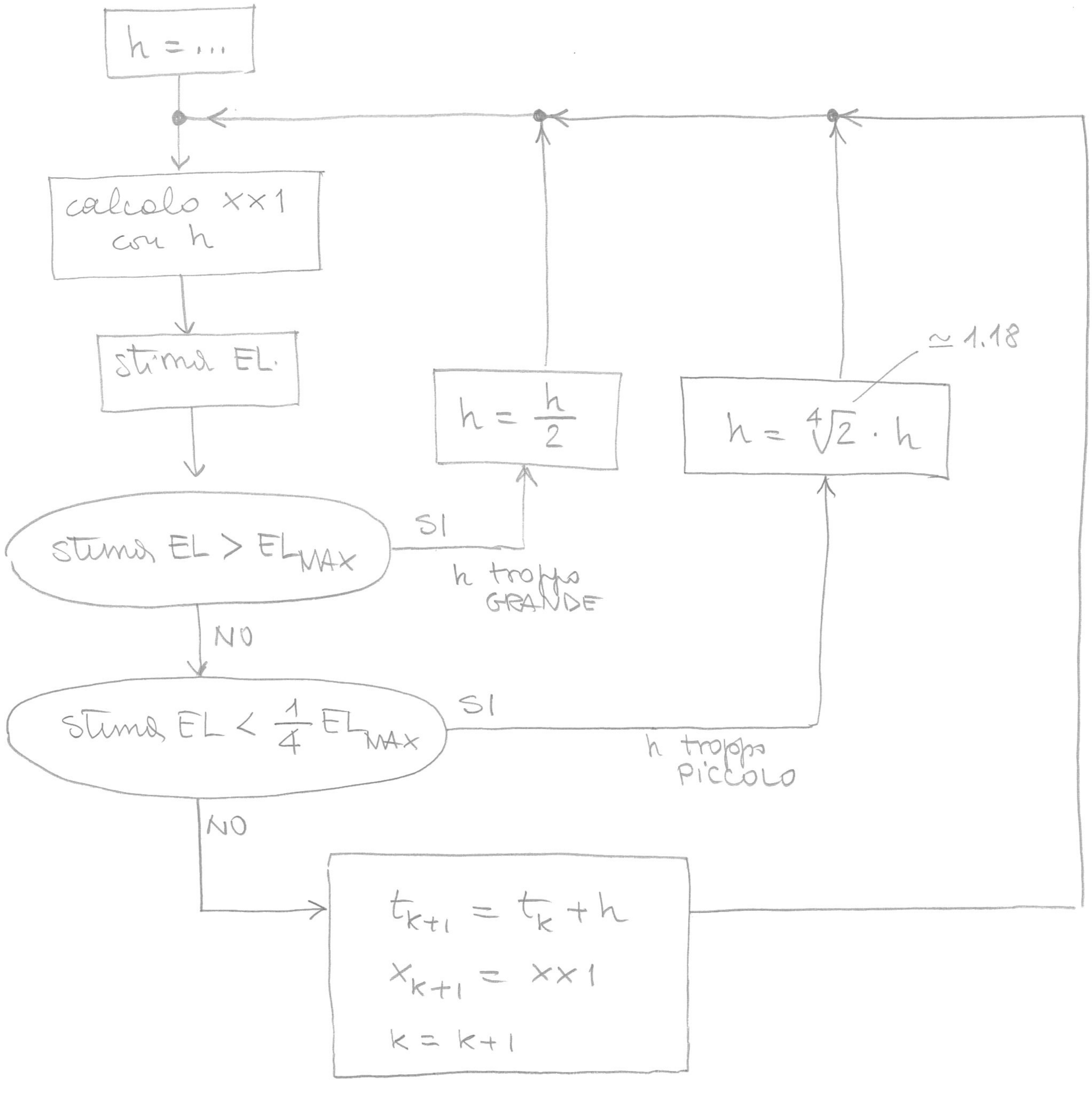
$\| xx1 - xx2 \| \simeq \| x1 - x(t_k+h;\, x_k, t_k) \| -$

$$- \| xx2 - x(t_k; x_k, t_k) \| \simeq$$

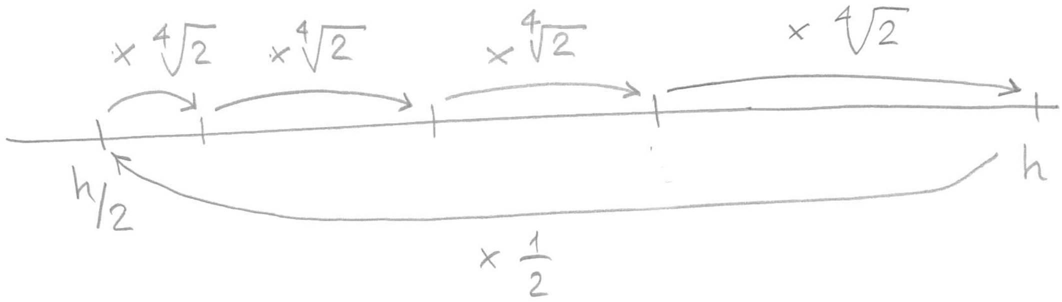
$$\simeq Ch^2 - C\frac{h^2}{2} = \frac{1}{2} Ch^2$$

$$\Rightarrow \boxed{2\,|xx1 - xx2| = |C|\, h^2 = |el_k|}$$

- <u>scelta</u> del passo:



Oss:



Pb di Cauchy: $\begin{cases} \dot{x} = x + t \\ x(0) = 1 \end{cases}$ $\quad t \in [0, t_f]$

$$x(t) = -1 - t + 2e^{t}$$

- **for** $i = 1:6$,

  EL_MAX = 10^{-i}; x∅ = 1; t∅ = 0; tf = 3;

  [T, X, PASSO] = eulero_pv(x∅, t∅, tf, f, EL_MAX, "zitto");

  ET_MAX(i) = ...

