

## ▷ Lezione VIII

# Equazioni differenziali ordinarie

**Libro** Capitolo 8, sezione 8.2, 8.3, 8.8, 8.9, 8.10

## Equazioni differenziali ordinarie

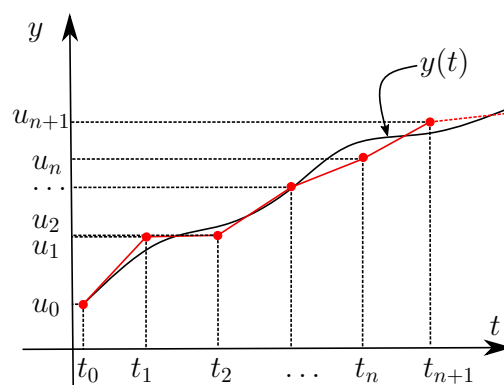
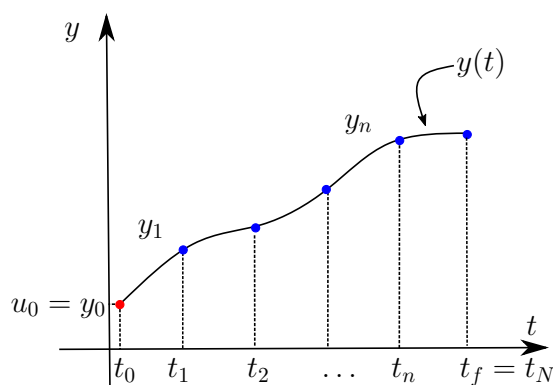
Consideriamo il seguente problema differenziale in cui ricerchiamo la funzione  $y$  dipendente dal parametro  $t \in [t_0, t_f]$  tale che

$$\begin{cases} y'(t) = f(t, y(t)) \\ y(t_0) = \bar{y}_0, \end{cases}$$

dove  $t_0$  e  $t_f$  sono l'istante iniziale e finale della simulazione, rispettivamente. Inoltre  $f$  è una funzione data, così come  $\bar{y}_0 \in \mathbb{R}$  che rappresenta il valore al tempo iniziale della soluzione  $y$ . Per poter risolvere numericamente il problema suddividiamo l'intervallo  $I = [t_0, t_f]$  in  $N$  sottointervalli con la stessa dimensione data da

$$h = \frac{t_f - t_0}{N}$$

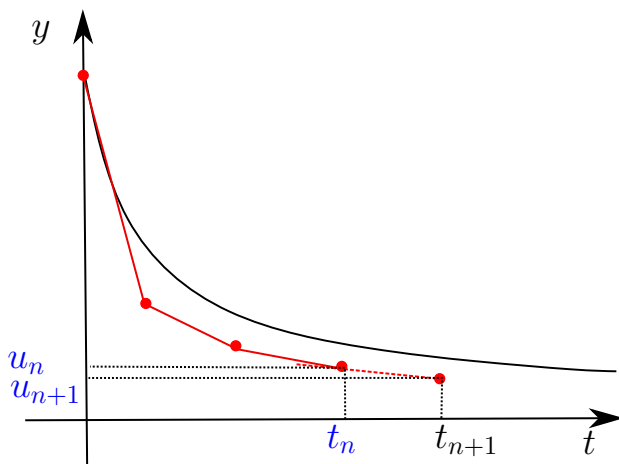
e definiamo gli istanti discreti in cui calcoleremo la soluzione approssimata come  $t_n = t_0 + nh$ . Infine chiamiamo  $u_n$  l'approssimazione di  $y$  all'istante  $t_n$ , ovvero vogliamo costruire  $u_n \approx y(t_n)$ . Inoltre, per comodità indicheremo con  $y_n = y(t_n)$



Una possibilità per discretizzare il problema di Cauchy presentato prima è utilizzare una semplice approssimazione del tipo

$$\begin{cases} \frac{u_{n+1} - u_n}{h} = f(t_n, u_n) \\ u_0 = \bar{y}_0 \end{cases} \quad \Rightarrow \quad \begin{cases} u_{n+1} = u_n + hf(t_n, u_n) \\ u_0 = \bar{y}_0 \end{cases}$$

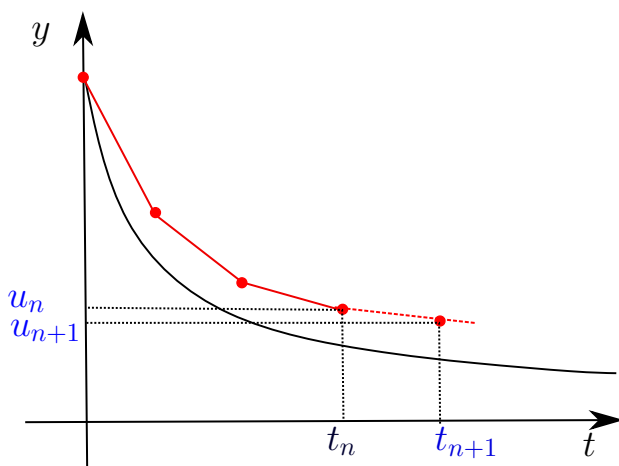
Quello che otteniamo è quindi una sequenza di valori  $\{u_0, u_1, u_2, \dots, u_N\}$  che approssimano la reale soluzione  $y$  negli istanti  $\{t_0, t_1, t_2, \dots, t_N\}$ . Graficamente abbiamo che



