

Laboratorio di Fisica Computazionale

Luca Cassia - [MAT. 728341]

Spring 2012

Indice

1	Integrazione Numerica	2
1.1	Newton-Cotes	2
1.2	Quadrature Gaussiane	3
1.2.1	Integrazione Composta	5
1.3	Monte Carlo	5
1.3.1	Campionamento di Importanza	5
1.4	Esempi	6
1.4.1	Newton-Cotes 1	7
1.4.2	Newton-Cotes 2	8
1.4.3	Quadratura Gaussiana 5	9
1.4.4	Monte Carlo	9
2	Algoritmo Metropolis	11
2.1	Oscillatore Armonico	12
2.1.1	Azione e Termalizzazione	13
2.1.2	Autocorrelazione	14
2.1.3	Funzioni di Correlazione	16
2.1.4	Calcolo di Grandezze Quanto-Meccaniche Secondarie .	17
3	Metodo Runge-Kutta	21
3.1	Simulazione di Sistemi Dinamici	22
3.1.1	Pendolo Caotico	24
4	Metodo Implicito	26
4.1	Equazione di Schrödinger	27
4.1.1	Scattering	29

Capitolo 1

Integrazione Numerica

In analisi numerica, l'integrazione numerica consiste in un insieme di metodi che stimano il valore di un integrale definito, senza dover calcolare la primitiva della funzione integranda. In questa sezione si illustrano alcuni dei principali metodi deterministici e non deterministici.

1.1 Newton-Cotes

Le regole di quadratura di Newton-Cotes sono formule che consistono nel valutare l'integrandi in punti equispaziati dell'intervallo di integrazione.

Si assume che il valore di una funzione $f : [a, b] \subset \mathbb{R} \rightarrow \mathbb{R}$ sia noto nei punti x_i , per $i = 0, \dots, n$ tali che

$$x_i = a + \left(\frac{b-a}{n} \right) i$$

La formula di Newton-Cotes di grado n si ottiene interpolando f nei punti x_i con i polinomi della base di Lagrange [3], e integrando la polinomiale risultante, $L(x)$, nell'intervallo $[a, b]$.

$$\int_a^b f(x) dx \approx \int_a^b L(x) dx = \int_a^b \left(\sum_{i=0}^n f(x_i) \ell_i(x) \right) dx = \sum_{i=0}^n f(x_i) \underbrace{\int_a^b \ell_i(x) dx}_{w_i}$$

dove gli $\ell_i(x)$ sono i polinomi di Lagrange così definiti

$$\ell_i(x) := \prod_{\substack{0 \leq j \leq n \\ j \neq i}} \frac{x - x_j}{x_i - x_j}$$

$$\ell_i(x_j) = \delta_{ij}$$

La formula di Newton-Cotes assume così la semplice forma di media dei valori $f(x_i)$ pesati sui coefficienti w_i (indipendenti da f)

$$\int_a^b f(x) dx \approx \sum_{i=0}^n w_i f(x_i)$$

Si dimostra che l'errore dell'interpolazione di f con un polinomio è

$$E(x) = \frac{1}{(n+1)!} f^{(n+1)}(\xi(x)) \prod_{i=0}^n (x - x_i)$$

per un certo $\xi \in [a, b]$ dipendente da x . Non avendo tuttavia alcuna informazione su come individuare il punto ξ in genere si effettua solo una stima del limite superiore sull'errore $E(x)$ per ξ tale che $f(\xi) = \max_{x \in [a,b]} f(x)$.

Al primo ordine dell'approssimazione Newton-Cotes la formula si riduce al cosiddetto metodo dei “trapezi”:

$$\int_a^b f(x) dx = \frac{b-a}{2} [f(a) + f(b)] + \underbrace{\frac{1}{2} \int_a^b f''(\xi(x))(x-a)(x-b)}_{E_1}$$

mentre al secondo ordine non è altro che la regola di Simpson:

$$\begin{aligned} \int_a^b f(x) dx &= \frac{b-a}{3} \left[f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right] + \\ &\quad + \underbrace{\frac{1}{6} \int_a^b f'''(\xi(x))(x-a)(x-\frac{a+b}{2})(x-b)}_{E_2} \end{aligned}$$

