Il problema di Cauchy

Dati

- un intervallo $I = [t_0, t_0 + T]$,
- una funzione $f : I \times \mathbb{R} \to \mathbb{R}$,
- e un valore iniziale $y_0 \in \mathbb{R}$;

si cerca una funzione derivabile $y : I \to \mathbb{R}$ tale che

$$\begin{cases} y'(t) = f(t, y(t)) & \forall\, t \in I \\ y(t_0) = y_0 \,. \end{cases}$$

Il metodo di Eulero in avanti

Dato $N \in \mathbb{N}$

- si considera $h = T/N$
- e l'insieme di punti $\{t_n\}_{n=0}^{N} \subset I$, $t_n := t_0 + hn$.

Il metodo di Eulero in avanti calcola

$$\begin{cases} u_{n+1} = u_n + h\,f(t_n, u_n) & \text{per } n = 0, \ldots, N-1 \\ u_0 = y_0\,. \end{cases}$$

Ci aspetiamo che sia $u_n \approx y(t_n)$ e che

$$\max_{0 \le n \le N} |y(t_n) - u_n| \to 0 \text{ per } N \to \infty\,.$$

Esercizio

Scrivere una funzione di Matlab che implementi il metodo di Eulero in avanti.

```
function [t,u]=Eulero(fun,t0,T,y0,N)
h=T/N;
t=[t0:h:t0+T];
u(1)=y0;
for n=1:N
  u(n+1)=u(n)+h*feval(fun,t(n),u(n));
end
```

Il metodo di Heune

Dato $N \in \mathbb{N}$

- si considera $h = T/N$
- e l'insieme di punti in I, $t_n := t_0 + hn$, $n = 0, \ldots, N$.

Il metodo di Heune calcola

$$\begin{cases} u_{n+1} = u_n + \frac{h}{2}(K_1 + K_2) \text{ per } n = 0, \ldots, N-1 \\ \text{dove} \\ \quad K_1 = f(t_n, u_n) \\ \quad K_2 = f(t_n + h, u_n + hK_1) \\ u_0 = y_0 \,. \end{cases}$$

Ci aspetiamo che sia $u_n \approx y(t_n)$ e che

$$\max_{0 \le n \le N} |y(t_n) - u_n| = O(h^2)\,.$$

Esercizio

Scrivere una funzione di Matlab che implementi il metodo di Heun in avanti.

```
function [t,u]=Heun(fun,t0,T,y0,N)
h=T/N;
t=[t0:h:t0+T];
u(1)=y0;
for n=1:N
  K1=feval(fun,t(n),u(n));
  K2=feval(fun,t(n)+h,u(n)+h*K1);
  u(n+1)=u(n)+h/2*(K1+K2);
end
```

Un metodo a due passi

Dato $N \in \mathbb{N}$

- si considera $h = T/N$
- e l'insieme di punti in I, $t_n := t_0 + hn$, $n = 0, \ldots, N$.

Il metodo di Adams-Bashforth a due passi calcola

$$\begin{cases} u_{n+1} = u_n + \frac{h}{2}\left[3f(t_n, u_n) - f(t_{n-1}, u_{n-1})\right] \text{ per } n = 1, \ldots, N-1 \\ u_0 = y_0 \\ u_1 = ?\,. \end{cases}$$

Esercizio

- Scrivere una funzione di Matlab che implementi questo metodo usando il metodo di Eulero in avanti per calcolare u_1.

```
function [t,u]=AB(fun,t0,T,y0,N)
h=T/N;
t=[t0:h:t0+T];
u(1)=y0;
% Eulero in avanti
f(1)=feval(fun,t(1),u(1));
u(2)=u(1)+h*f(1);
% Adams Bashforth
for n=2:N
  f(n)=feval(fun,t(n),u(n));
  u(n+1)=u(n)+h/2*(3*f(n)-f(n-1));
end
```

Un metodo implicito. Il metodo di Crank-Nicolson

$$\begin{cases} u_{n+1} = u_n + \frac{h}{2}\left[f(t_n, u_n) + f(t_{n+1}, u_{n+1})\right] & \text{per } n = 0, \ldots, N-1 \\ u_0 = y_0 \,. \end{cases}$$

Esercizio

- Scrivere una funzione di Matlab che implementi il metodo di Crank-Nicolson per l'approssimazione della soluzione del problema di Cauchy

$$\begin{cases} y' = -ty & \forall\, t \in [0, 2] \\ y(0) = 1 \,. \end{cases}$$

```
function [t,u]=CN(N)
h=2/N;
t=[0:h:2];
u(1)=1;
for n=1:N
  u(n+1)=(2-h*t(n))/(2+h*t(n+1))*u(n);
end
```