

Equazioni differenziali ordinarie

Lucia Gastaldi

DICATAM - Sez. di Matematica,
<http://www.ing.unibs.it/gastaldi/>

Indice

- 1 Il problema di Cauchy
 - Esistenza, unicità e dipendenza continua dai dati
- 2 Metodi numerici
 - Il metodo di Eulero
 - Il metodo di Crank-Nicolson
 - Metodi di tipo Runge-Kutta
 - Metodi a più passi
- 3 Stabilità assoluta
 - Regione di assoluta stabilità
- 4 Function di Matlab per la soluzione di ODE
 - Solutori
 - Sistema di equazioni differenziali
 - Risultati non attendibili di ode45
- 5 Applicazioni

Il problema di Cauchy

Problema di Cauchy

Sia I un intervallo di \mathbb{R} , data $f : [t_0, T] \times \mathbb{R} \rightarrow \mathbb{R}$, trovare $y : [t_0, T] \rightarrow \mathbb{R}$ derivabile tale che

$$(C) \begin{cases} y'(t) = f(t, y(t)) & t \in [t_0, T] \\ y(t_0) = y_0. \end{cases}$$

Teorema

Se la funzione f è **continua** in $[t_0, T] \times \mathbb{R}$ e **lipschitziana** rispetto a y , cioè esiste $L > 0$ tale che

$$|f(t, y_1) - f(t, y_2)| \leq L|y_1 - y_2| \quad \forall t \in [t_0, T], \quad \forall y_1, y_2 \in \mathbb{R},$$

allora esiste una ed una sola $y : [t_0, T] \rightarrow \mathbb{R}$ soluzione del **problema di Cauchy**.

Dipendenza continua dai dati

Consideriamo il seguente problema:

$$\begin{cases} z'(t) = f(t, z(t)) + \varepsilon_1 g(t) & \text{per } t \in [t_0, T] \\ z(t_0) = y_0 + \varepsilon_0 \end{cases}$$

essendo $g \in C^0([t_0, T])$ con $M = \max_{t \in [t_0, T]} |g(t)|$.

Definizione

Il problema di Cauchy si dice **stabile** se per ogni **perturbazione** $(\varepsilon_0, \varepsilon_1)$ tale che:

$$|\varepsilon_0| < \varepsilon, \quad |\varepsilon_1| < \varepsilon, \quad \text{con } \varepsilon > 0,$$

$$\text{vale: } \exists C > 0 : \quad \max_{t \in [t_0, T]} |y(t) - z(t)| < C\varepsilon.$$

Dipendenza continua dai dati

Proposizione

Se $f : [t_0, T] \times \mathbb{R} \rightarrow \mathbb{R}$ è continua su $[t_0, T] \times \mathbb{R}$ e lipschitziana rispetto a y essendo L la costante di Lipschitz, cioè se vale

$$|f(t, y_1) - f(t, y_2)| \leq L|y_1 - y_2| \quad \forall (t, y_1), (t, y_2) \in [t_0, T] \times \mathbb{R}.$$

Allora il problema di Cauchy (C) è stabile e vale la seguente maggiorazione

$$\max_{t \in [t_0, T]} |y(t) - z(t)| \leq |\varepsilon_1| \max_{t \in I} |g(t)| \frac{e^{L(T-t_0)} - 1}{L} + |\varepsilon_0| e^{L(T-t_0)}.$$

Il metodo di Eulero esplicito

Consideriamo:

- una partizione dell'intervallo $I = [t_0, T]$ in un numero finito di intervalli N ;
- $h = (T - t_0)/N$ **passo di discretizzazione**;
- $t_n = t_0 + hn$

In ogni punto t_n si cerca un valore u_n che *approssimi* il valore di $y_n = y(t_n)$.

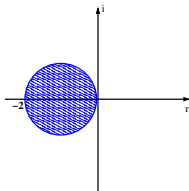
Metodo di Eulero in avanti

$$u_{n+1} = u_n + hf(t_n, u_n) \quad n = 0, 1, \dots, N_h.$$

- Metodo esplicito.
- Metodo del primo ordine:

$$\tau_n(h) = \frac{h}{2} y''(\xi_n).$$

- **Regione di stabilità assoluta:**
 $|1 + h\lambda| < 1.$



Esercizio 1

Dato $N \in \mathbb{R}$. Si consideri passo costante $h = (T - t_0)/N$.
Scrivere un programma che realizzi il metodo di Eulero esplicito a passo costante mediante una function

```
function [t,u]=euleroesp(f,t0,T,y0,N)
```

dove

- t , u sono i vettori che contengono i valori di t_n e u_n per $n = 0, \dots, N$.
- f è il nome di una function che contiene l'espressione di $f(t, y)$ in funzione di t , y .
- $t0$, T sono gli estremi dell'intervallo I .
- $y0$ è il valore iniziale.
- N il numero dei passi da effettuare a passo costante.

