CALCOLO NUMERICO con ELEMENTI DI PROGRAMMAZIONE (BATR) - (A.A. 2012-2013)

Prof. F. Pitolli

Esercitazione sui problemi di Cauchy

Ing. Gabriele Colosimo, Ing. Andrea Nascetti

Area di Geodesia e Geomatica
Dipartimento di Ingegneria Civile Edile e Ambientale
Università di Roma “La Sapienza“

<gabriele.colosimo, andrea.nascetti>@uniroma1.it

Indice

Indice

Problema di Cauchy: $\begin{cases} y'(t) = f(t, y(t)) & t \in [t_0, t_0 + \beta] \\ y(t_0) = y_0 & \text{condizione iniziale} \end{cases}$

Discretizzazione di I: $t_i = t_0 + ih \qquad i = 0, \ldots, n \qquad h = \dfrac{\beta}{n}$

Metodo numerico:

I valori **esatti** $y(t_i)$ vengono **approssimati** con i valori y_i.
Sviluppo in **serie di Taylor**:

$$\begin{aligned} y(t_1) &= y(t_0 + h) = y(t_0) + y'(t_0)h + \frac{1}{2}y''(\tau_1)h^2 = \\ &= y(t_0) + f(t_0, y(t_0))h + \frac{1}{2}y''(\tau_1)h^2 \end{aligned}$$

$$\tau_1 \in [t_0, t_1]$$

Soluzione **approssimata**:

$$y_{i+1} = y_i + hf(t_i, y_i)$$

Errore globale di troncamento:

$$e_i(t_i) = y(t_i) - y_i = \overline{P_i T_i}$$

Esercizio 6.1 (libro esercizi)

Problema di Cauchy: $\begin{cases} y'(t) = f\big(t, y(t)\big) = y - t & t \in I \\ y(0) = 2 & \text{condizione iniziale} \end{cases}$

La soluzione analitica è

$$g(t) : y = e^t + t + 1$$

Risolvere il problema di Cauchy con le seguenti condizioni:

1. $I = [0, 2]; \qquad t_i = t_0 + ih \qquad i = 0, \ldots, n \qquad h = 0.1$
2. $I = [0, 2]; \qquad t_i = t_0 + ih \qquad i = 0, \ldots, n \qquad h = 0.4$
3. $I = [0, 10]; \qquad t_i = t_0 + ih \qquad i = 0, \ldots, n \qquad h = 2$

Logica di implementazione

1. L'algoritmo è chiaro?
   - nel nostro caso si. . .

Logica di implementazione

1. L'algoritmo è chiaro?
   - nel nostro caso si. . .
2. Quali sono le mie variabili di input? (**dipende dal problema**)
   - numero di passi (n)
   - ampiezza (**costante**) del passo (h)
   - condizioni iniziali del problema $\big(y(0)\,;\,t(0)\big)$

Logica di implementazione

1. L'algoritmo è chiaro?
   - nel nostro caso si...
2. Quali sono le mie variabili di input? (**dipende dal problema**)
   - numero di passi (n)
   - ampiezza (**costante**) del passo (h)
   - condizioni iniziali del problema $\big(y(0)\,;\,t(0)\big)$
3. Cosa deve restituire il programma in output?
   - il valore di $t_i\,;\,y_i$ per ogni passo $i = 0, \ldots, N$
   - il valore dell'errore e_i per ogni passo $i = 0, \ldots, N$

Logica di implementazione

1. L'algoritmo è chiaro?
   - nel nostro caso si...
2. Quali sono le mie variabili di input? (**dipende dal problema**)
   - numero di passi (n)
   - ampiezza (**costante**) del passo (h)
   - condizioni iniziali del problema $\big(y(0)\,;t(0)\big)$
3. Cosa deve restituire il programma in output?
   - il valore di $t_i\,; y_i$ per ogni passo $i = 0, \ldots, N$
   - il valore dell'errore e_i per ogni passo $i = 0, \ldots, N$
4. Come strutturo il programma?
   - prevedo un ciclo di iterazioni?
   - quali costrutti uso?
   - devo scrivere delle funzioni particolari?
   - restituisco l'output sullo schermo o su un file?

