

1 Introduzione: matematica

1.1 Derivate numeriche

Per prima cosa inizio con un piccolo elenco di derivate numeriche, usando il metodo delle differenze numeriche:

La derivata prima (in avanti) è:

$$\frac{\partial F}{\partial x}(a) \simeq \frac{F(a+h) - F(a)}{h} + O(h) \quad (1.1.1)$$

Ma la sua precisione è al primo ordine, per cui utilizzerò la versione cosiddetta "centrale":

$$\frac{\partial F}{\partial x}(a) \simeq \frac{F(a+h) - F(a-h)}{2 * h} + O(h^2) \quad (1.1.2)$$

Per la derivata seconda il discorso è simile ma in entrambi i casi la precisione è sempre al secondo ordine:

$$\frac{\partial^2 F}{\partial x^2}(a) \simeq \frac{\frac{\partial F}{\partial x}(a+h) - \frac{\partial F}{\partial x}(a)}{h} = \frac{F(a+2h) + F(a) - 2F(a+h)}{h^2} + O(h^2) \quad (1.1.3)$$

$$\frac{\partial^2 F}{\partial x^2}(a) \simeq \frac{F(a+h) + F(a-h) - 2F(a)}{h^2} + O(h^2) \quad (1.1.4)$$

1.2 Equazione del calore

Per prima cosa parto dall'equazione del calore

$$\frac{\partial T}{\partial t} = K \frac{\partial^2 T}{\partial x^2} \quad (1.2.1) \quad \{\text{eq:cal}\}$$

Per calcolare l'equazione come prima cosa calcoliamo le derivate:

$$\frac{\partial T}{\partial t}(x, t) = \frac{T(x, t + \Delta t) - T(x, t)}{\Delta t} \quad (1.2.2)$$

e

$$\frac{\partial^2 T}{\partial x^2}(x, t) = \frac{T(x + \Delta x, t) + T(x - \Delta x, t) - 2T(x, t)}{\Delta x^2} \quad (1.2.3)$$

Per prima cosa eguaglio le derivate (ma devo ricordare che la risoluzione che si avrebbe precisione Δx^2 nello spazio e Δt nel tempo):

$$T(x, t + \Delta t) = k \frac{\Delta t}{\Delta x^2} (T(x + \Delta x, t) + T(x - \Delta x, t) - 2T(x, t)) + T(x, t) \quad (1.2.4)$$

e andando avanti:

$$\frac{T(x, t + \Delta t) - T(x, t)}{\Delta t} = k \frac{T(x + \Delta x, t) + T(x - \Delta x, t) - 2T(x, t)}{\Delta x^2} \quad (1.2.5)$$

A questo punto calcolo ogni punto conoscendo i tre del passo precedente, ma non approfondirò la cosa.

Oppure posso mettere uguagliare con la derivata spaziale all'istante successivo:

$$\frac{T(x, t + \Delta t) - T(x, t)}{\Delta t} = k \frac{T(x + \Delta x, t + \Delta t) + T(x - \Delta x, t + \Delta t) - 2T(x, t + \Delta t)}{\Delta x^2} \quad (1.2.6)$$

proseguo:

$$T(x, t + \Delta t) - k \frac{\Delta t}{\Delta x^2} (T(x + \Delta x, t + \Delta t) + T(x - \Delta x, t + \Delta t) - 2T(x, t + \Delta t)) = T(x, t) \quad (1.2.7)$$

Anche in questo caso non approfondisco, preferisco ricavare il metodo di Crank-Nicolson per poi generalizzarlo e descriverne l'algoritmo utilizzato per la risoluzione dell'equazione:

1.3 Un'esempio di come ricavare il metodo di Crank Nicolson: l'equazione del calore

Innanzitutto utilizziamo la definizione centrale della derivata prima in modo da avere una precisione di Δt^2 anche per quanto riguarda il tempo, ma la calcolo in $t + \Delta t/2$ con incremento Δt :

$$\frac{\partial T}{\partial t}(x, t + \Delta t/2) \simeq \frac{T(x, t + \Delta t/2 + \Delta t/2) - T(x, t - \Delta t/2 + \Delta t/2)}{2 * \Delta t/2} = \frac{T(x, t + \Delta t) - T(x, t)}{\Delta t} \quad (1.3.1)$$