Traccia dell'esercizio

- Calcolare $h=(T-t_0)/N$.
- Calcolare i punti del vettore t con il comando `t=linspace(t0,T,N+1)`.
- Inizializzare il vettore u con il valore iniziale y_0 ($u=y_0$).
- Ciclo `for n=1:N`.
 - Calcola il valore della componente $u(n+1)$
 $u(n+1)=u(n)+hf(t(n),u(n));$

Esercizio 2

Sia $N=[20 \ 40 \ 60 \ 80 \ 100 \ 150 \ 200 \ 400]$.

Scrivere un programma di tipo script che per ogni valore di N :

- calcola, usando la function dell'esercizio precedente, la soluzione del seguente problema di Cauchy

$$\begin{cases} y'(t) = t - 2y & 0 \leq t \leq 20 \\ y(0) = 0.75. \end{cases}$$

- riporta il grafico della soluzione esatta e della soluzione approssimata in una stessa figura, essendo la soluzione esatta

$$y(t) = e^{-2t} + \frac{1}{2}t - \frac{1}{4};$$

- calcola l'errore: $err(N) = \max_{1 \leq n \leq N} |u_n - y(t_n)|$.

Fare il grafico in scala bilogarithmica dell'errore in funzione di N .

Traccia dell'esercizio

- Inserire i dati della equazione differenziale insieme alla soluzione esatta.
- Assegnare N .
- Per ciascun valore di N :
 - calcolare la soluzione dell'equazione differenziale con il comando mediante la function `euleroesp` implementata;
 - valutare la soluzione esatta nel vettore t ;
 - plottare la soluzione esatta e quella approssimata;
 - calcolare l'errore: `E(i)=norm(sol-u,inf)` (i rappresenta la variabile del ciclo).
- Plottare in scala bilogarithmica l'errore e l'andamento teorico $1/N$ con il comando: `loglog(N,E,N,1./N)`.
- Calcolare l'ordine del metodo
$$p = (\log(E(2:end)) - \log(E(1:end-1))) \dots$$
$$\dots / (\log(N(1:end-1)) - \log(N(2:end)))$$

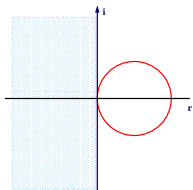
Il metodo di Eulero implicito

$$u_{n+1} = u_n + hf(t_{n+1}, u_{n+1}) \quad n = 0, 1, \dots, N_h.$$

- Metodo implicito.
- Metodo del primo ordine:

$$\tau_n(h) = \frac{h}{2} y''(\xi_n).$$

- **Regione di stabilità assoluta:**
 $|1 - h\lambda| > 1.$



Osservazione

ad ogni passo si deve risolvere l'**equazione non lineare**

$$\xi - u_n - hf(t_{n+1}, \xi) = 0,$$

usando uno dei metodi per il calcolo degli zeri di una funzione.

La function `euleroimp`

La function `euleroimp` risolve un'equazione differenziale ordinaria con il metodo di **Eulero implicito**.

L'equazione non lineare viene risolta usando la function `fzero` per la ricerca degli zeri di funzione.

```
[t,u]=euleroimp(f,t0,T,y0,N)
```

Il metodo di Crank-Nicolson

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

- Metodo implicito.
- Metodo del secondo ordine: $\tau_n(h) = -\frac{h^2}{12}y'''(\xi_n)$.
- **Regione di stabilità assoluta:** $\forall h\lambda < 0$.

Derivazione: dalla formula di quadratura dei trapezi si ottiene:

$$\begin{aligned}y(t_{n+1}) &= y(t_n) + \int_{t_n}^{t_{n+1}} f(\tau, y(\tau))d\tau \\ &\cong y(t_n) + \frac{h}{2}(f(t_n, y(t_n)) + f(t_{n+1}, y(t_{n+1}))).\end{aligned}$$

La function `cranknic`

La function `cranknic` risolve un'equazione differenziale ordinaria con il metodo di **Crank-Nicolson**.

L'equazione non lineare viene risolta usando la function `fzero` di Matlab.

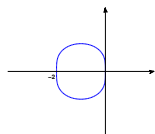
```
[t,u]=cranknic(f,t0,T,y0,N)
```

Il metodo di Heun

Metodo di Heun

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

- Metodo esplicito.
- Metodo del secondo ordine:
 $\tau_n(h) = Ch^2 y'''(\xi_n)$.
- **Regione di stabilità assoluta:**
 $-2 < h\lambda < 0$.



