



1506
UNIVERSITÀ
DEGLI STUDI
DI URBINO
CARLO BO

Relazione relativa al progetto d'esame di Programmazione Logica e Funzionale

sessione estiva — a.a. 2023/2024

Corso di Laurea in Informatica Applicata
Università di Urbino

STUDENTI:

Barzotti Nicolas
matricola: 313687
Ramagnano Gabriele
matricola: 315439

DOCENTE:

Marco Bernardo

Indice

1	Specifica del Problema	2
2	Analisi del Problema	3
2.1	Dati di Ingresso del Problema	3
2.2	Dati di Uscita del Problema	3
2.3	Relazioni Intercorrenti tra i Dati del Problema	3
3	Progettazione dell'Algoritmo	7
3.1	Scelte di Progetto	7
3.1.1	Haskell	7
3.1.2	Prolog	7
3.2	Passi dell'Algoritmo	7
4	Testing	8
4.1	Haskell	8
4.2	Prolog	8

1 Specifica del Problema

Scrivere un programma Haskell e un programma Prolog che acquisiscono da tastiera un insieme finito di parametri numerici e poi stampano su schermo il risultato del calcolo numerico per l'integrazione di equazioni differenziali di moto fugoide senza attrito, di moto fugoide con attrito, di convezione lineare a una dimensione ed di Burgers a una dimensione.

2 Analisi del Problema

2.1 Dati di Ingresso del Problema

I dati in ingresso al problema sono stati suddivisi in base all'equazione da integrare numericamente, ne segue quindi la loro descrizione.

Fugoide Senza Attrito

Il dato in ingresso è un numero reale maggiore di zero. Questo rappresenta il passo temporale per l'integrazione dell'equazione del moto fugoide senza attrito.

Fugoide Con Attrito

Il dato in ingresso è un numero reale maggiore di zero. Questo rappresenta il passo temporale per l'integrazione dell'equazione del moto fugoide con attrito.

Convezione

I dati in ingresso per l'integrazione dell'equazione di convezione sono:

1. un numero intero naturale, rappresenta il numero di punti della funzione d'onda;
2. un numero reale maggiore di zero, rappresenta la lunghezza del passo temporale.

Burgers

Il dato in ingresso per l'integrazione dell'equazione di Burgers è un numero intero naturale. Questo rappresenta il numero di punti della funzione d'onda.

2.2 Dati di Uscita del Problema

Fugoide Senza Attrito

Il dato in uscita dell'integrazione dell'equazione del moto fugoide senza attrito è una sequenza di numeri reali che rappresentano la funzione di traiettoria dell'areomobile.

Fugoide Con Attrito

Il dato in uscita dell'integrazione dell'equazione del moto fugoide con attrito è una sequenza di numeri reali che rappresentano la funzione di traiettoria dell'areomobile.

Convezione

Il dato in uscita dell'integrazione dell'equazione di convezione lineare a una dimensione è una sequenza di numeri reali che rappresentano i valori finali della funzione d'onda quadra.

Burgers

Il dato in uscita all'integrazione dell'equazione di Burgers a una dimensione è una sequenza di numeri reali che rappresentano i valori finali della la funzione a dente di sega.

2.3 Relazioni Intercorrenti tra i Dati del Problema

Fugoide Senza Attrito

L'equazione per il moto di fugoide senza attrito è un'equazione differenziale ordinaria del secondo ordine:

$$z(t)'' = g - \frac{g z(t)}{z_t} = g \left(1 - \frac{z(t)}{z_t} \right). \quad (1)$$

Possiamo trasformare questa equazione del secondo ordine in un sistema di equazioni del primo ordine:

$$z'(t) = b(t) \quad (2)$$

$$b'(t) = g \left(1 - \frac{z(t)}{z_t} \right). \quad (3)$$

Un altro modo di considerare un sistema di due equazioni ordinarie del primo ordine è scrivere il sistema differenziale come un'unica equazione vettoriale:

$$\vec{u} = \begin{pmatrix} z \\ b \end{pmatrix} \quad (4)$$

$$\vec{u}'(t) = \begin{pmatrix} b \\ g - g \frac{z(t)}{z_t} \end{pmatrix}. \quad (5)$$

La soluzione approssimativa al tempo t_n è u_n e la soluzione numerica dell'equazione differenziale consiste nel calcolare una sequenza di soluzioni con la seguente equazione:

$$u_{n+1} = u_n + \Delta t f(u_n) + O(\Delta t)^2. \quad (6)$$

Questa formula è chiamata metodo di Eulero. Per le equazioni di moto fugoide, il metodo di Eulero fornisce il seguente algoritmo:

$$z_{n+1} = z_n + \Delta t b_n \quad (7)$$

$$b_{n+1} = b_n + \Delta t \left(g - \frac{g}{z_t} z_n \right). \quad (8)$$

dove:

- Δt è la lunghezza del passo temporale;
- g è la forza gravitazionale terrestre;
- z_n è l'altitudine del velivolo al passo n ;
- z_t è l'altitudine centrale del velivolo;
- b_n è la velocità del velivolo al passo n .

Il numero di passi di simulazione n viene calcolato $n = \frac{T}{\Delta t}$, dove T è il tempo totale di simulazione.