Per lo spazio utilizziamo le derivate calcolate negli esempi precedenti. A questo punto abbiamo $\frac{\partial T}{\partial t}(x, t + \Delta t/2)$, $\frac{\partial^2 T}{\partial x^2}(x, t)$ e $\frac{\partial^2 T}{\partial x^2}(x, t + \Delta t)$ per rispettare l'equazione facciamo la media delle due derivate spaziali ai tempi t e $t + \Delta t$.

$$\frac{T(x, t + \Delta t) - T(x, t)}{\Delta t} = \frac{K}{2} \left(\frac{T(x + \Delta x, t) + T(x - \Delta x, t) - 2T(x, t)}{\Delta x^2} + \frac{T(x + \Delta x, t + \Delta t) + T(x - \Delta x, t + \Delta t) - 2T(x, t + \Delta t)}{\Delta x^2} \right) \quad (1.3.2) \quad \{\text{eq:Heat}\}$$

In pochi passaggi si arriva a separare le parti a tempo differente:, con $\eta = K \frac{\Delta t}{\Delta x^2}$:

$$\left(\frac{2}{\eta} + 2 \right) T(x, t + \Delta t) - T(x + \Delta x, t + \Delta t) - T(x - \Delta x, t + \Delta t) = \left(\frac{2}{\eta} - 2 \right) T(x, t) + T(x + \Delta x, t) + T(x - \Delta x, t) \quad (1.3.3)$$

Per ogni istante di tempo ho una matrice tridiagonale per il passo temporale che conosco e per quello successivo. In subsection 2.2 descriverò come si risolve una di queste matrici.

2 Costruzione dell'algoritmo

2.1 Ottenere il sistema

Ho fatto un esempio col l'equazione del calore. Prima di procedere alla spiegazione su come si semplifica e si risolve un sistema di equazioni riconducibile ad una matrice tridiagonale spiegherò come condurre la PDE più generale ad un sistema del genere.

Partendo da una generica equazione di secondo ordine:

$$\partial_t F = D_2 \partial_x^2 F + D_1 \partial_x F + V(x, t) F + U(x, t) \quad (2.1.1) \quad \{\text{eq:gen}\}$$

Il coefficiente della derivata temporale è ignorato perché è incluso negli altri coefficienti, ovviamente D_2 , il coefficiente della derivata seconda spaziale non deve essere mai uguale a zero! Inoltre preferisco lasciare D_1 e D_2 come costanti nel tempo e nello spazio, e lascio la dipendenza temporale e spaziale a V e U , che possono essere costanti a loro volta.

In seguito indicherò la il passo nello spazio a pedice con i e quello nel tempo in apice con j . Discretizzando l'equazione ottengo quindi (ho mantenuto la definizione di derivata centrale anche per quella prima spaziale in modo da mantenere la precisione):

$$\begin{aligned} \frac{F_i^{j+1} - F_i^j}{\Delta t} = \frac{1}{2} \left(D_2 \frac{F_{i+1}^j + F_{i-1}^j - 2F_i^j}{\Delta x^2} + D_1 \frac{F_{i+1}^j - F_{i-1}^j}{2\Delta x} + F_i^j V_i^j + U_i^j + \right. \\ \left. D_2 \frac{F_{i+1}^{j+1} + F_{i-1}^{j+1} - 2F_i^{j+1}}{\Delta x^2} + D_1 \frac{F_{i+1}^{j+1} - F_{i-1}^{j+1}}{2\Delta x} + F_i^{j+1} V_i^{j+1} + U_i^{j+1} \right) \end{aligned} \quad (2.1.2)$$

Salto i passaggi e indico con $\eta = \frac{D_2 \Delta t}{\Delta x^2}$ posso scrivere:

$$\begin{aligned} \left(-\eta + D_1 \frac{\Delta t}{2\Delta x} \right) F_{i-1}^{j+1} + \left(2 + 2\eta - \Delta t V_i^{j+1} \right) F_i^{j+1} + \left(-\eta - D_1 \frac{\Delta t}{2\Delta x} \right) F_{i+1}^{j+1} - \Delta t U_i^{j+1} = \\ \left(\eta - D_1 \frac{\Delta t}{2\Delta x} \right) F_{i-1}^j + \left(2 - 2\eta + \Delta t V_i^j \right) F_i^j + \left(\eta + D_1 \frac{\Delta t}{2\Delta x} \right) F_{i+1}^j + \Delta t U_i^j \end{aligned} \quad (2.1.3)$$

