Compito 2: metodi embedded RK (*conficcati, incassati, innestati..?*) e primi paragoni con metodi RK espliciti

2020-2021 Approssimazione Numerica di ODEs

Primo Passo: Implementare in MATLAB la seguente funzione:

1. `function [y,nstep, nrech, neval, H_r,STIMA]=par_RK(f,t0,t_f,y0,tableau,TOL)`

corrispondenti ai metodi Embedded Runge-Kutta per approssimare il problema di valori iniziali:

$$\begin{cases} y' = f(t,y) \\ y(t_0) = y_0 \end{cases}$$

dove $y \in \mathbb{R}^d$ and $f : [0, t_f] \times \mathbb{R}^d \longrightarrow \mathbb{R}^d$ con $d \geq 1$ (il codice dovrebbe funzionare per problemi scalari e sistemi!!).

Questa familia di metodi sceglie ad ogni passo di tempo il *grid size* h_n per assicurare che il *errore locale stimato* sia sotto un valore pre-fissato di tolleranza TOL. Uno potrebbe anche considerare una combinazione di tolleranza assoluta `aTOL` e tolleranza relativa `rTOL`. Gli errori locali sono stimati durante il processo di integrazione in tempo con il Runge-Kutta `tableu`:

$$\begin{array}{c|c} c & A \\ \hline & b' \\ & B' \end{array}$$

Tipicamente, Il metodo $p(\hat{p})$ avanza la evoluzione temporale col tables di ordine p; mentre che usa il tableu di ordine $\hat{p}$ per stimare il errore.

La chiamata avviene come function handle; @nomefile_tableau. La dichiarazione all'interno della funzione par_RK.m si fa usando la funzione `feval`:

$$[c, A, b, B, p, P] = feval(tableau)$$

Se il metodo RK embedded RK ha la proprietá *FSAL (First Same As Last)*, la implementazione dovrebbe tenere conto di questa proprietá.

Sintassi. Variabili di Input e output

Variabili di Input

`fun` Variable di caratteri e una "function handle".
La funzione *[y_b]=fun*(t_a, y_a) valuta la funzione f (lato destro della ODE) nel instante di tempo t_a e valore y_a,(cioe valuta la funzione f nel punto (t_a, y_a) del piano di fasi). Osservare che per un sistema il valore in output deve essere sempre vettoriale. Per ogni problema, si deve costruire un $.m$ -file diverso che contiene la funzione specifica del problema.

t_0 tempo iniziale

t_f tempo finale

`TOL` Tolleranza che si chiede per l'stima dell'errore locale.

y_0 vettore colonna che contiene il dato iniziale y_0

`tableau` Variable di caratteri (e una "function handle") che contiene il nome del file `.m`-dove vengono definiti i coefficienti del tableau di Butcher:

$$\begin{array}{c|c} c & A \\ \hline & b' \\ & B' \end{array}$$

corrispondente al metodo RK embedded che si vuole usare nell'approssimazione.

Output variables

`y`] column vector containing the approximate solution at time t_f

`nevals` Costo del algoritmo in termini di numero di valutazioni di funzione.

`nstep` numero di passi accettati

`nrech` numero di passi rifiutati

`H_r` Matrice che contiene tutti i valori delle lunghezze di passo h rifiutati e tempo $t \in [t_0, t_f]$ dove il rifiuto accade.

`STIMA` (Opzionale): Matrice (o vettore) che contiene tutte le stime dell'errore locale fatte durante la evoluzione temporale ed il corrispondente tempo $t_n \in [t_0, t_f]$ dove la stima viene effettuata

Scelta-Modifica e Variazione del passo di tempo (*meshsize*)

Come abbiamo visto a lezione, la idea dei metodi embedded RK é scegliere e adattare la lunghezza del passo $h = dt$ tenendo conto la stima del errore locale che puo ottenersi combinando le due approssimazione numeriche fornite embedded RK. Si puo implementare la formula da Hairer-Norsett-Wanner book (pag. 168) che coinvolge tolleranze assolute e relative; oppure usare una unica tolleranza ed scegliere il passo d'accordo con la formula:

$$h_{new} = h_{old} \cdot \mathrm{Fcmax} \cdot \max\left(fac0, \min\left(Fcmin, \left(\frac{TOL}{Stima + \epsilon} \right)^{1/(q+1)} \right) \right) \tag{0.1}$$

dove $Fcmax, fac0$ and $Fcmin$ sono valori di salvaguardia, che devono essere scelti: $Fcmax$ tipicamente si prende da 0.75 a 0.95; $fac0 \in [0.3, 0.6]$ e $Fcmin \in [1, 2.5]$. Non ostante uno potrebbe trovare altra fascia di valori. Si puo anche considerare una versione semplificata usando soltanto $Fcmax$ con tutti gli altri posti come uno.

Stima si riferisce alla stima dell'errore che si ha nel passo attuale.

Esercizio 0 *(Linear Model test example).* Considerare il problema modello

$$\dot{y} = \lambda y, \qquad \lambda \in \mathbb{R} \qquad y(0) = 1 \ .$$

Approssimare numericamente la dinamica del problema per diversi valori di λ (considerando $\lambda < 0$ e $\lambda > 0$). Iniziare con $\lambda = -1$ e poi prendere $\lambda = -50$ e $\lambda = -200$.

a) Realizzare un diagramma (t, h) del tempo vs il paso di tempo per il calcolo della soluzione con diverse toleranze. Cosa osserva? Riesce a paragonare i risoltati con i RK espliciti e/o eulero implicito?