Fugoide Con Attrito

L'equazione per il moto fugoide con attrito è un sistema di equazioni differenziali ordinarie del primo ordine:

$$m \frac{dv}{dt} = -W \sin \theta - D \quad (9)$$

$$mv \frac{d\theta}{dt} = -W \cos \theta + L. \quad (10)$$

Per visualizzare le traiettorie di volo previste da questo modello, che dipendono sia dalla velocità di avanzamento v sia dall'angolo della traiettoria θ , occorre calcolare la posizione dell'aliante ad ogni istante di tempo t . La posizione dell'aliante su un piano verticale sarà determinata dalle coordinate (x, y) :

$$x'(t) = v \cos(\theta) \quad (11)$$

$$y'(t) = v \sin(\theta). \quad (12)$$

L'intero sistema di equazioni discretizzate con il metodo di Eulero è:

$$v^{n+1} = v^n + \Delta t \left(-g \sin \theta^n - \frac{C_D}{C_L} \frac{g}{v_t^2} (v^n)^2 \right) \quad (13)$$

$$\theta^{n+1} = \theta^n + \Delta t \left(-\frac{g}{v^n} \cos \theta^n + \frac{g}{v_t^2} v^n \right) \quad (14)$$

$$x^{n+1} = x^n + \Delta t v^n \cos \theta^n \quad (15)$$

$$y^{n+1} = y^n + \Delta t v^n \sin \theta^n. \quad (16)$$

Scritto in forma vettoriale risulta:

$$u'(t) = f(u)$$

$$u = \begin{pmatrix} v \\ \theta \\ x \\ y \end{pmatrix} \quad f(u) = \begin{pmatrix} -g \sin \theta - \frac{C_D}{C_L} \frac{g}{v_t^2} v^2 \\ -\frac{g}{v} \cos \theta + \frac{g}{v_t^2} v \\ v \cos \theta \\ v \sin \theta \end{pmatrix}. \quad (17)$$

dove:

- Δt è la lunghezza del passo temporale;
- g è la forza gravitazionale terrestre;
- x è lo spostamento orizzontale del velivolo;
- y è l'altitudine del velivolo;
- v è la velocità del velivolo;
- v_t è la velocità di trim;
- θ è l'angolo d'inclinazione del velivolo;
- C_D è il coefficiente di resistenza dell'aria;
- C_L è il coefficiente di portanza.

Il numero di passi di simulazione n viene calcolato $n = \frac{T}{\Delta t}$, dove T è il tempo totale di simulazione. Le condizioni iniziali sono rappresentate dalle costanti di integrazione definite dal valore della derivata al tempo $t = 0$:

$$\begin{aligned} v(0) &= v_0 & \text{and} & & \theta(0) &= \theta_0 \\ x(0) &= x_0 & \text{and} & & y(0) &= y_0. \end{aligned}$$

Convezione

L'equazione di convezione lineare unidimensionale è un'equazione differenziale alle derivate parziali:

$$\frac{\partial u}{\partial t} + c \frac{\partial u}{\partial x} = 0. \quad (18)$$

Per la soluzione numerica di $u(x, t)$ si sono utilizzati i pedici per denotare la posizione spaziale, come in u_i , e gli apici per denotare l'istante temporale, come in u^n :

$$\begin{array}{ccc} \bullet & \bullet & \bullet \\ u_{i-1}^{n+1} & u_i^{n+1} & u_{i+1}^{n+1} \\ \bullet & \bullet & \bullet \\ u_{i-1}^n & u_i^n & u_{i+1}^n \\ \bullet & \bullet & \bullet \\ u_{i-1}^{n-1} & u_i^{n-1} & u_{i+1}^{n-1}. \end{array}$$

L'equazione per fornire la soluzione numerica del problema è data da:

$$u_i^{n+1} = u_i^n - c \frac{\Delta t}{\Delta x} (u_i^n - u_{i-1}^n). \quad (19)$$

dove:

- Δt è la lunghezza del passo temporale;
- Δx è la lunghezza del passo spaziale;
- c è la velocità dell'onda;

Le condizioni iniziali per una funzione d'onda quadra sono definite così:

$$u(x, 0) = \begin{cases} 2 & \text{dove } 0.5 \leq x \leq 1, \\ 1 & \text{altrimenti} \end{cases} \quad (20)$$

dove il dominio della soluzione numerica è definito in $x \in (0, 2)$. In questo modo la lunghezza del passo spaziale viene calcolata come $\Delta x = \frac{d}{n-1}$, dove d è la distanza tra il limite inferiore e il limite superiore del dominio della soluzione numerica, n il numero di punti della funzione d'onda discretizzata. Il numero totale di passi temporali i è invece costante.

Poniamo inoltre delle condizioni al contorno su x : sia $u = 1$ quando $x = 0$.

Burgers

L'equazione di Burgers unidimensionale è un'equazione differenziale alle derivate parziali:

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = \nu \frac{\partial^2 u}{\partial x^2}. \quad (21)$$

L'equazione per fornire la soluzione numerica del problema è data da:

$$u_i^{n+1} = u_i^n - u_i^n \frac{\Delta t}{\Delta x} (u_i^n - u_{i-1}^n) + \nu \frac{\Delta t}{\Delta x^2} (u_{i+1}^n - 2u_i^n + u_{i-1}^n) \quad (22)$$

dove u :

$$2\nu \left(-\frac{(-8t+2x)e^{-\frac{(-4t+x)^2}{4\nu(t+1)}}}{4\nu(t+1)} - \frac{(-8t+2x-4\pi)e^{-\frac{(-4t+x-2\pi)^2}{4\nu(t+1)}}}{4\nu(t+1)} \right) - \frac{\quad}{e^{-\frac{(-4t+x-2\pi)^2}{4\nu(t+1)}} + e^{-\frac{(-4t+x)^2}{4\nu(t+1)}}} + 4 \quad (23)$$

e le condizioni iniziali sono definite con $u(x, 0)$. Le condizioni al contorno sono:

$$u(0) = u(2\pi). \quad (24)$$

3 Progettazione dell'Algoritmo

3.1 Scelte di Progetto

3.1.1 Haskell

3.1.2 Prolog

3.2 Passi dell'Algoritmo

4 Testing

4.1 Haskell

4.2 Prolog