Per rendere più chiara spiegazione e risoluzione della matrice tridiagonale riassumo l'equazione precedente in:

$$\begin{aligned} a_i^j &= -1 + \frac{D_1}{D_2} \frac{\Delta x}{2} & ak_i^j &= 1 - \frac{D_1}{D_2} \frac{\Delta x}{2} & \text{parametri dei } (F_{i-1}^*) \\ d_i^j &= \frac{1}{\eta} \left(2 - \Delta t V_i^{j+1} \right) + 2 & dk_i^j &= \frac{1}{\eta} \left(2 + \Delta t V_i^j \right) - 2 & \text{parametri dei } (F_i^*) \\ c_i^j &= -1 - \frac{D_1}{D_2} \frac{\Delta x}{2} & ck_i^j &= 1 + \frac{D_1}{D_2} \frac{\Delta x}{2} & \text{parametri dei } (F_{i+1}^*) \\ e_i^j &= -\frac{\Delta x^2}{D_2} U_i^{j+1} & ek_i^j &= \frac{\Delta x^2}{D_2} U_i^j & \text{funzioni esterne} \end{aligned} \quad (2.1.4) \quad \{\text{eq:par}\}$$

2.2 La matrice Tridiagonale: soluzione

Per ogni istante di tempo j ho un sistema di N equazioni nella forma:

$$a_i F_{i-1}^{j+1} + d_i F_i^{j+1} + c_i F_{i+1}^{j+1} + e_i = ak_i F_{i-1}^j + dk_i F_i^j + ck_i F_{i+1}^j + ek_i \quad (2.2.1)$$

Dove j rappresenta l'istante di tempo che conosco e $j+1$ quello che sto calcolando. La prima cosa da fare è portare nel membro a destra tutte le cose che conosco, dando per scontato che l'unica incognita dell'equazione è la funzione:

$$a_i F_{i-1}^{j+1} + d_i F_i^{j+1} + c_i F_{i+1}^{j+1} = ak_i F_{i-1}^j + dk_i F_i^j + ck_i F_{i+1}^j + ek_i - e_i \quad (2.2.2)$$

Per proseguire con la risoluzione è meglio passare alla rappresentazione matriciale del sistema (rappresento gli N punti rispettando le convenzioni del C, quindi $i = 0 \rightarrow N-1$):

$$\begin{pmatrix} d_0 & c_0 & & & \\ a_1 & d_1 & c_1 & & \\ & & \dots & & \\ & & & a_{N-2} & d_{N-2} & c_{N-2} \\ & & & a_{N-1} & d_{N-1} & \end{pmatrix} F^{j+1} = \begin{pmatrix} dk_0 & ck_0 & & & \\ ak_1 & dk_1 & ck_1 & & \\ & & \dots & & \\ & & & ak_{N-2} & dk_{N-2} & ck_{N-2} \\ & & & ak_{N-1} & dk_{N-1} & \end{pmatrix} F^j + \begin{pmatrix} ek_0 - e_0 \\ ek_1 - e_1 \\ \dots \\ ek_{N-2} - e_{N-2} \\ ek_{N-1} - e_{N-1} \end{pmatrix} \quad (2.2.3)$$

Per comodità compatto il lato conosciuto in un vettore B^j le cui componenti sono:

$$b_i^j = ak_i F_{i-1}^j + dk_i F_i^j + ck_i F_{i+1}^j + ek_i - e_i \quad (2.2.4) \quad \{\text{eq:bi}\}$$

$$\begin{pmatrix} d_0 & c_0 & & & \\ a_1 & d_1 & c_1 & & \\ & & \dots & & \\ & & & \dots & \\ & & & a_{N-2} & d_{N-2} & c_{N-2} \\ & & & & a_{N-1} & d_{N-1} \end{pmatrix} F^{j+1} = B^j \quad (2.2.5)$$

A questo punto procedo con il trasformare la matrice nella somma di una matrice identità e di una matrice con valori non nulli solo nelle celle sopra alla diagonale, faccio vedere i primi passaggi:

$$\begin{pmatrix} d_0 & c_0 & 0 & \dots & 0 \\ a_1 & d_1 & c_1 & \dots & 0 \\ & \cdot & \cdot & \cdot & \end{pmatrix} F^{j+1} = \begin{pmatrix} b_0 \\ b_1 \\ \dots \end{pmatrix} \rightarrow \begin{pmatrix} 1 & \frac{c_0}{d_0} & 0 & \dots & 0 \\ a_1 & d_1 & c_1 & \dots & 0 \\ & \cdot & \cdot & \cdot & \end{pmatrix} F^{j+1} = \begin{pmatrix} \frac{b_0}{d_0} \\ b_1 \\ \dots \end{pmatrix} \quad (2.2.6)$$