Tale metodo è detto metodo di Eulero in avanti, detto anche Eulero esplicito dato che posso calcolare direttamente  $u_{n+1}$  partendo dalla soluzione al passo precedente  $u_n$ . Un modo alternativo è discretizzare il problema di Cauchy utilizzando un'approssimazione del tipo

$$\begin{cases} \frac{u_{n+1} - u_n}{h} = f(t_{n+1}, u_{n+1}) \\ u_0 = \bar{y}_0 \end{cases} \Rightarrow \begin{cases} u_{n+1} = u_n + hf(t_{n+1}, u_{n+1}) \\ u_0 = \bar{y}_0 \end{cases}$$

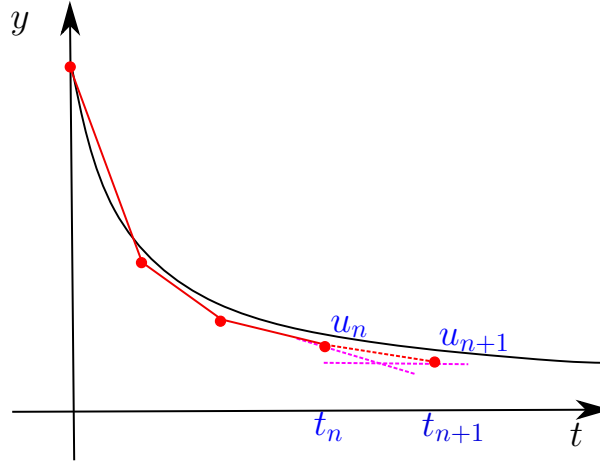
Quello che otteniamo è, come per il metodo di Eulero esplicito, una sequenza di valori  $\{u_0, u_1, u_2, \dots, u_N\}$  che approssimano la reale soluzione  $y$  negli istanti  $\{t_0, t_1, t_2, \dots, t_N\}$ . Graficamente abbiamo che



Tale metodo è detto metodo di Eulero all'indietro, o anche Eulero implicito. In questo caso per calcolare la nuova soluzione  $u_{n+1}$  devo risolvere un problema che può essere, in generale, non-lineare. Risulta quindi più costoso dell'algoritmo presentato precedentemente. Un'ulteriore possibilità è quella di utilizzare una media della funzione rispetto ai due metodi precedenti, il problema di Cauchy è ora approssimato come

$$\begin{cases} \frac{u_{n+1} - u_n}{h} = \frac{f(t_n, u_n) + f(t_{n+1}, u_{n+1})}{2} \\ u_0 = \bar{y}_0 \end{cases} \Rightarrow \begin{cases} u_{n+1} = u_n + \frac{h}{2}[f(t_n, u_n) + f(t_{n+1}, u_{n+1})] \\ u_0 = \bar{y}_0 \end{cases}$$

Quello che otteniamo è, come per i metodi precedenti, una sequenza di valori  $\{u_0, u_1, u_2, \dots, u_N\}$  che approssimano la reale soluzione  $y$  negli istanti  $\{t_0, t_1, t_2, \dots, t_N\}$ . Graficamente abbiamo che



Tale metodo è detto metodo di Crank-Nicolson, e come il metodo di Eulero all'indietro è un metodo implicito, ovvero, per calcolare la nuova soluzione  $u_{n+1}$  devo risolvere un problema non-lineare.

Un modo alternativo per derivare il metodo di Crank-Nicolson è tramite la quadratura numerica, ovvero

$$y'(t) = f(t, y(t)) \xrightarrow{\text{integro su } [t_n, t_{n+1}]} \int_{t_n}^{t_{n+1}} y'(t) dt = \int_{t_n}^{t_{n+1}} f(t, y(t)) dt$$

utilizzando un'approssimazione del secondo integrale con il metodo dei trapezi sul secondo termine otteniamo che

$$\int_{t_n}^{t_{n+1}} f(t, y(t)) dt \approx \frac{h}{2} [f(t_n, y_n) + f(t_{n+1}, y_{n+1})]$$

approssimando  $y_n$  con  $u_n$  e  $y_{n+1}$  con  $u_{n+1}$  ottengo la formula

$$u_{n+1} - u_n = \frac{h}{2} [f(t_n, u_n) + f(t_{n+1}, u_{n+1})]$$

che è appunto la stessa espressione vista prima valutata in  $y$ .

Possiamo ora introdurre una definizione generale che include tutti i metodi precedenti come caso particolare. Chiamiamo  $\theta$ -metodo uno schema che ha questa espressione,

$$\begin{cases} u_{n+1} = u_n + h[\theta f(t_{n+1}, u_{n+1}) + (1 - \theta)f(t_n, u_n)] \\ u_0 = \bar{y}_0 \end{cases}$$

dove  $\theta$  è un parametro che assume valore nell'intervallo  $[0, 1]$ . Per  $\theta = 0$  otteniamo il metodo di Eulero in avanti, per  $\theta = 1$  otteniamo il metodo di Eulero all'indietro, mentre per  $\theta = \frac{1}{2}$  otteniamo il metodo di Crank-Nicolson.