1.2 Quadrature Gaussiane

La regola di quadratura Gaussiana a n -punti è un metodo di integrazione numerica costruito in modo tale da fornire un risultato esatto per polinomi di grado inferiore a $2n$, attraverso una scelta appropriata di punti x_i e pesi w_i , per $i = 1, \dots, n$. Sul dominio di integrazione convenzionale $[-1, 1]$ la regola è così formulata

$$\int_{-1}^1 f(x) dx \approx \sum_{i=1}^n w_i f(x_i)$$

L'accuratezza del risultato è tanto più grande quanto meglio f è approssimata da un polinomio. Se tuttavia la funzione integranda può essere scritta

come $f(x) = W(x)g(x)$, dove $g(x)$ è approssimativamente polinomiale e $W(x)$ è nota, allora esistono w'_i tali che

$$\int_{-1}^1 f(x) dx = \int_{-1}^1 W(x)g(x) dx \approx \sum_{i=1}^n w'_i g(x_i)$$

$W(x)$ viene detta funzione peso, mentre i punti x_i sono le radici di un polinomio appartenente alla classe dei polinomi ortogonali.

Nel caso considerato $W(x) = 1$ ed i polinomi associati sono i polinomi ortogonali di Legendre $P_n(x)$ [3]. Il peso i -esimo associato al nodo Gaussiano x_i è dato da

$$w_i = \frac{2}{(1-x_i^2)[P'_n(x_i)]^2}$$

Analogamente alle regole di Newton-Cotes, l'errore teorico del metodo della quadratura Gaussiana è

$$E(x) = \frac{1}{n!} f^{(n)}(\xi(x)) \prod_{i=1}^n (x - x_i)$$

che può tuttavia essere ridotto a

$$E(x) = \frac{1}{(2n)!} f^{(2n)}(\xi(x)) \prod_{i=1}^n (x - x_i)^2$$

semplicemente considerando due volte ogni punto di interpolazione x_i .

Infine, se si vuole calcolare l'integrale su $[a, b]$ invece che sull'intervallo $[-1, 1]$, si deve effettuare il cambio di variabile

$$\int_a^b f(x) dx = \frac{b-a}{2} \int_{-1}^1 f\left(\frac{b-a}{2}x + \frac{a+b}{2}\right) dx$$

ed applicando il metodo di Gauss si ottiene

$$\int_a^b f(x) dx \approx \frac{b-a}{2} \sum_{i=1}^n w_i f\left(\frac{b-a}{2}x_i + \frac{a+b}{2}\right)$$

I nodi ed i pesi per il polinomio di Legendre di quinto grado sono riportati in tabella:

x_i	w_i
0	$\frac{128}{255}$
$\pm \frac{1}{3} \sqrt{5 - 2\sqrt{10/7}}$	$\frac{322+13\sqrt{70}}{900}$
$\pm \frac{1}{3} \sqrt{5 + 2\sqrt{10/7}}$	$\frac{322-13\sqrt{70}}{900}$

1.2.1 Integrazione Composta

Una tecnica utile a migliorare la precisione del calcolo dell'integrale numerico consiste nello spezzare l'intervallo d'integrazione in n sottointervalli

$$\int_a^b f(x) dx = \sum_{i=0}^{n-1} \int_{\frac{b-a}{n}i}^{\frac{b-a}{n}(i+1)} f(x) dx$$

Dopodiché si applica uno dei metodi di quadratura numerica descritto ad ogni intervallino ed infine si somma ad ottenere il risultato richiesto.

1.3 Monte Carlo

L'integrazione Monte Carlo [8], a differenza dei metodi di quadratura precedentemente descritti, fa uso di sampling casuali e per questo motivo rientra nella categoria dei metodi non deterministici.

Nella sua versione più semplice l'algoritmo consiste nell'estrarrre uniformemente punti dalla regione di integrazione per stimare l'integrale ed il relativo errore. Si supponga che il sample sia costituito da N punti x_1, \dots, x_N appartenenti alla regione di integrazione di misura V , allora la stima dell'integrale è data da

$$I \approx E_N \equiv V \frac{1}{N} \sum_{i=1}^N f(x_i) = V \langle f \rangle$$

Poichè $\{x_i\}$ è una sequenza di punti equidistribuiti in V , si può dimostrare che $I = \lim_{N \rightarrow \infty} E_N$. Tenendo presente che la varianza della funzione integranda è

$$\text{Var}(f) \equiv \sigma_N^2 = \frac{1}{N-1} \sum_{i=1}^N (f(x_i) - \langle f \rangle)^2$$

la varianza di E_N è quindi

$$\text{Var}(E_N) = \frac{V^2}{N^2} \sum_{i=1}^N \text{Var}(f) = V^2 \frac{\text{Var}(f)}{N} = V^2 \frac{\sigma_N^2}{N}$$

Dal momento che le considerazioni appena fatte rimangono valide anche nel caso multidimensionale, quello che si deduce è che l'errore sulla stima dell'integrale scala come $1/\sqrt{N}$, indipendentemente dal numero di dimensioni.