Proseguendo, chiamando $h_0 = \frac{c_0}{d_0}$ e $p_0 = \frac{b_0}{d_0}$, multiplico la prima riga per a_1 e la sottraggo alla seconda, in modo da eliminare a_1 dalla seconda riga:

$$\begin{pmatrix} 1 & h_0 & 0 & \dots & 0 \\ a_1 - a_1 & d_1 - a_1 h_0 & c_1 & \dots & 0 \\ & \cdot & \cdot & \cdot & \end{pmatrix} F^{j+1} = \begin{pmatrix} p_0 \\ b_1 - a_1 p_0 \\ \dots \end{pmatrix} \rightarrow \begin{pmatrix} 1 & h_0 & 0 & \dots & 0 \\ 0 & 1 & \frac{c_1}{d_1 - a_1 h_0} & \dots & 0 \\ & \cdot & \cdot & \cdot & \end{pmatrix} F^{j+1} = \begin{pmatrix} p_0 \\ \frac{b_1 - a_1 p_0}{d_1 - a_1 h_0} \\ \dots \end{pmatrix} \quad (2.2.7)$$

A questo punto chiamo $h_1 = \frac{c_1}{d_1 - a_1 h_0}$ e $p_1 = \frac{b_1 - a_1 p_0}{d_1 - a_1 h_0}$ e ripeto il ragionamento precedente sottraendo la seconda riga alla terza:

$$\begin{pmatrix} 1 & h_0 & 0 & \dots & 0 \\ 0 & 1 & h_1 & \dots & 0 \\ 0 & a_3 - a_3 & d_3 - a_3 h_1 & c_3 & \dots \\ & \cdot & \cdot & \cdot & \end{pmatrix} F^{j+1} = \begin{pmatrix} p_0 \\ p_1 \\ b_3 - a_3 p_1 \\ \dots \end{pmatrix} \rightarrow \begin{pmatrix} 1 & h_0 & 0 & \dots & 0 \\ 0 & 1 & h_1 & \dots & 0 \\ 0 & 0 & 1 & \frac{c_3}{d_3 - a_3 h_1} & \dots \\ & \cdot & \cdot & \cdot & \end{pmatrix} F^{j+1} = \begin{pmatrix} p_0 \\ p_1 \\ \frac{b_3 - a_3 p_1}{d_3 - a_3 h_1} \\ \dots \end{pmatrix} \quad (2.2.8)$$

A questo punto chiamo $h_3 = \frac{c_3}{d_3 - a_3 h_1}$ e $p_3 = \frac{b_3 - a_3 p_1}{d_3 - a_3 h_1}$ e proseguo, ottengo così le regole:

$$h_i = \frac{c_i}{d_i - a_i h_{i-1}} \quad (2.2.9) \quad \{\text{eq:hi}\}$$

e

$$p_i = \frac{b_i - a_i p_{i-1}}{d_i - a_i h_{i-1}} \quad (2.2.10) \quad \{\text{eq:pi}\}$$

A questo punto ho semplificato il sistema:

$$(2.2.11)$$

$$\begin{pmatrix} 1 & h_0 & & & \\ & 1 & h_1 & & \\ & & \dots & & \\ & & & \dots & \\ & & & & 1 & h_{N-2} \\ & & & & & 1 \end{pmatrix} F^{j+1} = P \quad (2.2.12)$$

Per risolvere il sistema devo calcolare il vettore delle P per poi ottenere i valori di F^{j+1} a partire dall'ultimo $F_{N-1}^{j+1} = p_{N-1}$ con la formula:

$$F_i^{j+1} = p_i + h_i F_{i+1}^{j+1} \quad (2.2.13)$$

A questo punto ho bisogno di conoscere come trattare le condizioni al contorno.