### Consistenza, stabilità e convergenza

Vogliamo valutare l'errore commesso utilizzando un metodo numerico per la soluzione del problema di Cauchy e verificare se

$$\lim_{h \rightarrow 0} |y(t_n) - u_n| = \lim_{h \rightarrow 0} |e_n| = 0 \quad \forall n = 0, \dots, N$$

ovvero se il metodo ha un errore  $e_n = y(t_n) - u_n$  che converge a zero. Il primo step è analizzare la *consistenza* dei metodi visti. Consideriamo ad esempio il metodo di Eulero Esplicito e valutiamo il residuo  $\epsilon_{n+1}$  che si ottiene sostituendo la soluzione esatta nel metodo numerico. Abbiamo infatti che

$$\begin{aligned} u_{n+1} &= u_n + hf(t_n, u_n) \\ y(t_{n+1}) &\neq y(t_n) + hf(t_n, y(t_n)) \end{aligned}$$

dove la non uguaglianza è data dal fatto che, in generale, la soluzione  $y$  non soddisfa lo schema numerico. Per ottenere l'uguaglianza usiamo uno sviluppo di Taylor e  $\xi$  è un punto nell'intervallo  $[t_n, t_{n+1}]$ .

$$y(t_{n+1}) = y(t_n) + hf(t_n, y(t_n)) + h^2 \frac{1}{2} f'(\xi, y(\xi))$$

Abbiamo quindi che l'errore al passo  $n$  è dato da  $\epsilon_{n+1} = h^2 \frac{1}{2} f'(\xi, y(\xi)) = \mathcal{O}(h^2)$ . Definiamo l'errore di troncamento locale  $\tau_{n+1}$  tale che  $\epsilon_{n+1} = h\tau_{n+1}$ . Osserviamo che

$$y(t_{n+1}) - u_{n+1} = y(t_n) - u_n + h\tau_{n+1}.$$

Definiamo anche l'errore di troncamento globale  $\tau$  come

$$\max_{n=0, \dots, N} \tau_n(h) = \tau(h).$$

### Definizione 8.1 - consistenza

Un metodo numerico per la risoluzione di un problema di Cauchy è detto consistente se  $\tau(h)$ , errore di troncamento globale è tale che

$$\lim_{h \rightarrow 0} \tau(h) = 0$$

e diremo che è consistente di ordine  $p$  se

$$\tau(h) \sim ch^p.$$

Osserviamo che non è sufficiente che  $h\tau_n(h)$  tenda a zero con  $h$ : infatti, ricordiamo che il residuo  $h\tau_n(h)$  è relativo a un singolo passo del metodo, ma per  $h \rightarrow 0$  il numero di passi  $N \rightarrow \infty$ .

Un altro aspetto importante è la *stabilità*: risulta importante anche capire come il metodo numerico considerato propaga le eventuali perturbazioni sui dati. Consideriamo ad esempio un problema e il corrispondente problema perturbato

$$\begin{cases} u_{n+1} = u_n + hf(t_n, u_n) \\ u_0 = \bar{y}_0 \end{cases} \xrightarrow{\text{perturbo il sistema}} \begin{cases} z_{n+1} = z_n + hf(t_n, z_n) + \rho_n \\ z_0 = \bar{y}_0 + \rho_0 \end{cases}$$

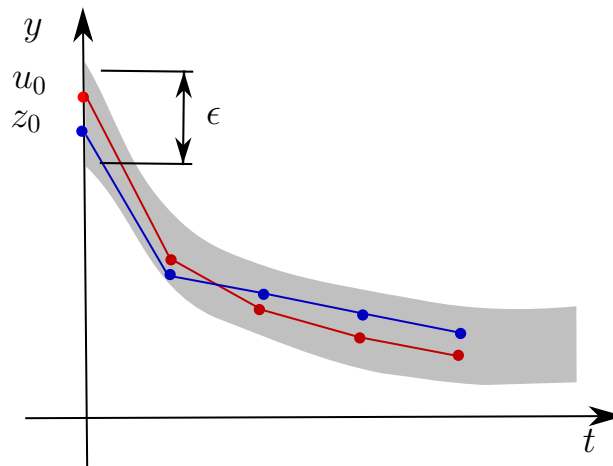
dove  $\rho_n$  è la perturbazione, derivante da approssimazione dei dati o arrotondamento. Diamo la seguente definizione

**Definizione 8.2 - zero stabilità**

Un metodo numerico per la risoluzione di un problema di Cauchy è detto zero-stabile se esiste  $h_0 > 0$ ,  $c > 0$  e  $\epsilon_0 > 0$  tale che per ogni  $h \in (0, h_0]$  e per ogni  $\epsilon \in (0, \epsilon_0]$  se  $|\rho_n| < \epsilon$  allora abbiamo che

$$|u_n - z_n| < c\epsilon \quad \forall n = 0, \dots, N.$$

Abbiamo quindi che, se il metodo è zero-stabile, la distanza fra  $|u_n - z_n|$  rimane limitata per tutti i passi temporali se le perturbazioni sono limitate, si veda la figura



Esiste un teorema che garantisce che se un metodo numerico è consistente e zero-stabile allora è anche convergente.

**Teorema 8.1 di Lax-Richtmyer**

Uno schema alle differenze finite consistente per un problema ben posto di evoluzione è convergente se e solo se è zero-stabile.

**Analisi di convergenza****Definizione 8.3 - convergenza**

Un metodo numerico per la risoluzione di un problema di Cauchy è detto convergente se

$$\lim_{h \rightarrow 0} |y(t_n) - u_n| = 0 \quad \forall n = 0, \dots, N.$$

In particolare se vale

$$|y(t_n) - u_n| \sim ch^p$$

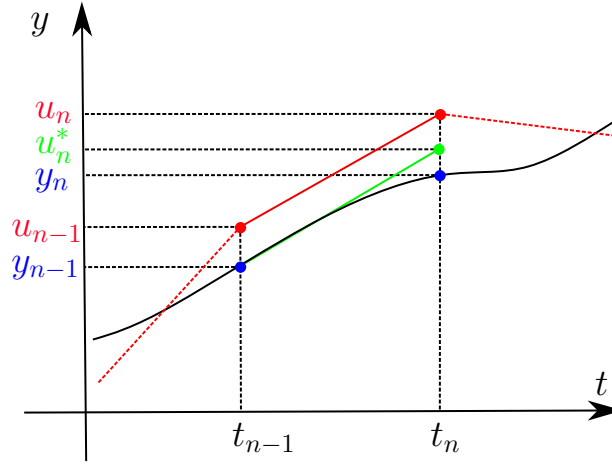
dove  $p$  è una costante, diremo che il metodo converge con ordine  $p$ , ovvero che il metodo ha ordine di accuratezza  $p$ .