Se decidesse di usare $\lambda \in \mathbb{C}$ cossa osserva?

Esercizio 1 **Blow-up** Considerare il problema:

$$\dot{y} = y^2 \quad y(0) = y_0 > 0 \quad \longrightarrow \quad y(t) = \frac{y_0}{1 - t\, y_0} \quad \text{for } t < 1/y_0 \ .$$

Osservare che la soluzione di questo problema esiste soltanto per un tempo finito e dopo di che $\lim_{t \to 1/y_0} y(t) = \infty$. Usare eulero esplicito, eulero implicito, e almeno un RK esplicito e un embedded RK per la approssimazione.

Realizzare i seguenti grafici per diversi valori della toleranza TOL (oppure, *Rtol* e *Atol*)

a) Diagramma (t, y) tempo verso la approssimazione della soluzione.

b) Diagramma (t, h):tempo versus il paso di tempo $h = dt$ con che avanza il metodo. Si deve anche segnalare i valori di h che vengono scartati o rifiutati.

c) Diagramma (t, Est) dove Est é il vettore che contiene le stime degli errori locali che vengono calculati ad ogni paso di tempo.

d) Diagramma (t, ERR) dove ERR é il vettore che contiene il vero errore commesso ad ogni paso di tempo. Comparare questo grafico con quello precedente. Saprebbe spiegare quello che osserva?

Esercizio 2 Considerare il problema

$$\dot{y} = \cos^2(\alpha y) \quad \longrightarrow \quad y(t) = \frac{\arctan(\alpha(t-c))}{\alpha} \quad y(0) \in \left(-\frac{\pi}{2\alpha}, \frac{\pi}{2\alpha}\right)$$

per $\alpha = 40$ e $y(0) = -0.0386 \approx \frac{\pi}{2\alpha}$. Usare eulero esplicito, un RK explicito e uno (o vari) embedded RK per la approssimazione nel intervallo di tempo $[0, 2]$. In questo esercizio si vuole studiare in modo particolare gli metodi adattativi in tempo (embedded RK). Realizzare i seguenti grafici per diversi valori della toleranza TOL (oppure, *Rtol* e *Atol*)

a) Diagramma (t, y) tempo verso la approssimazione della soluzione.

b) Diagramma (t, h):tempo versus il paso di tempo $h = dt$ con che avanza il metodo. Si deve anche segnalare i valori di h che vengono scartati o rifiutati.

c) Diagramma (t, Est) dove Est é il vettore che contiene le stime degli errori locali che vengono calcolati ad ogni paso di tempo.

d) Diagramma (t, ERR) dove ERR é il vettore che contiene il vero errore commesso ad ogni paso di tempo. Comparare questo grafico con quello precedente. Riuscirebbe ad spiegare quello che osserva? (deve pensare a come é la soluzione esatta del problema).

Esercizio 3 Considerare il problema del pendulo (esercizio 6 del primo compitino) E fare lo studio della approssimazione con un paio di metodi adattativi in tempo (embedded RK), corrispondenti a ordine alto e medio-basso. Che osserva? Come sono le traiettorie? Fare uno studio comparativo con gli altri metodi implementati. Fare uno studio di convergenza.

Esercizio 4. (Brusselator) Considerare il sistema autonomo di "Brusselator" nell'intervallo di tempo $[t_0, t_f] = [0, 20]$. Il sistema modella alcune reazioni chimiche multi-cellulare (see Hairer-Nörsett-Wanner book).

$$\left\{ \begin{aligned} y_1' &= 1 + y_1^2\, y_2 - 4\, y_1 \\ y_2' &= 3\, y_1 - y_1^2\, y_2 \end{aligned} \right. \tag{0.2}$$

con dato iniziale

$$y(t_0) = \begin{pmatrix} 3/2 \\ 3 \end{pmatrix} \qquad \text{en} \quad t_0 = 0\,. \tag{0.3}$$

- Approssimare la soluzione del sistema e rappresentare graficamente le traiettorie (entrambi componenti separatamente), per diversi valori di $TOL = 10^{-4}$, 10^{-6}. Studiare cosa succede per diverse dati iniziali $y(t_0)$ e rappresentare le traiettorie.
- Per l'analisi di convergenza fare i seguenti diagrammi:
  a) Diagramma (t, y_1), (t, y_2): time- versus ogni componente della soluzione.
  b) Diagramma (t, h): time-versus paso di tempo $h = dt$ accettato e valori dei passi di tempo rifiutati.
  c) Diagramma (t, Est) dove Est é la matrice che contiene le estime fatte dal metodo dell errore locale ad ogni passo di tempo.
- I grafici possono farsi per diversi valori di $TOL = 10^{-3}, 10^{-6}$, $10^{-8}\, 10^{-10}$. (2 o 3 valori!)

In generale, fare uno studio numerico della convergenza dei metodi embedded RK, verificando la teoria vista a lezione. In particolare si possono fare:

1. *Convergence Diagrams:* Diagrammi che mostrano paso di tempo h versus error. Verificare cosi il ordine di convergenza dei metodi. Per fare i diagrammi, si deve scegliere tra un h_{max} o un h_{medio} (averaged).
2. *Efficiency Diagrams.* Diagramma che mostra costo (nevals) versus error
3. Fare studio-Comparativo con i metodi RK espliciti (e altri metodi)