3 Condizioni al contorno

In seguito espongo come è possibile adottare alcune condizioni al contorno:

- Dirichlet: Conosco i valori della funzione negli estremi del dominio
- Neumann: Conosco i valori della derivata della funzione negli estremi del dominio
- Robin: Conosco una combinazione lineare tra il valore della funzione e la sua derivata negli estremi del dominio
- Miste: Negli estremi ho tipi differenti di condizioni al contorno

3.1 Dirichlet

Conosco il valore della funzione negli estremi del dominio.

$$F(x, t) = f(x, t) \forall x \in \partial D \quad (3.1.1)$$

Assegno a F_0^{j+1} e F_{N-1}^{j+1} il valore noto, e' quindi inutile calcolare la prima e l'ultima riga della matrice $N \times N$ e posso trattare tutto come se la matrice fosse $N - 2 \times N - 2$, con indici da 1 a $N - 2$ per tenere conto delle condizioni devo cambiare i valori:

$$\begin{array}{cc|cc} a'_0 = 0 & ak'_0 = 0 & a'_1 = 0 & ak'_1 = ak_1 \\ d'_0 = 1 & dk'_0 = 0 & d'_1 = d_1 & dk'_1 = dk_1 \\ c'_0 = 0 & ck'_0 = 0 & c'_1 = c_1 & ck'_1 = ck_1 \\ e'_0 = -F_0^{j+1} & ek'_0 = 0 & e'_1 = e_1 + a_1 F_0^{j+1} & ek'_1 = ek_1 \end{array} \quad (3.1.2)$$

Che equivale a scrivere:

$$\begin{array}{ccc} b'_0 = F_0^{j+1} & h'_0 = 0 & p'_0 = F_0^{j+1} \\ b'_1 = b_1 - a_1 F_0^{j+1} & h'_1 = \frac{c_1}{d'_1} & p'_1 = \frac{b'_1}{d'_1} \end{array} \quad (3.1.3)$$

Di conseguenza $F_1^{j+1} = p'_1 + h'_1 F_2^{j+1}$ e $F_0^{j+1} = p'_0 + h'_0 F_1^{j+1} = F_0^{j+1}$. Mentre se la condizione si presenta per l'ultimo punto del dominio:

$$\begin{array}{cc|cc} a'N - 2 = a_{N-2} & ak'_{N-2} = ak_{N-2} & a'_{N-1} = 0 & ak'_{N-1} = 0 \\ d'_{N-2} = d_{N-2} & dk'_{N-2} = dk_{N-2} & d'_{N-1} = 1 & dk'_{N-1} = 0 \\ c'_{N-2} = 0 & ck'_{N-2} = ck_{N-2} & c'_{N-1} = 0 & ck'_{N-1} = 0 \\ e'_{N-2} = e_{N-2} + c_{N-2} F_{N-1}^{j+1} & ek'_{N-2} = -F_{N-1}^{j+1} & e'_{N-1} = -F_{N-1}^{j+1} & ek'_{N-1} = 0 \end{array} \quad (3.1.4)$$

Che posso riscrivere:

$$\begin{array}{ccc} b'_{N-2} = b_{N-2} - c_{N-2} F_{N-1}^{j+1} & h'_{N-2} = 0 & p'_{N-2} = p_{N-2} \\ b'_{N-1} = F_{N-1}^{j+1} & h'_{N-1} = 0 & p'_{N-1} = F_0^{j+1} \end{array} \quad (3.1.5)$$

e di conseguenza $F_{N-1}^{j+1} = p'_{N-1} = F_{N-1}^{j+1}$ e $F_{N-2}^{j+1} = p'_{N-2} + h'_{N-2} F_{N-1}^{j+1} = p'_{N-2}$.

3.2 Neumann

Conosco il valore della derivata negli estremi del dominio.

$$\frac{\partial F}{\partial x}(x, t) = f(x, t) \forall x \in \partial D \quad (3.2.1)$$

La spiegazione e l'esempio per questa risoluzione lo fornisco nel paragrafo dedicato a Robin.