Studiamo ora l'analisi di convergenza per il metodo di Eulero in avanti, ovvero vogliamo capire se la seguente espressione è vera

$$\lim_{h \rightarrow 0} |y(t_n) - u_n| = \lim_{h \rightarrow 0} |e_n| = 0 \quad \forall n = 0, \dots, N.$$

Introduciamo  $u_n^*$  come la soluzione che otterrei al passo  $n$  partendo dalla soluzione esatta al passo  $n - 1$ , ovvero

$$u_n^* = y(t_{n-1}) + hf(t_{n-1}, y(t_{n-1}))$$



L'errore è così scomponibile

$$e_n = [y(t_n) - u_n^*] + [u_n^* - u_n],$$

in cui il primo termine rappresenta dell'errore prodotto dal metodo dal passo  $t_{n-1}$  al passo  $t_n$ , mentre il secondo è la propagazione degli errori prodotti e accumulati fino al passo  $t_{n-1}$ . Abbiamo quindi che il metodo converge se

$$y(t_n) - u_n^* \xrightarrow{h \rightarrow 0} 0 \quad \text{e} \quad u_n^* - u_n \xrightarrow{h \rightarrow 0} 0$$

possiamo quindi analizzare i due contributi separatamente. Per il primo sviluppiamo  $y$  con Taylor e abbiamo

$$y(t_n) = y(t_{n-1}) + hy'(t_{n-1}) + \frac{1}{2}h^2y''(\xi) = y(t_{n-1}) + hf(t_{n-1}, y(t_{n-1})) + \frac{1}{2}h^2y''(\xi)$$

dato che  $y$  è soluzione del problema di Cauchy e dove  $t_{n-1} \leq \xi \leq t_n$ . Il contributo  $y(t_n) - u_n^*$  dell'errore risulta

$$y(t_n) - u_n^* = y(t_{n-1}) + hf(t_{n-1}, y(t_{n-1})) + \frac{1}{2}h^2y''(\xi) - y(t_{n-1}) - hf(t_{n-1}, y(t_{n-1})) = \frac{1}{2}h^2y''(\xi),$$

Abbiamo così ottenuto l'errore di troncamento locale per il metodo di Eulero in avanti, che è dato da

$$\tau_n(h) = \frac{1}{2}hy''(\xi) \quad t_{n-1} \leq \xi \leq t_n.$$

L'errore di troncamento globale è quindi

$$\tau(h) = \max_{n=0, \dots, N} |\tau_n(h)| \leq \frac{1}{2}h \max_{t \in I} |y''(t)| = ch \quad \text{con} \quad c = \frac{1}{2} \max_{t \in I} |y''(t)|.$$

Quello che otteniamo è quindi che il metodo di Eulero in avanti è consistente al prim'ordine, cioè che l'errore di troncamento globale converge a zero

$$\lim_{h \rightarrow 0} \tau(h) \leq \lim_{h \rightarrow 0} ch = 0.$$

Analizziamo la seconda parte dell'errore  $e_n$ , il termine  $u_n^* - u_n$ . Richiediamo che  $f$  sia una funzione Lipschitziana nel secondo argomento, ossia che esista un  $L > 0$  tale che

$$|f(t, y_1(t)) - f(t, y_2(t))| \leq L |y_1(t) - y_2(t)| \quad \forall t \in I.$$

Sfruttando questa proprietà abbiamo che l'errore è dato da

$$\begin{aligned} |u_n^* - u_n| &= |y(t_{n-1}) + hf(t_{n-1}, y(t_{n-1})) - u_{n-1} + hf(t_{n-1}, u_{n-1})| = \\ &= |y(t_{n-1}) - u_{n-1} + h[f(t_{n-1}, y(t_{n-1})) - f(t_{n-1}, u_{n-1})]| \leq \\ &\leq |y(t_{n-1}) - u_{n-1}| + hL |y(t_{n-1}) - u_{n-1}| = |e_{n-1}| + hL |e_{n-1}| = (1 + hL) |e_{n-1}|. \end{aligned}$$

E quindi abbiamo la stima  $|u_n^* - u_n| \leq (1 + hL) |e_{n-1}|$ . Globalmente possiamo quindi calcolare l'errore  $e_n$  come somma dei due contributi appena analizzati, abbiamo

$$\begin{aligned} |e_n| &= |[y(t_n) - u_n^*] + [u_n^* - u_n]| \leq |y(t_n) - u_n^*| + |u_n^* - u_n| \leq h\tau(h) + (1 + hL) |e_{n-1}| \leq \\ &\leq h\tau(h) + (1 + hL) [h\tau(h) + (1 + hL) |e_{n-2}|] \leq \dots \leq \\ &\leq h\tau(h) [1 + (1 + hL) + (1 + hL)^2 + \dots + (1 + hL)^{n-1}] = \\ &= h\tau(h) \frac{(1 + hL)^n - 1}{hL} = \tau(h) \frac{(1 + hL)^n - 1}{L}, \end{aligned}$$

avendo supposto che  $e_0 = y(t_0) - u_0 = 0$  e usando la relazione

$$\sum_{k=0}^{n-1} x^k = \frac{x^n - 1}{x - 1}.$$

Sfruttiamo ora la disuguaglianza  $1 + x \leq e^x$ , che nel nostro caso diventa  $1 + hL \leq e^{hL}$  e otteniamo

$$|e_n| \leq \tau(h) \frac{(1 + hL)^n - 1}{L} \leq \tau(h) \frac{e^{hLn} - 1}{L} = \tau(h) \frac{e^{L(t_f - t_0)} - 1}{L} = M\tau(h)$$

dove appunto  $M$  è una costante che dipende dal problema ma non da  $h$ . Abbiamo che  $|\tau(h)| < ch$  e quindi globalmente risulta  $|e_n| < Mch = \tilde{c}h$ , ovvero abbiamo dimostrato che il metodo di Eulero in avanti è convergente al prim'ordine.