Derivazione: dalla formula di quadratura dei trapezi

$$\begin{aligned} y(t_{n+1}) &= y(t_n) + \int_{t_n}^{t_{n+1}} f(t, y(t)) dt \\ &\approx y(t_n) + \frac{h}{2} (f(t_n, u_n) + f(t_{n+1}, y(t_{n+1}))). \end{aligned}$$

Si valuta $y(t_{n+1})$ con Eulero esplicito $y(t_{n+1}) \approx u_n + hf(t_n, u_n)$

Il metodo di Runge-Kutta del 4° ordine

Metodo di Runge-Kutta del 4° ordine

$$u_{n+1} = u_n + \frac{h}{6} (K_1 + 2K_2 + 2K_3 + K_4)$$

$$K_1 = f(t_n, u_n)$$

$$K_2 = f\left(t_n + \frac{h}{2}, u_n + \frac{h}{2}K_1\right)$$

$$K_3 = f\left(t_n + \frac{h}{2}, u_n + \frac{h}{2}K_2\right)$$

$$K_4 = f(t_n + h, u_n + hK_3)$$

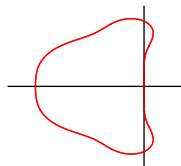
- Metodo esplicito del quarto ordine:

$$\tau_n(h) = Ch^4 y^{(5)}(\xi_n).$$

- **Regione di stabilità assoluta:**

$$|1 + q + q^2/2 + q^3/12 + q^4/24| < 1$$

dove $q = h\lambda$ ($-2.78 < \operatorname{Re} q < 0$).



Convergenza dei metodi numerici

Esercizio 3

Analizzare il comportamento dell'errore in funzione di h per almeno uno dei metodi seguenti: Eulero implicito, Heun, Crank-Nicolson e Runge-Kutta. A tal scopo usare i comandi

```
[t,u]=euleroimp(f,t0,T,y0,N)
```

```
[t,u]=Heun(f,t0,T,y0,N)
```

```
[t,u]=cranknic(f,t0,T,y0,N).
```

```
[t,u]=RK4(f,t0,T,y0,N).
```

Propagazione degli errori di macchina

Esercizio 4

Plottare in scala bilogarithmica l'errore di approssimazione che si ottiene risolvendo con il metodo di Runge-Kutta del 4° ordine la seguente equazione differenziale:

$$y' = \frac{y}{t+1} + 5 \frac{t+1}{1+25t^2} \quad \text{per } t \in [0, 1], \quad y(0) = 0$$

la cui soluzione esatta è $y(t) = (t+1) \arctan(5t)$.

Usare i seguenti valori per il numero di passi

$N = [10 \ 20 \ 40 \ 80 \ 125 \ 250 \ 500 \ 750]$

$N = [N \ 1000 \ 5000 \ 7500 \ 10000 \ 10500 \ 11000 \ 11500 \ 12000]$.

I metodi di Adams espliciti

Metodo di Adams-Bashforth a due passi

$$u_{n+1} = u_n + \frac{h}{2}(3f(t_n, u_n) - f(t_{n-1}, u_{n-1}))$$

- Metodo del secondo ordine: $\tau_n(h) = Ch^2y^{(3)}(\xi_n)$.

Metodo di Adams-Bashforth a tre passi

$$u_{n+1} = u_n + \frac{h}{12}(23f(t_n, u_n) - 16f(t_{n-1}, u_{n-1}) + 5f(t_{n-2}, u_{n-2}))$$

- Metodo del terzo ordine: $\tau_n(h) = Ch^3y^{(4)}(\xi_n)$.

I metodi di Adams impliciti

Metodo di Adams-Moulton a due passi

$$u_{n+1} = u_n + \frac{h}{12}(5f(t_{n+1}, u_{n+1}) + 8f(t_n, u_n) - f(t_{n-1}, u_{n-1}))$$

- Metodo del secondo ordine: $\tau_n(h) = Ch^2y^{(3)}(\xi_n)$.

Metodo di Adams-Moulton a tre passi

$$u_{n+1} = u_n + \frac{h}{24}(9f(t_{n+1}, u_{n+1}) + 19f(t_n, u_n) - 5f(t_{n-1}, u_{n-1}) + f(t_{n-2}, u_{n-2}))$$

- Metodo del terzo ordine: $\tau_n(h) = Ch^3y^{(4)}(\xi_n)$.

Backward Differentiation Formulae

BDF a due passi

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

- Metodo implicito del secondo ordine: $\tau_n(h) = Ch^2y^{(3)}(\xi_n)$.

BDF a tre passi

$$u_{n+1} = \frac{18}{11}u_n - \frac{9}{11}u_{n-1} + \frac{2}{11}u_{n-2} + \frac{6}{11}hf(t_{n+1}, u_{n+1})$$

- Metodo implicito del terzo ordine: $\tau_n(h) = Ch^3y^{(4)}(\xi_n)$.

Regione di assoluta stabilità

Esercizio 5

Risolvere il problema modello

$$y' = \lambda y, t \in [0, 20], \quad y(0) = 1$$

per $\lambda = -1, -5, -10$.

- Tenuto conto dei risultati sulla regione di assoluta stabilità, trovare sperimentalmente il valore più piccolo di N per il quale le function `euleroesp`, `Heun` e `RK4` forniscono una soluzione tendente a 0 al crescere di t_n .
- Posto h^* il valore di h corrispondente al valore di N trovato, verificare che il valore di λh^* si avvicina al limite massimo della regione di assoluta stabilità.
- Verificare che le function `euleroimp` e `cranknic` danno sempre una soluzione tendente a 0.