Codice sorgente I

```
/* Programma di Eulero:
 *
 * Calcola la soluzione numerica di equazioni differenziali del
   primo ordine
 * con il metodo di Eulero esplicito
 *
 * Funzioni:
 * - f(t,y): termine noto dell equazione differenziale
 * - g(t): soluzione analitica del problema di Cauchy
 *
 * Input:
 * - x0, y0: condizione iniziale
 * - h: passo di discretizzazione
 * - n: numero di passi
 *
 * Output:
 * - xi: nodo i-esimo
 * - yi: approssimazione al nodo xi
 *
```

Codice sorgente II

```c
 */

#include <stdio.h>
#include <math.h>

double f(double t, double y);
double g(double t);

int main()
{
  // Allocazione e inizializzazione delle variabili
  int n = 0, k = 0;
  double t0 = 0., y0 = 0., h = 0.;
  double ti = 0., yi = 0., yi_1=0. /* la nostra y(i+1)*/;
  double err = 0.;

  // Recupero dei dati dati di input
  printf("Inserire il numero n di passi = \n");
  scanf("%d", &n);
```

Codice sorgente III

```c
printf("Inserire il valore di h = \n");
scanf("%lf", &h);
printf("Inserire il valore di t0 = \n");
scanf("%lf", &t0);
printf("Inserire il valore di y0 = \n");
scanf("%lf", &y0);

printf("n = %d \t h = %14.8lf \t t0 = %14.8lf \t y0 = %14.8lf\n\n", n, h, t0, y0);

printf(" Indice        t              y              errore\n");
// Implementazione dell'algoritmo di Eulero
ti = t0;
yi = y0;

for (k=1; k<=n; k++)
{
  yi_1 = yi + h * f(ti,yi);
  // Calcolo dell'errore
```

Codice sorgente IV

```
      err = g(ti + h) - yi_1;
      // Stampa dei risultati
      printf("%4d % 14.8lf % 14.8lf % 14.8lf\n", k, ti + h, yi_1,
          err);
      // Aggiornamento variabili
      yi = yi_1;
      ti = ti + h;
  }/* for */

  return 0;
}

// Implementazione della funzione f(t,y) del problema di Cauchy
 double f(double t, double y)
 {
  return y-t;
 }

// Implementazione della soluzione analitica g(t)
```

Codice sorgente V

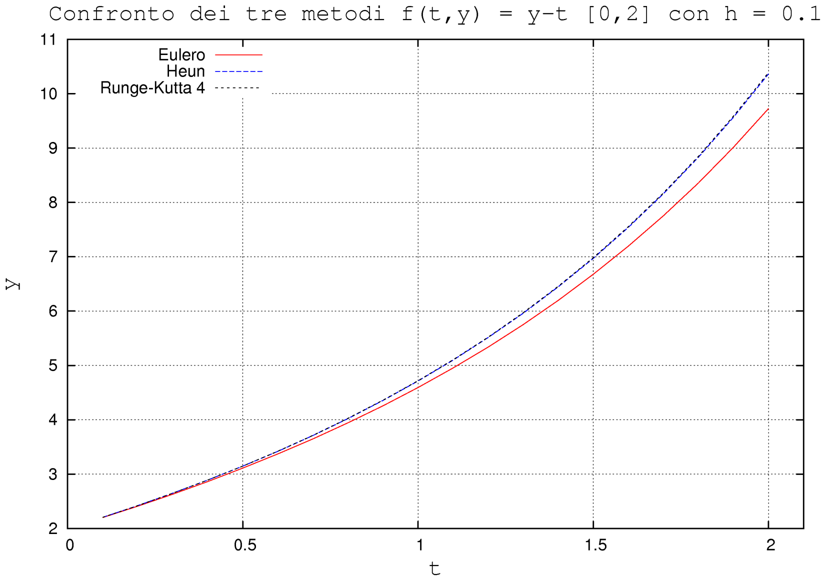
```
double g(double t)
{
 return exp(t)+t+1;
}
```