Analogamente possiamo ricavare anche il metodo di Eulero all'indietro (implicito) una stima dell'errore del tipo

$$|e_n| \sim ch$$

ovvero convergenza lineare. Nel caso del metodo di Crank-Nicolson otteniamo invece un andamento quadratico,

$$|e_n| \sim ch^2.$$

I metodi visti precedentemente sono detti metodi ad un passo, perché ricavo la soluzione  $u_{n+1}$  conoscendo solamente la soluzione al passo  $u_n$ . Per tali metodi esiste una proprietà importante: tutti i metodi ad un passo consistenti sono zero-stabili. Possiamo quindi dire che per questa classe di metodi la consistenza equivale alla convergenza, dato che grazie al teorema di equivalenza abbiamo che, in generale, un metodo consistente e zero-stabile è convergente.

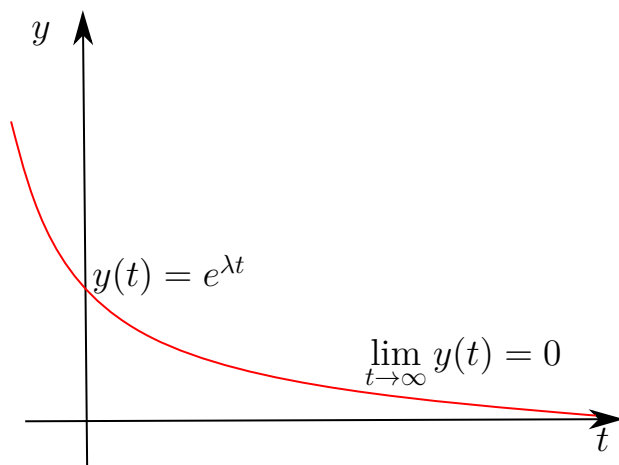
### Assoluta stabilità

L'assoluta stabilità di uno schema numerico per la risoluzione di un problema di Cauchy è un concetto di stabilità su intervalli illimitati. In altre parole vogliamo studiare cosa succede per  $n \rightarrow \infty$  mantenendo  $h$  fisso. Consideriamo il problema detto *problema modello* seguente

$$\begin{cases} y'(t) = \lambda y(t) \\ y(0) = 1 \end{cases}$$

dove  $\lambda < 0$  è un parametro dato e per semplicità reale (si potrebbe estendere l'analisi a  $\lambda$  complessi). La soluzione esatta del problema modello è calcolabile analiticamente ed è pari a

$$y(t) = e^{\lambda t}$$



Notiamo quindi che la soluzione esatta converge a zero per  $t \rightarrow \infty$ .

#### Definizione 8.4 - assoluta stabilità

Un metodo numerico per la soluzione di un problema di Cauchy è assolutamente stabile se, dato un passo  $h$ , otteniamo

$$\lim_{n \rightarrow \infty} u_n = 0$$

quando è applicato al problema modello discusso precedentemente.

Consideriamo il metodo di Eulero in avanti e applichiamo al problema modello, per  $u_0 = 1$  abbiamo

$$\begin{aligned} u_{n+1} &= u_n + hf(t_n, u_n) = u_n + h\lambda u_n = (1 + h\lambda)u_n = (1 + h\lambda)^2 u_{n-1} = \dots = \\ &= (1 + h\lambda)^{n+1} u_0 = (1 + h\lambda)^{n+1}. \end{aligned}$$

Possiamo quindi scrivere che il metodo di Eulero in avanti genera la sequenza delle soluzioni data da

$$u_n = (1 + \lambda h)^n \quad \Rightarrow \quad \lim_{n \rightarrow \infty} u_n = \lim_{n \rightarrow \infty} (1 + \lambda h)^n = 0 \quad \Longleftrightarrow \quad |1 + \lambda h| < 1.$$



Abbiamo quindi i seguenti casi da analizzare

$$\begin{aligned} 1 + h\lambda < 1 &\rightarrow h\lambda < 0 \rightarrow \forall h \\ 1 + h\lambda > -1 &\rightarrow h\lambda > -2 \rightarrow h < \frac{2}{|\lambda|} \end{aligned}$$

Quindi il metodo di Eulero in avanti risulta assolutamente stabile solo se prendo  $h$  sufficientemente piccolo, il metodo viene detto *condizionatamente* assolutamente stabile.