Controllo del passo per stabilità assoluta

Metodo di Eulero stabilizzato

Posto $u_0 = y_0$, $h_0 = 2\alpha/|\lambda(t_0)|$, per ogni n tale che $t_n < T$ si calcola

- $t_{n+1} = t_n + h_n$,
- $u_{n+1} = u_n + h_n \lambda(t_n) u_n$,
- $h_{n+1} = 2\alpha/|\lambda(t_n + 1)|$, dove $0 < \alpha < 1$.

La function `eestab` calcola la soluzione numerica del problema di Cauchy con il metodo di Eulero stabilizzato mediante il seguente comando:

```
[t,u]=eestab(f,t0,T,y0,df,alpha,hmax)
```

essendo

- `df` l'espressione analitica di $\partial f / \partial y$
- `alpha` il parametro α che compare nel metodo
- `hmax` il valore massimo che può assumere h per motivi di accuratezza.

Esercizio 6

Risolvere l'equazione differenziale

$$y'(t) = -\frac{10 - t^2}{1 + t^4}y(t), \quad t \in [0, 5]; \quad y(0) = 1.$$

con il metodo di Eulero in avanti **stabilizzato**.

Usare i seguenti valori $\alpha = .9, .8, .75, 2/3, 1/2, 2/5, 1/3, 1/6$.

Riportare in un grafico la funzione $|\lambda(t)| = \frac{10-t^2}{1+t^4}$.

Confrontare con i risultati ottenuti a passo costante con $h = h_0 = 2 * \alpha / |\lambda(0)|$. (Per trovare N , calcolare $N = (T - t_0) / h_0$.)

Riportare in una tabella il numero di passi impiegati con Eulero stabilizzato e Eulero a passo costante.

Esercizio 7

Sono date le seguenti equazioni differenziali:

$$y' = \frac{4y^2}{1+y^2} - 4y + t/2 \quad y(0) = 1$$
$$y' = -\frac{4y^2}{1+y^2} - 4y + t/2 \quad y(0) = 1$$

Per ciascuna equazione differenziale,

- determinare il valore di

$$h_0 = \frac{2}{\max_{(t,y)} \left| \frac{\partial f}{\partial y} \right|}.$$

- Risolvere l'equazione differenziale nell'intervallo $[0, 50]$ usando il metodo di Eulero esplicito a passo costante con i seguenti valori di h : $h = 0.6$, $h = 0.4$, $h = 0.2$ e $h = 0.05$. Si osservi che non per tutti i valori $h < h_0$.
- Risolvere l'equazione differenziale usando il metodo di Eulero stabilizzato scegliendo α opportunamente.

Problema stiff

$$\begin{aligned}y' &= \lambda(y - g(t)) + g'(t) \quad \text{per } t \in [0, 4] \\y(0) &= y_0\end{aligned}$$

Soluzione esatta $y(t) = (y_0 - g(0))e^{\lambda t} + g(t)$.

Posto $\lambda = -25$, $y_0 = 4$ e $g(t) = t^2$, calcolare la soluzione mediante il metodo di Eulero esplicito con $N = 120$. Calcolare l'errore rispetto alla soluzione esatta.

Per trovare N in modo che l'errore sia inferiore a $1/10$, si deve risolvere

$$C \frac{h}{2} |y''(t)| \leq \frac{1}{10} \quad h \approx \frac{0.2}{1 + 2500e^{-25t}}$$

Scegliere $N = 500$ quindi $h = 4/500 = 0.008$.

Si nota che per $t > 0.4$ la soluzione era già sufficientemente accurata con la scelta $N = 120$. Quindi risolvere nell'intervallo $[0, 0.4]$ con $N = 50$ e prolungare la soluzione nell'intervallo $[0.4, 4]$ con $N = 40$ che dovrebbe garantire l'accuratezza richiesta.

La soluzione numerica ha oscillazioni che aumentano al crescere di t perché la condizione di assoluta stabilità non è soddisfatta. Con $N = 45$ le oscillazioni spariscono e l'errore è uguale a quello ottenuto con $N = 500$ sull'intero intervallo.

Risolutori di equazioni differenziali ordinarie

Problemi non stiff

- ode45 Metodo di Runge-Kutta (4,5).
- ode23 Metodo di Runge-Kutta (2,3).
- ode113 Metodo di Adams-Bashforth-Moulton PECE.

Problemi stiff

- ode15s Metodo multistep basato su una formula di tipo BDF.
- ode23s Metodo ad un passo.

Opzioni

- odeset Crea o modifica le OPTIONS.