3.3 Robin

Conosco una combinazione lineare tra la derivata e il valore della funzione negli estremi del dominio.

$$\frac{\partial F}{\partial x}(x, t) + r(x, t)F(x, t) = f(x, t) \forall x \in \partial D \quad (3.3.1) \quad \{\text{eq:Robin}\}$$

Come prima, per mantenere la precisione del metodo (Δx^2) non posso usare la definizione centrale, ho quindi bisogno di inventarmi un "nodo fantasma" F_{-1}^{j+1} (o F_N^{j+1} se fosse la condizione nell'ultimo punto del dominio):

$$\frac{\partial F}{\partial x}(x(i=0), t(j=j+1)) = \frac{F_1^{j+1} - F_{-1}^{j+1}}{2\Delta x} \quad (3.3.2)$$

Prendo la (3.3.1) e faccio le adeguate sostituzioni per $i=0$ e al tempo geerico $j=n$:

$$\frac{F_1^n - F_{-1}^n}{2\Delta x} + R_0^n F_0^n = f_0^n \rightarrow F_{-1}^n = F_1^n + 2\Delta x (R_0^n F_0^n - f_0^n) \quad (3.3.3)$$

Partendo dalla forma matriciale del problema generico:

$$a_0 F_{-1}^{j+1} + d_0 F_0^{j+1} + c_0 F_1^{j+1} + e_0 = a k_0 F_{-1}^j + d k_0 F_0^j + c k_0 F_1^j + e k_0 \quad (3.3.4)$$

e sostituendo i vari F_{-1}^n :

$$a_0 \left(F_1^{j+1} + 2\Delta x (R_0^{j+1} F_0^{j+1} - f_0^{j+1}) \right) + d_0 F_0^{j+1} + c_0 F_1^{j+1} = a k_0 \left(F_1^j + 2\Delta x (R_0^j F_0^j - f_0^j) \right) + d k_0 F_0^j + c k_0 F_1^j \quad (3.3.5)$$

A questo punto raccolgo i termini con lo stesso punto della funzione:

$$(d_0 + 2a_0 R_0^{j+1} \Delta x) F_0^{j+1} + (c_0 + a_0) F_1^{j+1} - 2a_0 f_0^{j+1} \Delta x = (d k_0 + 2a k_0 R_0^j \Delta x) F_0^j + (c k_0 + a k_0) F_1^j - 2a k_0 f_0^j \Delta x \quad (3.3.6)$$

E quindi i parametri interessati della matrice diventano:

$$\begin{aligned} a'_0 &= 0 & a k'_0 &= 0 \\ d'_0 &= d_0 + 2a_0 R_0^{j+1} \Delta x & d k'_0 &= d k_0 + 2a k_0 R_0^j \Delta x \\ c'_0 &= a_0 + c_0 & c k'_0 &= a k_0 + c k_0 \\ e'_0 &= e_0 - 2a_0 f_0^{j+1} \Delta x & e k'_0 &= e k_0 - 2a k_0 f_0^j \Delta x \end{aligned} \quad (3.3.7)$$

Che equivale a scrivere:

$$\begin{aligned} b'_0 &= d k'_0 F_0^j + c k'_0 F_1^j + 2\Delta x (a_0 f_0^{j+1} - a k_0 f_0^j) \\ h'_0 &= \frac{c'_0}{d'_0} \\ p'_0 &= \frac{b'_0}{d'_0} \end{aligned} \quad (3.3.8)$$

e di conseguenza $F_0^{j+1} = p'_0 + h'_0 F_1^{j+1}$.

Mentre se la condizione si presenta come ultimo punto del dominio:

$$\frac{F_N^n - F_{N-2}^n}{2\Delta x} + R_{N-1}^n F_{N-1}^n = f_{N-1}^n \rightarrow F_N^n = F_{N-2}^n - 2\Delta x (R_{N-1}^n F_{N-1}^n - f_{N-1}^n) \quad (3.3.9)$$

Salto i passaggi, molto simili a quelli della spiegazione precedente, per il calcolo di b'_{N-1} andranno usati i seguenti parametri:

$$\begin{aligned} a'_{N-1} &= a_{N-1} + c_{N-1} & a k'_{N-1} &= a k_{N-1} + c k_{N-1} \\ d'_{N-1} &= d_{N-1} - 2c_{N-1} R_{N-1}^{j+1} \Delta x & d k'_{N-1} &= d k_{N-1} - 2c k_{N-1} R_{N-1}^j \Delta x \\ c'_{N-1} &= 0 & c k'_{N-1} &= 0 \\ e'_{N-1} &= e_{N-1} + c_{N-1} f_{N-1}^{j+1} \Delta x & e k'_{N-1} &= e k_{N-1} + c k_{N-1} f_{N-1}^{j+1} \Delta x \end{aligned} \quad (3.3.10)$$

Che equivale a scrivere:

$$\begin{aligned} b_{N-1} &= a k'_{N-1} F_{N-2}^j + d k'_{N-1} F_{N-1}^j - 2\Delta x (c_{N-1} f_{N-1}^{j+1} - c k_{N-1} f_{N-1}^j) \\ h'_{N-1} &= 0 \\ p'_{N-1} &= \frac{b'_{N-1} + a'_{N-1} p_{N-2}}{d'_{N-1} - a'_{N-1} h_{N-2}} \end{aligned} \quad (3.3.11)$$

e di conseguenza $F_{N-1}^{j+1} = p'_{N-1}$, anche perchè è il primo punto da cui si parte per calcolare il valore della funzione in $j+1$.