Analizziamo ora Eulero all'indietro applicato al problema modello

$$u_{n+1} = u_n + hf(t_{n+1}, u_{n+1}) = u_n + h\lambda u_{n+1} \rightarrow (1 - h\lambda)u_{n+1} = u_n$$

riorganizzando i termini abbiamo

$$u_{n+1} = \frac{1}{1 - h\lambda} u_n = \left( \frac{1}{1 - h\lambda} \right)^2 u_{n-1} = \dots = \left( \frac{1}{1 - h\lambda} \right)^{n+1} u_0 = \left( \frac{1}{1 - h\lambda} \right)^{n+1}.$$

Il denominatore risulta sempre maggiore strettamente di 1, dato che  $h$  e  $\lambda$  hanno segno opposto ( $\lambda < 0$ ). Quindi la condizione che

$$\frac{1}{1 - h\lambda} < 1$$

è sempre verificata per ogni valore di  $h$ , ovvero abbiamo che

$$\lim_{n \rightarrow \infty} u_n = 0 \quad \forall h.$$

Il metodo di Eulero all'indietro è quindi *incondizionatamente* assolutamente stabile. È possibile svolgere dei conti analoghi anche per il metodo di Crank-Nicolson ed ottenere che anche in questo caso il metodo risulta incondizionatamente assolutamente stabile. Infatti abbiamo

$$u_{n+1} = u_n + \frac{h}{2}[f(t_n, u_n) + f(t_{n+1}, u_{n+1})] = u_n + \frac{h\lambda}{2}(u_n + u_{n+1}),$$

riorganizzando i termini abbiamo

$$u_{n+1} = \frac{1 + \frac{h\lambda}{2}}{1 - \frac{h\lambda}{2}} u_n = \left( \frac{1 + \frac{h\lambda}{2}}{1 - \frac{h\lambda}{2}} \right)^2 u_{n-1} = \dots = \left( \frac{1 + \frac{h\lambda}{2}}{1 - \frac{h\lambda}{2}} \right)^{n+1}.$$

Il termine in parentesi risulta strettamente minore di 1 e quindi otteniamo che

$$\lim_{n \rightarrow \infty} u_n = 0 \quad \forall h.$$

Se considero ora problema di Cauchy più generale del problema modello, definito come

$$\begin{cases} y'(t) = f(t)y(t) \\ y(t_0) = 1 \end{cases}$$

diciamo che lo schema è incondizionatamente assolutamente stabile allora la soluzione numerica  $u_n \rightarrow 0$  se  $y(t) \rightarrow 0$  per  $t \rightarrow \infty$ . In questo caso nel problema modello  $\lambda$  verrebbe sostituita dal valore, non costante, dato da  $f$  che richiediamo sia  $f(t) < 0$  per ogni istante di tempo  $t$ . Richiediamo quindi che esistano  $\lambda_*$  e  $\lambda^*$ , entrambi positivi, tali che

$$-\lambda^* \leq f(t) \leq -\lambda_* \quad \forall t$$

allora il metodo di Eulero in avanti sarà stabile solo se il passo  $h$  verifica

$$h \leq \frac{2}{\lambda^*}.$$

La scelta di  $h$  deve quindi tener conto sia delle esigenze di accuratezza che degli eventuali limiti di stabilità. Infatti se  $h$  è piccolo per ragioni di accuratezza allora conviene utilizzare il metodo di Eulero in avanti perchè è il meno costoso e richiede comunque una limitazione sul passo  $h$ . Al contrario, se per ragioni di accuratezza posso scegliere un passo  $h$  ampio allora conviene utilizzare Eulero implicito che non ha limiti sul passo temporale.

## Sistemi di equazioni differenziali ordinarie

Consideriamo ora il caso di sistemi di equazioni differenziali ordinarie di ordine 1, che vengono espresse come

$$\begin{cases} y_1'(t) = f_1(t, y_1(t), y_2(t), \dots, y_m(t)) \\ y_2'(t) = f_2(t, y_1(t), y_2(t), \dots, y_m(t)) \\ \dots \\ y_m'(t) = f_m(t, y_1(t), y_2(t), \dots, y_m(t)) \end{cases}$$

con valori iniziali dati dalle seguenti condizioni

$$\begin{cases} y_1(t) = \bar{y}_{0,1} \\ y_2(t) = \bar{y}_{0,2} \\ \dots \\ y_m(t) = \bar{y}_{0,m} \end{cases}.$$

Il maniera compatta possiamo scrivere il sistema precedente introducendo

$$\begin{aligned} \mathbf{y}(t) &= [y_1(t), y_2(t), \dots, y_m(t)]^\top \\ \mathbf{f}(t, \mathbf{y}(t)) &= [f_1(t, \mathbf{y}(t)), f_2(t, \mathbf{y}(t)), \dots, f_m(t, \mathbf{y}(t))]^\top \\ \mathbf{y}_0 &= [\bar{y}_{0,1}, \bar{y}_{0,2}, \dots, \bar{y}_{0,m}]^\top \end{aligned}$$

Possiamo quindi scrivere il sistema precedente come

$$\begin{cases} \mathbf{y}'(t) = \mathbf{f}(t, \mathbf{y}(t)) \\ \mathbf{y}(t) = \mathbf{y}_0 \end{cases}$$

Da notare che un'equazione differenziale ordinaria di ordine superiore ad 1 può essere riscritta come un sistema di equazioni differenziali ordinarie del prim'ordine, come vedremo successivamente.

Possiamo applicare gli schemi visti nel caso scalare anche a sistemi di ODE. Il  $\theta$ -metodo viene scritto come

$$\mathbf{u}^{n+1} = \mathbf{u}^n + h[\theta \mathbf{f}(t^{n+1}, \mathbf{u}^{n+1}) + (1 - \theta) \mathbf{f}(t^n, \mathbf{u}^n)]$$

dove appunto il vettore  $\mathbf{u}^n$  rappresenta la soluzione approssimata di  $\mathbf{y}$  all'istante  $t^n = t^0 + nh$ . In generale abbiamo

$$\mathbf{u}^n = [u_1^n, u_2^n, \dots, u_m^n]^\top.$$

Le proprietà di consistenza, zero-stabilità e convergenza sono le stesse che nel caso scalare. Analogamente, le proprietà di assoluta stabilità sono equivalenti. Nel caso del problema modello generalizzato abbiamo