Come si risolve una equazione differenziale usando i solutori di Matlab

- Scrivere una function che accetta due argomenti t e y e restituisce il valore della funzione

```
function dy=F(t,y)
dy=(1-t*y-t^2*y^2)/t^2;
```

```
F=@(t,y) (1-t*y-t^2*y^2)/t^2;
```

- Applicare un solutore mediante i comandi
`[t,u] = ode23 (@F, [t0 T], y0)`
`[t,u] = ode23 (F, [t0 T], y0)`
- Usare il comando `plot` per vedere i risultati: `plot(t,u)`

Come usare i solutori di ODE in Matlab

Il comando più semplice per risolvere un'equazione differenziale è:
`[t,u] = solver (odefun,tspan, y0)`

Input

<code>odefun</code>	function in cui si valuta $f(t, y)$
<code>tspan</code>	vettore contenente gli estremi di integrazione
<code>y0</code>	dato iniziale (vettore colonna)

Output

<code>t</code>	vettore colonna degli istanti t in cui viene calcolata la soluzione approssimata
<code>u</code>	array contenente la soluzione, le righe sono le componenti di y ad un certo istante t .

Argomenti aggiuntivi

```
[t,u] = solver (odefun,tspan, y0,options)
```

`options` struttura che contiene i parametri per cambiare le proprietà di default del solutore

Per definire le `options` si usa il comando `odeset`.

```
>> odeset
```

fornisce i valori di default e il nome delle variabili che si possono definire.

odeset

Variabile	default	descrizione
RelTol	1.e-3	tolleranza per l'errore relativo
AbsTol	1.e-6	tolleranza per l'errore assoluto
MaxStep	tspan /10	valore massimo per il passo
InitialStep	calcolato	passo iniziale scelto
OutputFcn	Function	controlla l'output
OutputFcn	odeplot	grafico in funzione di t
OutputFcn	odephas2	plot del piano delle fasi
OutputFcn	odephas3	plot del piano delle fasi in 3D
OutputSel	vettore di interi	specifica le componenti del vettore soluzione che si vogliono come output

Sistema di equazioni differenziali

Problema - Modello preda-predatore

Siano a , b , c e d numeri reali positivi. Cercare $y_1(t)$ e $y_2(t)$ tali che risolvano nell'intervallo $[0, 7]$ il seguente sistema di equazioni differenziali ordinarie:

$$\begin{cases} y_1' = (a - by_2)y_1 \\ y_2' = (-c + dy_1)y_2 \\ y_1 = \alpha \\ y_2 = \beta. \end{cases}$$

Porre: $a = 1$, $b = 1$, $c = 2$, $d = 3$

$\alpha = 0.5, 0.7, 1.4, 2.6$

$\beta = 1$

Come si risolve il sistema

- Scrivere una function che accetta due argomenti t e y e restituisce il valore della funzione a valori vettoriali

```
function dy=lotkavolterra(t,y)
a=1; b=1; c=2; d=3;
dy=[(a-b*y(2))*y(1); (-c+d*y(1))*y(2)];
```

- Applicare un solutore mediante il comando
`[t,u] = ode45 (@lotkavolterra, [0 7], [α ; β])`
- Usare il comando `plot` per vedere i risultati:
`plot(t,u)` oppure `plot(u(:,1),u(:,2))`
- Per ottenere il piano delle fasi si può procedere come segue:
`options=odeset('OutputFcn','odephas2')`
`[t,u]=ode45(@lotkavolterra,[0 7],[α ; β],options)`

Equazione differenziale di ordine n

Problema

Sia $f : I \times \mathbb{R}^n \rightarrow \mathbb{R}$. Consideriamo l'equazione differenziale:

$$\left\{ \begin{array}{l} y^{(n)}(t) = f(t, y(t), y'(t), \dots, y^{(n-1)}(t)) \quad t \in I \\ y(t_0) = \alpha_1 \\ y'(t_0) = \alpha_2 \\ \dots \\ y^{(n-1)}(t_0) = \alpha_n \end{array} \right.$$

Equazione differenziale di ordine n

Il problema di Cauchy per l'equazione differenziale di ordine n è equivalente ad un sistema differenziale del primo ordine.

Si pone

$$y_1(t) = y(t), \quad y_2(t) = y'(t), \quad \dots, \quad y_n(t) = y^{(n-1)}(t).$$

Osserviamo che

$$\begin{aligned} y_2(t) &= y'(t) = y_1'(t), \\ &\dots \\ y_n(t) &= y^{(n-1)}(t) = y_{n-1}'(t). \end{aligned}$$

Quindi si ottiene il seguente sistema di equazioni differenziali

$$\left\{ \begin{array}{l} y_1'(t) = y_2(t) \\ y_2'(t) = y_3(t) \\ \dots \\ y_n'(t) = f(t, y_1(t), y_2(t), \dots, y_n(t)) \\ \\ y_1(t_0) = \alpha_1 \\ y_2(t_0) = \alpha_2 \\ \dots \\ y_n(t_0) = \alpha_n \end{array} \right.$$

Equazione lineare del secondo ordine a coefficienti costanti

Siano δ (smorzamento) e ω_n (pulsazione naturale) due numeri reali, si consideri il seguente problema di Cauchy:

$$\begin{cases} y'' + 2\delta\omega_n y' + \omega_n^2 y = 1 \\ y(0) = \alpha, \quad y'(0) = \beta \end{cases}$$