Se faccio in modo di eliminare il coefficiente che moltiplica il valore della funzione (gli R) ottengo le condizioni a contorno di Neuman.

3.4 Osservazione

Il modo in cui ho trattato i parametri per quanto riguarda le condizioni al contorno di Dirichlet nei punti 0 e $N - 1$, non è matematicamente correttissimo infatti i parametri andrebbero messi tutti a 0 in quanto quei punti non fanno parte dell'algoritmo. Ho impostato i valori per avere un algoritmo che possa svolgere il calcolo rispettando le condizioni al contorno senza sapere quali siano, mettendo nelle mani dell'utente che si occuperà di impostare i corretti parametri della matrice la gestione delle condizioni.

4 Applicazione

4.1 Equazione del calore

Riprendiamo l'equazione del calore:

$$\frac{\partial T}{\partial t}(x, t) = k \frac{\partial^2}{\partial x^2} T(x, t) \quad (4.1.1)$$

Per rispettare la convenzione di prima $D_2 = k$, $D_1 = U = V(x, t) = 0$.

A questo punto posso sostituire, con $\eta = k \frac{\Delta t}{\Delta x^2}$:

$$\begin{aligned} a_i^j &= -1 & ak_i^j &= 1 \\ d_i^j &= \frac{2}{\eta} + 2 & dk_i^j &= \frac{2}{\eta} - 2 \\ c_i^j &= -1 & ck_i^j &= 1 \\ e_i^j &= 0 & ek_i^j &= 0 \end{aligned} \quad (4.1.2) \quad \{\text{eq:par}$$

e procedere con i calcoli.

4.2 Equazione di Schrödinger

Lavoriamo con l'equazione di Schrödinger dipendente dal tempo:

$$i\hbar \frac{\partial \psi}{\partial t}(x, t) = \left[-\frac{\hbar^2}{2m} \frac{\partial^2}{\partial x^2} + V(x, t) \right] \psi(x, t) \quad (4.2.1)$$

prima di tutto portiamola in una forma tale che non ci sia nulla a moltiplicare la derivata temporale:

$$\frac{\partial \psi}{\partial t}(x, t) = \left[i \frac{\hbar}{2m} \frac{\partial^2}{\partial x^2} + \frac{A(x, t)}{i\hbar} \right] \psi(x, t) \quad (4.2.2)$$

Per rispettare la convenzione di prima $D_2 = i \frac{\hbar}{2m}$, $D_1 = U = 0$ e $V(x, t) = \frac{A(x, t)}{i\hbar}$. Ho usato A per indicare il potenziale in modo da evitare confusione.

A questo punto posso sostituire, con $\eta = i \frac{\hbar}{2m} \frac{\Delta t}{\Delta x^2}$:

$$\begin{aligned} a_i^j &= -1 & ak_i^j &= 1 \\ d_i^j &= \frac{1}{\eta} \left(2 - \Delta t V_i^{j+1} \right) + 2 & dk_i^j &= \frac{1}{\eta} \left(2 + \Delta t V_i^j \right) - 2 \\ c_i^j &= -1 & ck_i^j &= 1 \\ e_i^j &= 0 & ek_i^j &= 0 \end{aligned} \quad (4.2.3) \quad \{\text{eq:par}$$

