

Relazione di Tirocinio

Leonardo Cruciani

July 7, 2019

1 Introduzione teorica

2 Interpolazione trilineare

Per il caso di interpolazione lineare si vede facilmente che un punto $(x, f(x))$ che giace sulla retta passante per $(x_0, f(x_0))$ e $(x_1, f(x_1))$ soddisfa la relazione:

$$\frac{f(x) - f(x_0)}{x - x_0} = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$$

Da cui si ottiene:

$$f(x) = \frac{1}{x_1 - x_0} ((x - x_0)f(x_1) + (x_1 - x)f(x_0))$$

La generalizzazione al caso bi- e tri- lineare consiste nella reiterazione del procedimento per ciascuna variabile. Introducendo l'interpolante lineare lungo \hat{x} :

$$g(x, y_i) = \frac{1}{x_1 - x_0} ((x - x_0)f(x_1, y_i) + (x_1 - x)f(x_0, y_i))$$

Si ottiene (per semplicità di notazione $f_{ij} = f(x_i, y_j)$ dove $i, j = 0, 1$):

$$\begin{aligned} f(x, y) &= \frac{1}{y_1 - y_0} ((y - y_0)g(x, y_1) + (y_1 - y)g(x, y_0)) = \\ &= \frac{1}{(x_1 - x_0)(y_1 - y_0)} [(y - y_0)(x - x_0)f_{11} + (y - y_0)(x_1 - x)f_{01} + \\ &\quad + (y_1 - y)(x - x_0)f_{10} + (y_1 - y)(x_1 - x)f_{00}] = \\ &= \frac{1}{\Delta\sigma} \begin{bmatrix} x - x_0 & x - x_1 \end{bmatrix} \cdot \begin{bmatrix} f_{00} & f_{01} \\ f_{10} & f_{11} \end{bmatrix} \cdot \begin{bmatrix} y - y_0 \\ y_1 - y \end{bmatrix} = \frac{1}{\Delta\sigma} \bar{x}^\tau \cdot \bar{\bar{F}} \cdot \bar{y} \end{aligned}$$

Con una conveniente notazione vettoriale.

Si procede come prima con:

$$h(z_i) = \frac{1}{(x_1 - x_0)(y_1 - y_0)} \bar{x}^\tau \cdot \bar{\bar{F}}(z_i) \cdot \bar{y}$$

In tal modo:

$$\begin{aligned} f(x, y, z) &= \frac{1}{z_1 - z_0} ((z - z_0)h(x, y, z_1) + (z_1 - z)h(x, y, z_0)) = \\ &= \frac{1}{(x_1 - x_0)(y_1 - y_0)(z_1 - z_0)} [(z - z_0) \bar{x}^\tau \cdot \bar{\bar{F}}(z_1) \cdot \bar{y} + (z - z_0) \bar{x}^\tau \cdot \bar{\bar{F}}(z_0) \cdot \bar{y}] = \\ &= \frac{1}{\Delta\tau} \bar{x}^\tau \cdot [(z - z_0)\bar{\bar{F}}(z_1) + (z_1 - z)\bar{\bar{F}}(z_0)] \cdot \bar{y} = \frac{1}{\Delta\tau} \bar{x}^\tau \cdot [\bar{z} \otimes \bar{\bar{F}}] \cdot \bar{y} \end{aligned}$$

Notazione scalare: svolgendo i prodotti si ottiene la seguente interpolazione di f , che verrà manipolata al fine di facilitarne l'implementazione in un codice :

$$f(x, y, z) = \sum_{i,j,k}^{0,1} \tilde{x}_i \tilde{y}_j \tilde{z}_k f(x_i, y_j, z_k) \quad \begin{cases} \tilde{x}_0 = x_1 - x \\ \tilde{x}_1 = x - x_0 \end{cases}$$

Il vettore a due componenti \tilde{x} può essere rappresentato più concisamente con:

$$\tilde{x}[i] = (-)^{i-1} (x - x_{|i-1|})$$

Si può quindi applicare la seguente notazione anche alle componenti y e z ed introdurre tre liste $x_0[l]$, $y_0[m]$ e $z_0[n]$ contenenti gli estremi degli intervalli in cui sono stati divisi gli assi.

Per identificare ciascuno dei parallelepipedi che compongono il dominio 3D di f saranno necessari tutti e tre gli indici l, m, n .

L'implementazione è quindi immediata, una volta stabilito che parallelepipedo $P(l, m, n)$ appartiene il punto (x, y, z) nel quale si vuole stimare il valore di f , il programma procede a calcolare:

$$R = \sum_{i,j,k}^{0,1} (-)^{i+j+k+1} (x - x_0[l + |1 - i|]) (y - y_0[m + |1 - j|]) (z - z_0[n + |1 - k|]) T$$

Dove i valori tabulabili T sono:

$$T = f(x_0[l + i], y_0[m + j], z_0[n + k])$$