La soluzione per $\alpha = \beta = 0$ e $|\delta| < 1$ è:

$$y(t) = 1 - Ae^{-\delta\omega_n t} \sin(\omega t + \varphi)$$

essendo

$$A = \frac{1}{\sqrt{1 - \delta^2}}, \quad \omega = \omega_n \sqrt{1 - \delta^2}, \quad \varphi = \arcsin(\delta).$$

Posto $z_1 = y$, $z_2 = y'$, l'equazione si riconduce al seguente sistema del primo ordine:

$$\begin{cases} z_1' = z_2 \\ z_2' = -2\delta\omega_n z_2 - \omega_n^2 z_1 + 1 \\ z_1(0) = \alpha, \quad z_2(0) = \beta \end{cases}$$

Esercizio 8

Posto $\omega_n = 1$, e $\alpha = \beta = 0$, calcolare, nell'intervallo $[0, 50]$, la soluzione dell'equazione differenziale mediante la function `ode45` e il metodo di Eulero esplicito con passo $h = 0.02$. Considerare i seguenti valori di δ nell'intervallo $[0, 2]$:

`delta=[0,0.1,0.2,0.5,1,1.5,2]`. Plottare le soluzioni ottenute in un grafico in funzione di t e rappresentarle nel piano delle fasi.

Calcolare inoltre:

- il valore finale;
- la massima sovraelongazione: differenza fra il valore massimo e il valore finale;
- Istante di massima sovraelongazione T_m : istante in cui si raggiunge il massimo.

find

Il comando **find** individua gli **indici** degli elementi di un vettore diversi da zero

$$K = \text{find}(X)$$

individua il vettore **K** degli indici **I** per cui $X(I) \neq 0$.

Si ricorda che una espressione logica è **vera** se ha valore diverso da zero.

Esempio

Posto $x = \text{linspace}(0, 2 * \pi)$ e $y = \sin(x)$, il comando

$$K = \text{find}(y == 1)$$

trova l'indice **K** per cui si ha $y(K) = 1$. La componente $x(K)$ è quindi il valore di **x** per cui vale $\sin(x(K)) = 1$.

Sistema stiff

Si consideri il sistema $y' = Ay$ per $t \in [0, 100]$ con $y(0) = (1, 1)^T$ e

$$A = \begin{pmatrix} 0 & 1 \\ -\lambda_1\lambda_2 & \lambda_1 + \lambda_2 \end{pmatrix} \quad \text{con } \lambda_1 \ll \lambda_2 < 0.$$

La soluzione è $y(t) = (C_1 e^{\lambda_1 t} + C_2 e^{\lambda_2 t}, C_1 \lambda_1 e^{\lambda_1 t} + C_2 \lambda_2 e^{\lambda_2 t})^T$ con

$$C_1 = \frac{\lambda_2 - 1}{\lambda_2 - \lambda_1}, \quad C_2 = \frac{1 - \lambda_1}{\lambda_2 - \lambda_1}.$$

Posto $\lambda_1 = -100$ e $\lambda_2 = -1$ calcolare la soluzione del sistema con il metodo di Eulero esplicito usando $h = 0.0207$, $h = 0.02$ e $h = 0.01$.

Equazione di van der Pol

$$y'' - \mu(1 - y^2)y' + y = 0 \quad \text{dove } \mu > 0.$$

L'equazione si riduce al seguente sistema:

$$\begin{cases} y_1'(t) = y_2(t) \\ y_2'(t) = \mu(1 - y_1^2)y_2 - y_1 \end{cases}$$

Si costruisce la function che definisce il sistema:

```
function dy = vdpol(t,y,mu)
dy = [y(2); mu*(1-y(1)^2)*y(2)-y(1)];
```

Risolvere i seguenti valori $\mu = 0.1, 1, 10, 100$, risolvere il sistema usando `ode23`, `ode45`, `ode23s` e `ode15s`

```
[t,u]=odeXX(@vdpol,[0 100],[1;1],[ ],mu)
```

Riportare in una tabella i passi di integrazione dei vari metodi al variare di μ .

Risultati non attendibili di ode45

Esercizio 9

Si consideri la seguente equazione differenziale:

$$y'(t) = 100 - y(t) \quad \text{per } t \in [0, 200], \quad y(0) = 0,$$

la cui soluzione è $y(t) = 100(1 - e^{-t})$.

- Calcolare la soluzione con `ode45`.
- Confrontare la soluzione con la soluzione esatta calcolando l'errore relativo al variare del tempo.
- Calcolare la soluzione ottenuta a passo costante con `RK4` con 250 passi e l'errore relativo al variare del tempo.
- Calcolare la soluzione con `ode15s`, confrontarla con le soluzioni precedentemente ottenute e calcolare l'errore relativo al variare del tempo.
- Verificare il comportamento della function `ode45` plottando i valori dei passi.