$$\begin{cases} \mathbf{y}'(t) = J(t)\mathbf{y}(t) \\ \mathbf{y}(t) = \mathbf{y}_0 \end{cases}$$

dove  $J$  è una matrice data. La condizione di assoluta stabilità per Eulero Esplicito in questo caso diventa

$$h < \frac{2}{|\lambda|} \quad \text{dove} \quad \lambda = -\max_{t \in [t_0, t_f]} \rho(J)$$

dove  $\rho(J)$  è il raggio spettrale delle matrici  $J$ . Nel caso in cui volessi utilizzare il metodo di Eulero implicito allora avrei

$$\mathbf{u}^{n+1} = \mathbf{u}^n + h\mathbf{f}(t^{n+1}, \mathbf{u}^{n+1})$$

che è un sistema di equazioni non-lineari in  $\mathbf{u}^{n+1}$ . Per risolvere quindi il problema differenziale devo utilizzare due cicli annidati, quello esterno risolve i passi temporali mentre quello interno è associato alla soluzione del problema non-lineare, ad esempio con il metodo di Newton.

## Equazioni differenziali di ordine $m$

Nel caso in cui abbiamo il seguente problema di Cauchy scalare di ordine  $m$

$$\begin{cases} y^{(m)}(t) = f(t, y(t), y'(t), \dots, y^{(m-1)}(t)) \\ y(t_0) = \bar{y}_0 \\ y'(t_0) = \bar{y}'_0 \\ \dots \\ y^{(m-1)}(t_0) = \bar{y}_0^{(m-1)} \end{cases}$$

tale sistema può sempre essere riformulato come un sistema di  $m$  equazioni differenziali ordinarie di ordine 1 introducendo incognite aggiuntive. Infatti, ponendo

$$\begin{cases} w_1(t) = y(t) \\ w_2(t) = y'(t) = w'_1(t) \\ w_3(t) = y''(t) = w'_2(t) \\ \dots \\ w_m(t) = y^{(m-1)}(t) = w'_{m-1}(t) \end{cases}$$

il sistema equivalente risulta

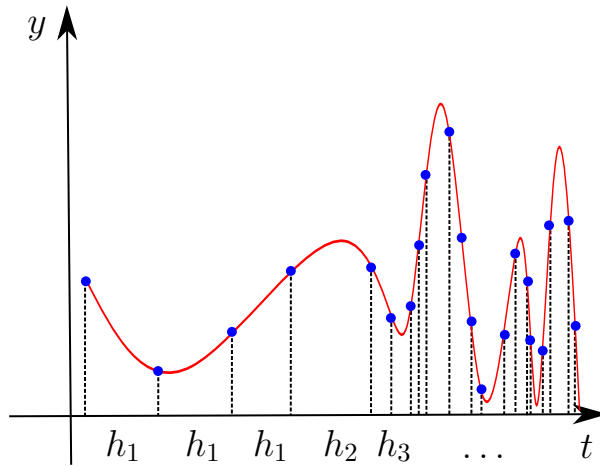
$$\begin{cases} w'_1(t) = w_2(t) \\ w'_2(t) = w_3(t) \\ \dots \\ w'_{m-1}(t) = w_m(t) \\ w'_m(t) = f(t, w_1(t), w_2(t), \dots, w_{m-1}(t)) \end{cases}$$

con condizioni iniziali date da

$$\begin{cases} w_1(t_0) = \bar{y}_0 \\ w_2(t_0) = \bar{y}'_0 \\ \dots \\ w_m(t_0) = \bar{y}_0^{(m)} \end{cases}.$$

## Adattività di passo (cenni)

Negli schemi numerici che abbiamo visto il passo  $h$  risulta definito a priori, e costante, tuttavia potrebbe risultare opportuno cambiare il passo  $h$  in base all'andamento della soluzione. Come mostrato in figura ha senso scegliere un  $h$  più piccolo dove la soluzione cambia più rapidamente: in corrispondenza di questi istanti di tempo mi aspetto infatti di commettere errori maggiori.



Per fare questa operazione ho bisogno di saper stimare l'errore al passo  $n$  detto  $e^n$ , e introdurre la semplice regola

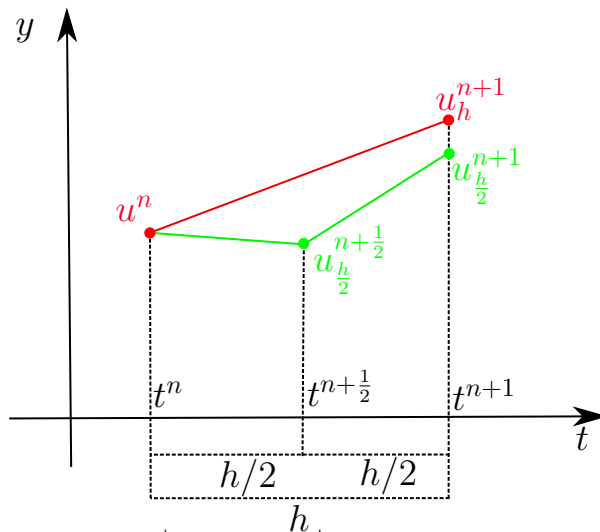
$$\begin{aligned} \text{se } e^n < \epsilon &\rightarrow h \text{ va bene} \\ \text{se } e^n > \epsilon &\rightarrow \text{devo ridurre } h \end{aligned}$$