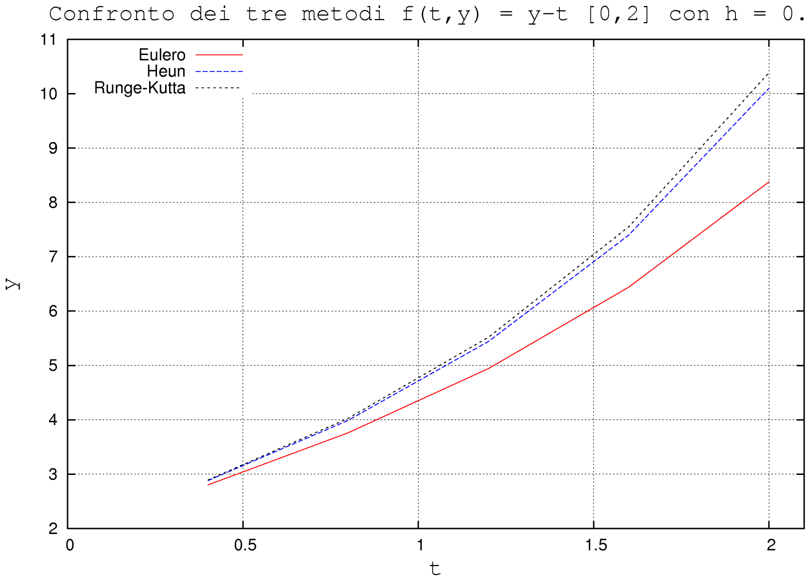
Indice

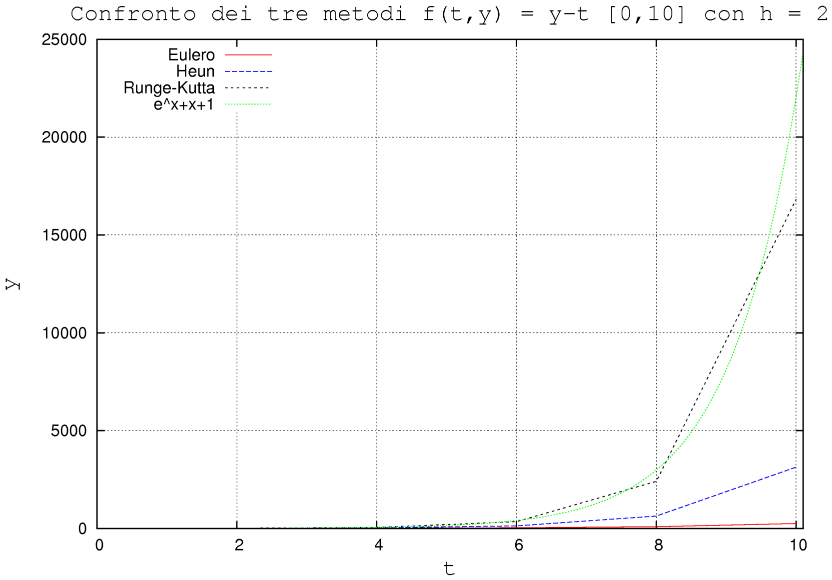
Esercizio

Dopo aver compreso il codice sorgente C del metodo di Eulero,

- Implementare il metodo di Heun
- Implementare il metodo di Runge-Kutta del 4° ordine
- Risolvere i problemi di Cauchy e confrontare (anche graficamente) l'andamento delle soluzioni e dell'errore dei tre metodi


Confronto dei tre metodi f(t,y) = y-t [0,2] con h = 0.1
Eulero
Heun
Runge-Kutta 4
y
t
11
10
9
8
7
6
5
4
3
2
0
0.5
1
1.5
2


Confronto dei tre metodi f(t,y) = y-t [0,2] con h = 0.4
Eulero
Heun
Runge-Kutta
y
t
11
10
9
8
7
6
5
4
3
2
0
0.5
1
1.5
2


Confronto dei tre metodi f(t,y) = y-t [0,10] con h = 2
Eulero
Heun
Runge-Kutta
e^x+x+1
y
t
0
5000
10000
15000
20000
25000
0
2
4
6
8
10

Esercizio 6.3 (libro esercizi)

Problema di Cauchy: $\begin{cases} y'(t) = f\big(t, y(t)\big) = -y + t & t \in I \\ y(0) = 1 & \text{condizione iniziale} \end{cases}$

La soluzione analitica è

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

Risolvere il problema di Cauchy con le seguenti condizioni:

1. $I = [0, 1]; \quad t_i = t_0 + ih \quad i = 0, \dots, n \quad h = 0.1$
2. $I = [0, 1]; \quad t_i = t_0 + ih \quad i = 0, \dots, n \quad h = 0.5$
3. verificare che la precisione del metodo di Heun con passo $h = 0.1$ sia raggiunta dal metodo di Eulero con passo $h = 0.01$