Oscillatori accoppiati

Consideriamo due oscillatori, ciascuno dei quali ha una frequenza naturale data dalle costanti ω_1 e ω_2 . Lo stato degli oscillatori è descritto dall'angoli di fase θ_1 e θ_2 . Le seguenti equazioni descrivono l'accoppiamento dei due oscillatori:

$$\begin{aligned}\theta_1' &= \omega_1 + k_1 \sin(\theta_2 - \theta_1) \\ \theta_2' &= \omega_2 + k_2 \sin(\theta_1 - \theta_2).\end{aligned}$$

Le costanti k_1 e k_2 indicano la forza delle costanti di accoppiamento. Questo modello può essere usato come punto di partenza per lo studio dei processi di sincronizzazione di fase.

In particolare consideriamo il seguente sistema:

$$\begin{aligned}\theta_1' &= 1 + \sin(\theta_2 - \theta_1) \\ \theta_2' &= 1.5 + \sin(\theta_1 - \theta_2) \\ \theta_1(0) &= 3, \quad \theta_2(0) = 0.\end{aligned}$$

Usando `ode45` calcolare la soluzione del sistema nell'intervallo $[0, 8]$.
Calcolare la soluzione del sistema nell'intervallo $[0, 200]$.

Ripetere la simulazione nell'intervallo $[0, 200]$ usando `ode15s`.

Esercizio 10

Risolvere usando le function di Matlab il problema di Cauchy e verificare le affermazioni sul comportamento della soluzione.

$$\begin{cases} y' = (\sin(t))^2 e^y (y - 1)(y - 2) & t \in [0, 10] \\ y(0) = \alpha \end{cases}$$

Si osserva che valgono le seguenti relazioni:

- Se $\alpha < 1$ allora $y(t) < 1$ per ogni $t > 0$;
- Se $\alpha = 1$ allora $y(t) = 1$ per ogni $t > 0$;
- Se $1 < \alpha < 2$ allora $1 < y(t) < 2$ per ogni $t > 0$;
- Se $\alpha = 2$ allora $y(t) = 2$ per ogni $t > 0$;
- Se $\alpha > 2$ allora $y(t) > 2$ per ogni $t > 0$.

Esercizio 11

Risolvere usando le function di Matlab il problema di Cauchy:

$$\begin{cases} y' = \frac{1}{t-1} + \sqrt{y^2 + 2} & t \in [0, 1] \\ y(0) = 1 \end{cases}$$

Si osserva che la funzione $f(t, y)$ tende all'infinito per $t \rightarrow 1$ ed infatti il solutore non riesce ad arrivare a calcolare la soluzione per $T = 1$. In un intervallo appena più piccolo la soluzione si trova senza difficoltà.

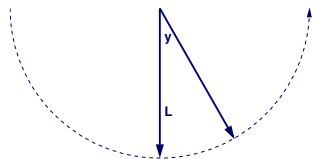
Esercizio 12

Risolvere usando le function di Matlab il problema di Cauchy:

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

Si osserva che la soluzione non esiste per ogni $t > 0$.
Trovare per tentativi l'intervallo massimale.

Moto di un pendolo con attrito



Un corpo di massa m è appeso all'estremità di un'asta rigida, di massa trascurabile e lunghezza L . L'asta è fissata all'altra estremità ad un perno e ruota in un piano verticale, soggetta alla forza gravitazionale (vedere figura a lato).

Supponiamo inoltre che le oscillazioni del pendolo siano smorzate da una forza d'attrito (per esempio la resistenza dell'aria) proporzionale alla velocità angolare $y'(t)$. Chiamato y l'angolo formato dall'asta con la verticale, l'equazione del moto è :

$$my''(t) = -mk \sin(y(t)) - hy'(t), \quad y(0) = \pi/4, \quad y'(0) = 0,$$

essendo $\alpha \in \mathbb{R}$, $k = g/L$ ed h il coefficiente di attrito.

Moto di un pendolo con attrito

Posto $k = 1$, $m = 1$, $L = 2$, risolvere l'equazione differenziale usando la function di Matlab **ode45** sia nel caso senza attrito $h = 0$ che con attrito $h = 0.2$.

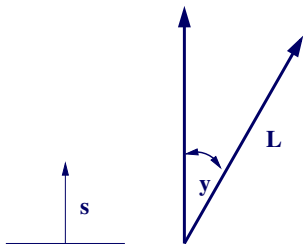
Riportare in una figura il grafico della soluzione $y(t)$ e in una seconda figura il piano delle fasi.

Per visualizzare il moto del pendolo si può usare il file di tipo script `plottapendolo1.m`.

Moto del pendolo inverso

Un famoso problema di meccanica non lineare è noto come problema del *pendolo inverso*. Il pendolo è un sbarra rigida di lunghezza $L = 10$ fissata ad un estremo ad un perno senza frizione. Il supporto si muove verticalmente azionato da un motore elettrico con la seguente legge di moto:

$$s = A \sin(\omega t).$$



Applicando la seconda legge di Newton si ottiene la seguente legge di moto

$$y''(t) = \frac{3}{2L}(g - A\omega^2 \sin(\omega t)) \sin y(t), \quad t \in [0, 2] \quad g = 386.09.$$

Un aspetto interessante di questo problema è che ci sono valori dei parametri per cui l'equazione è stabile per valori iniziali corrispondenti a una configurazione invertita.