Il problema è che per calcolare  $e^n = |y(t^n) - u^n|$  dovrei sapere la soluzione esatta  $y(t^n)$ . Quello che vorremmo costruire è uno stimatore dell'errore, ovvero un surrogato di  $e^n$  che può essere calcolato anche senza la conoscenza di  $y(t^n)$ . Consideriamo il metodo di Eulero esplicito, data la soluzione  $u^n$  al tempo  $t^n$ , la soluzione approssimata al tempo  $t^{n+1}$  con passo  $h$  è data da

$$u_h^{n+1} = u^n + hf(t^n, u^n)$$

mentre la soluzione approssimata al tempo  $t^{n+1}$  utilizzando due passi dimezzati,  $h/2$  è

$$\begin{aligned} u_{\frac{h}{2}}^{n+\frac{1}{2}} &= u^n + \frac{h}{2}f(t^n, u^n) \\ u_{\frac{h}{2}}^{n+1} &= u_{\frac{h}{2}}^{n+\frac{1}{2}} + \frac{h}{2}f(t^{n+\frac{1}{2}}, u_{\frac{h}{2}}^{n+\frac{1}{2}}) \end{aligned}$$



Possiamo quindi considerare  $e_h^n = |u_h^{n+1} - u_{\frac{h}{2}}^{n+1}|$  come uno stimatore dell'errore. In generale i metodi adattivi consentono di scegliere il passo temporale variabile in modo da controllare

l'errore. Per fare questo è necessario avere uno stimatore efficace dell'errore. Posso usare lo stesso metodo di avanzamento con due diversi valori del passo temporale, come visto in precedenza. Oppure posso usare due metodi di ordine diverso, come nel caso del comando Python `solve_ivp` di `scipy.integrate`.

## Metodi di alto ordine (cenni)

I metodi *a un passo* visti non possono fornire ordine di convergenza superiore a 2 (questo risultato è noto come *prima barriera di Dahlquist* dal matematico G.Dahlquist). Per ottenere ordini superiori si possono seguire due strade:

- considerare i cosiddetti metodi *multistep* che utilizzano più passi temporali
- considerare metodi *nonlineari*, come i metodi di Runge-Kutta.

I metodi multistep utilizzano più passi temporali per approssimare la derivata  $y'$ , o per approssimare il valore di  $f(t, y)$ . Ad esempio possiamo avere

$$u_{n+1} - \frac{4}{3}u_n + \frac{1}{3}u_{n-1} = \frac{2}{3}hf(t_{n+1}, u_{n+1}).$$

Ovviamente questi metodi a  $p$ -passi non possono essere utilizzati per costruire la soluzione nei primi istanti temporali, perché mancherebbero le condizioni iniziali. Bisogna quindi utilizzare un diverso metodo per costruire  $u_1, u_2$  ecc.

I metodi di Runge-Kutta, che possono essere impliciti o espliciti, sono metodi a più *stadi* che costruiscono la soluzione in modo nonlineare del tipo

$$u_{n+1} = u_n + hF(t_n, u_n, h; f)$$

dove  $F$  è la funzione incremento. Ad esempio, un semplice Runge-Kutta esplicito a due stadi, noto come metodo di Heun, è:

$$\begin{aligned} u^* &= u_n + hf(t_n, u_n) \\ u_{n+1} &= u_n + \frac{h}{2}(f(t_n, u_n) + f(t_{n+1}, u^*)) \end{aligned}$$

Notiamo la differenza fra la formula precedente e il metodo di Crank-Nicolson. In generale un metodo di Runge-Kutta a  $s$  stadi è definito come segue,

$$\begin{aligned} F(t_n, u_n, h; f) &= \sum_{i=1}^s b_i K_i \\ K_i &= f(t_n + c_i h, u_n + \sum_{j=1}^s a_{ij} K_j) \quad i = 1, \dots, s \end{aligned}$$

dove i coefficienti  $a_{ij}, b_i, c_i$  che devono soddisfare alcune condizioni perché il metodo sia consistente, in particolare richiediamo che

$$c_i = \sum_{j=1}^s a_{ij} \quad i = 1, \dots, s \quad \text{e} \quad \sum_{i=1}^s b_i = 1$$

Se i coefficienti  $a_{ij}$  sono nulli per  $j \geq i$  allora il metodo è esplicito, altrimenti risulta implicito. Essendo i metodi Runge-Kutta ad un passo allora la consistenza è equivalente alla convergenza, quindi se un metodo ha consistenza di ordine  $p$  allora avrà anche convergenza di ordine  $p$ . Abbiamo il seguente risultato per gli schemi espliciti

**Teorema 8.2**

Un metodo Runge-Kutta esplicito a  $s$  stadi non può avere ordine maggiore di  $s$ . Inoltre, non esistono metodi Runge-Kutta espliciti a  $s$  stadi con ordine  $s$  se  $s \geq 5$ .

Un esempio di metodo del quarto ordine esplicito è dato dal seguente

$$u_{n+1} = u_n + \frac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4)$$
$$K_1 = f_n \quad K_2 = f\left(t_n + \frac{h}{2}, u_n + \frac{h}{2}K_1\right) \quad K_3 = f\left(t_n + \frac{h}{2}, u_n + \frac{h}{2}K_2\right) \quad K_4 = f(t_{n+1}, u_n + hK_3)$$

Essendo metodi ad un passo si prestano bene ad essere usati per una strategia adattiva, sovente si usano due metodi di ordine diverso per stimare l'errore commesso e nel caso diminuire o aumentare il passo temporale.

La funzione Python `solve_ivp` di `scipy.integrate` implementa i metodi di Runge-Kutta `method = RK45`, di default, e `method = RK23` entrambi con passo temporale  $h$  adattivo, il primo usando i metodi di ordine 4 e 5 mentre il secondo di ordine 2 e 3.