  N(i) = length (PASSO)

  h_MEDIO (i) = mean (PASSO) = $\dfrac{t_f - t_0}{N(i)}$

  end

i	$\dfrac{ET\_MAX(i+1)}{ET\_MAX(i)}$	$\dfrac{N(i+1)}{N(i)}$	$\dfrac{h\_MEDIO(i+1)}{h\_MEDIO(i)}$
1	0.34	3.45	0.29
2	0.32	3.09	0.32
3	0.33	3.21	0.31
4	0.31	3.25	0.31
5	0.31	3.03	0.33

Oss: $E' = \frac{E}{\alpha} \Rightarrow$

Per un metodo di ORDINE p ...

- $h' \sim \frac{h}{\alpha^{\frac{1}{p+1}}}$
- $N' \sim N \, \alpha^{\frac{1}{p+1}}$
- $\max ET_k' = \frac{\max ET_k}{\alpha^{\frac{p}{p+1}}}$

Metodo di EULERO (ordine 1)

$$E' = \frac{E}{10} \Rightarrow \quad h' = \frac{h}{10^{1/2}}, \quad N' = N \, 10^{1/2}$$

$$\max ET_k' = \frac{\max ET_k}{10^{1/2}}$$

$10^{1/2} \simeq 3.16$ / $10^{-1/2} \simeq 0.316$

$\frac{N'}{N}$

$\frac{h'}{h}, \quad \frac{\max ET_k'}{\max ET_k}$

```
function [T, X, PASSO]=eulero_pv(x0, t0, tf, fct, EL_MAX, dialogo)
//  Integra numericamente, sull'intervallo [t0,tf], il problema
//  di Cauchy in R(n):
//  .
//  x = F(t,x)
//  x(t0) = x0
//
//  con il metodo di Eulero in avanti a passo variabile.
//
//  x0: condizione iniziale (colonna di n elementi)
//  t0: istante iniziale
//  tf: istante finale
//  fct: function per F - fct(t,x) deve essere una colonna
//  EL_MAX: errore locale massimo consentito
//  dialogo: se "loquace" mostra gli istanti di integrazione e, ad ogni iterazione,
//           se è stato necessario aumentare o diminuire la lunghezza del passo
//
//  T = [T(1),...,T(N)], nodi
//  X: matrice n x N - la colonna X(:,i) è la soluzione numerica
//     all'istante T(i)
//  PASSO: riga con PASSO(k) = h tale che T(k+1) = T(k) + h
//
n = length(x0); // determina il numero di equazioni del sistema
h_min = (tf - t0)/1d7; // passo minimo consentito
h_max = (tf - t0)/10;  // passo massimo consentito
T = [];
X = [];
PASSO = [];
//
T(1,1) = t0;
X(:,1) = x0;
//
h = h_max/10;  // passo iniziale
//
// ciclo principale
//
while (T(1,$) < tf) & (h >= h_min),  // l'iterazione si arresta se
                                     // si è raggiunto tf o se non si
                                     // è riuscito a rendere la stima dell'errore
                                     // locale inferiore a EL_MAX
    h_max_loc = min(tf - T(1,$), h_max);
    if h > h_max_loc then h = h_max_loc; end;
    xx1 = X(:,$) + h*fct(T(1,$),X(:,$));
    // stima EL
    xx2 = X(:,$) + (h/2)*fct(T(1,$),X(:,$));
    xx2 = xx2 + (h/2)*fct(T(1,$)+h/2,xx2);
    StimaEL = 2*norm(xx1 - xx2);
    // fine stima EL
    // decide se accettare il passo in base al valore di StimaEL
    if StimaEL > EL_MAX then  // passo troppo lungo: riduci h...
       h = h/2; if dialogo == "loquace" then printf("-"); end;
       elseif (StimaEL < EL_MAX/4) & (h < h_max_loc) then
              // passo troppo corto: aumenta h...
              h = min(1.18*h, h_max_loc); // 1.18 ~= 2^(1/4): 4 cicli per
                                          // "recuperare il dimezzamento"
              if dialogo == "loquace" then printf("+"); end;
              else T(1,$+1) = T(1,$) + h;  // valore di h accettato...
                   if dialogo == "loquace" then printf("\nT = %3.2e",T($)); end;
                   X(:,$+1) = xx1;
                   PASSO(1,$+1) = h;
    end;
end;
if dialogo == "loquace" then printf("\n"); end;
//
```