Moto del pendolo inverso

Usare la function `ode45` per calcolare la soluzione dell'equazione differenziale data in corrispondenza dei seguenti valori dei parametri:

A	ω	$y(0)$	$y'(0)$
0	0	2.4	0
0.5	5.3	3.1	0
10	100	0.1	0
2	100	0.1	0
0.5	100	0.05	0

Fare il grafico della sola funzione $y(t)$ in corrispondenza di ciascun set di dati.

Per visualizzare il movimento del pendolo, si può usare la function `plottapendolo` (nella cartella scambio dati con il docente), mediante il comando `plottapendolo(t,u,A,omega)`.

Interpretare i risultati dal punto di vista fisico.

Esercizio 13 - molle e masse

Si consideri il sistema meccanico composto da due corpi di massa 1 fissati tra di loro e a due supporti fissi da tre molle con costante di elasticità k . Le incognite del sistema y_1 e y_2 rappresentano lo spostamento rispetto alla posizione di equilibrio. La dinamica del sistema è governata dalle seguenti equazioni differenziali:

$$\begin{aligned}y_1''(t) &= -ky_1(t) + k(y_2(t) - y_1(t)) & t \in [0, 10] \\y_2''(t) &= -k(y_2(t) - y_1(t)) - ky_2(t) & t \in [0, 10] \\y_1(0) &= 1 & y_1'(0) &= \sqrt{3k} \\y_2(0) &= 1 & y_2'(0) &= -\sqrt{3k}\end{aligned}$$

Trasformare il sistema in un sistema del primo ordine.

Posto $k = 1$ calcolare la soluzione numerica mediante la function `ode45` plottando solo le componenti che danno la soluzione $(y_1(t), y_2(t))$.

Esercizio 14 - Apollo e luna

Il seguente sistema di equazioni differenziali descrive il moto di un corpo in orbita attorno a due altri corpi molto più pesanti.

$$\begin{aligned}x'' &= 2y' + x - \frac{\mu^*(x + \mu)}{r_1^3} - \frac{\mu(x - \mu^*)}{r_2^3} \\y'' &= -2x' + y - \frac{\mu^*y}{r_1^3} - \frac{\mu y}{r_2^3} \\ \mu &= \frac{1}{82.45}, \quad \mu^* = 1 - \mu \\ r_1 &= ((x + \mu)^2 + y^2)^{1/2} \quad r_2 = ((x - \mu^*)^2 + y^2)^{1/2}\end{aligned} \tag{1}$$

Si studiano le soluzioni periodiche di questo problema.

È noto che le seguenti condizioni iniziali

$$\begin{aligned}x(0) &= 1.2 & x'(0) &= 0 \\ y(0) &= 0 & y'(0) &= -1.04935751\end{aligned}$$

danno luogo ad una soluzione periodica di periodo
 $T = 6.19216933$.

Apollo e luna (cont)

Il sistema di equazioni differenziali (1) potrebbe rappresentare, ad esempio, il moto della capsula Apollo in orbita intorno alla terra e alla luna. I tre corpi determinano un piano nello spazio e fissiamo su questo piano un sistema di coordinate come segue. L'asse x è la retta che congiunge i due corpi pesanti, l'origine viene posta nel loro baricentro e la loro distanza è presa come unità di misura. Quindi se μ è il rapporto fra la massa della luna e quella della terra, allora la luna e la terra si trovano nei punti di coordinate $(1 - \mu, 0)$ e $(-\mu, 0)$, rispettivamente e il sistema di coordinate si muove in accordo con la luna che ruota intorno alla terra. Si suppone che il terzo corpo, l'Apollo, abbia massa trascurabile rispetto ai primi due e che la sua posizione sia una funzione del tempo $(x(t), y(t))$. Le equazioni che governano il moto dell'Apollo possono essere dedotte dalla legge del moto di Newton e dalla legge di gravitazione. I termini con le derivate prime nell'equazione provengono dal moto del sistema di coordinate ruotante.

Apollo e luna (cont)

Esercizio

- Ricondurre il sistema di 2 equazioni differenziali del secondo ordine ad un sistema di 4 equazioni differenziali del primo ordine.
- Usare le function di Matlab **ode45** oppure **ode23s** per calcolare la soluzione.
- Riportare l'orbita dell'Apollo in un grafico.
- Osservato che con i valori di default l'orbita non è perfettamente periodica, modificare la tolleranza relativa per l'errore ponendo nelle opzioni `RelTol=1.e-5`.

Suggerimenti Usare le seguenti options per produrre il grafico desiderato:

`opt=odeset('OutputFcn','odephas2','OutputSel',[n m])`
essendo `n` e `m` gli indici delle incognite associate a $x(t)$ e a $y(t)$.