5 Il programma

Per semplicità e alleggerire i calcoli ho impostato $\hbar = 1$, Gli eseguibili compilabili dal Makefile sono i seguenti, nei commenti è spiegato come eseguirli:

```
#https://root.cern.ch/interacting-shared-libraries-rootcint
#risolve un'equazione di Schodinger: come primo argomento il file del potenziale
#come secondo, opzionale, il file che indica le CI, di base e' gauss.set
single: MainC.cpp TridiagC.o CrankSolverC.o impostazioniC.o
    @echo Compilo $@
    @$(CC11) $(CFLAGS) -o $@ $^ -DUSECOMPLEX
#disegna soluzione, errore e pesi dell'integrale del file di dati in argomento
drawer: drawer.cpp
    @echo Compilo $@
    @$(CC11) $(CFLAGS) -o $@ $^ $(LIBROOT) $(CFLAGSROOT)
#controlla i .dat letti in namelist.txt
analisi: analisi.cpp
    @echo Compilo $@
    @$(CC11) $(CFLAGS) -o $@ $^ $(LIBROOT) $(CFLAGSROOT)
#si comporta come maincrankC, solo che carica i nomi dei file del potenziale da
#namelist.txt, e come argomento opzionale ha il file delle CI
experiment: Experiment.cpp TridiagC.o CrankSolverC.o impostazioniC.o
    @echo Compilo $@
    @$(CC11) $(CFLAGS) -o $@ $^ -DUSECOMPLEX
#visualizza vari potenziali in diversi files.
CVD: CVDrawer.cpp impostazioniC.o TridiagC.o
    @echo Compilo $@
    @$(CC11) $(CFLAGS) -o $@ $^ -DUSECOMPLEX $(LIBROOT) $(CFLAGSROOT)
#una interfaccia grafica per impostare un esperimento alla volta
gui: main.cpp Schrody.o DefineCC.o TridiagC.o CrankSolverC.o libSchrody.so libCC.so
    @echo Compilo il main
    @$(CC11) $(CFLAGS) -o $@ -DSTANDALONE $^ $(LIBROOT) $(CFLAGSROOT)
```

5.1 Impostazioni

Il programma più semplice, compilabile con `$> make maincrankC` carica i dati da tre file di impostazioni: il file generale delle impostazioni:

Listing 1: Impostazioni principali

```
#massa
1
#lunghezza
60
#tmax
10
#Ndipende_da_int=0
0
#Nl
10000
#spacestep.
0.0015
#Nt
1000
#timestep
0.001
#timeskip
100
```

```
#spaceskip
20
#precisione
5e-3
```

Il file delle condizioni iniziali:

Listing 2: Impostazioni CI

```
#Bump_Gauss
G
#impostazioni_onda_norm_stdev_midpoint
1 2 5
#impostazionicc0_012DCR_val_pesorobin
2 0 0
#impostazioniccN_012DCR_val_pesorobin
2 0 0
#Energia
10
```

Il file del potenziale:

Listing 3: Impostazioni potenziale

```
#tipo_Gauss_Muro_Salto
M
#impostazioni_Val_Pos_Vpar
0 29 31
```

5.2 Lancio senza potenziale: errore

Prima di tutto voglio effettuare qualche simulazione senza potenziale per poter osservare come si comporta l'onda scelta dell'integrale -¿ errore varie scelte di passi spaziali e temporali e visualizzazione dei vari errori

5.3 Condizioni al contorno

Quello che io voglio simulare è un pacchetto d'onda che sia partito da $-\infty$ e arrivi nel "punto interessante", che è l'area che osservo nella simulazione e poi prosegue fino all'infinito (o venga eventualmente riflesso).

Per quanto riguarda le condizioni al contorno ho avuto idee:

- Non tengo conto delle condizioni al contorno, faccio in modo di usare pacchetti d'onda ristretti e blocco la simulazione quando questa va ad avvicinarsi troppo ai bordi.
- Uso una forma d'onda tale per cui conosco la funzione e la sua derivata in ogni punto
- Emulo un ambiente chiuso (con ai lati muri di potenziale alti ∞)

La prima ipotesi è stata quella che ho usato in tutte le prove che ho effettuato prima di iniziare a raccogliere dati. Per la seconda ipotesi dovrei usare come condizione iniziale una funzione e calcolarne la derivata per ogni passo negli estremi a priori il discorso non si applica ma i pacchetti d'onda hanno una dispersione tendono ad allargarsi, per cui se per esempio avessi un pacchetto d'onda generico ($\Psi(x, t) = e^{ikx}\psi(x, t)$) la relazione dovrebbe essere $\partial_x \Psi(x, t) = ik\Psi(x, t) + e^{ikx}\partial_x \psi(x, t)$

5.4 Dispersione e forme dei pacchetti d'onda

dispersione teoria e pratica

5.5 Condizioni iniziali

teoria simulazione

5.6 Dispersione

teoria simulazione

5.7 Potenziali

teoria simulazione