1.3.1 Campionamento di Importanza

Dal punto di vista metematico, il campionamento di importanza corrisponde al cambio di variabile

$$\int f(x) dx = \int \frac{f(x)}{p(x)} p(x) dx = \int \frac{f(x)}{p(x)} dP(x)$$

con

$$p(x) = \frac{\partial^d}{\partial x_1 \dots \partial x_d} P(x)$$

Se si restringe $p(x)$ ad essere una funzione a valori non negativi normalizzata all'unità, allora si può interpretare $p(x)$ come una densità di probabilità. Se poi si ha a disposizione un generatore di numeri casuali corrispondente alla distribuzione $P(x)$ si può anche stimare l'integrale da un sample x_1, \dots, x_N di numeri casuali distribuiti secondo $P(x)$

$$E_N = \frac{1}{N} \sum_{n=1}^N \frac{f(x_n)}{p(x_n)}$$

L'errore statistico dell'integrazione Monte Carlo è dato da $\sigma(f/p)/\sqrt{N}$.

Il campionamento di importanza è efficace se si sceglie $p(x)$ tale che approssimi bene $|f(x)|$ e tale che si sia capaci di generare numeri casuali con distribuzione di probabilità $P(x)$.

1.4 Esempi

Si dimostrano ora i metodi descritti nei paragrafi precedenti applicandoli al calcolo di alcuni integrali particolari. Le funzioni integrande scelte sono:

funzione	primitiva
$\log(1 + x)$	$-x + \log(1 + x) + x \log(1 + x)$
$x^9 - x^7 + 3$	$\frac{x^{10}}{10} - \frac{x^8}{8} + 3x$

Tutti gli integrali sono calcolati tenendo fisso l'intervallo d'integrazione $I = [1, 2]$ e variando il numero di suddivisioni per ogni metodo. Conoscendo inoltre l'espressione analitica delle primitive delle due funzioni integrande è possibile calcolare i risultati esatti ed usare questi ultimi per confrontare l'errore del risultato numerico con l'errore previsto. Si sono calcolati analiticamente dunque alcuni valori utili alla stima di tale errore:

f	$\max(f^{(1)})$	$\max(f^{(4)})$
$\log(1 + x)$	0.5	-0.0740742
$x^9 - x^7 + 3$	1856	90048

In seguito si indicheranno con $E(f)$ l'errore stimato analiticamente e con $\Delta(f)$ l'errore calcolato numericamente.

1.4.1 Newton-Cotes 1

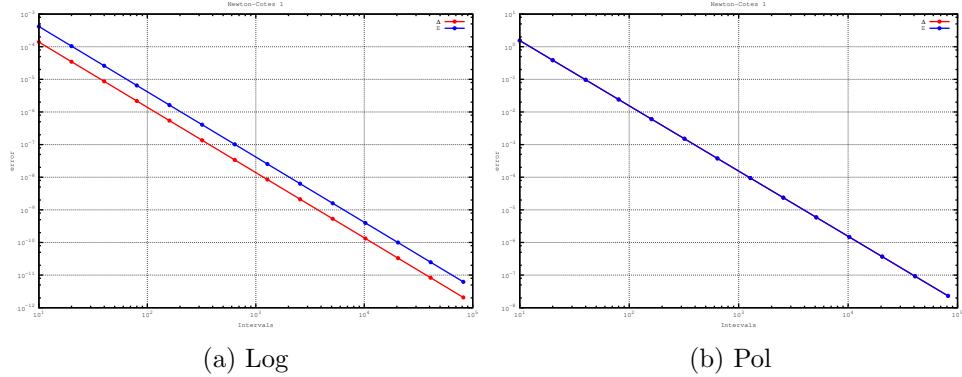


Figura 1.1: Confronto errore NC1 calcolato numericamente (rosso) e previsto analiticamente (blu).

n	$\Delta(Log)$	$E(Log)$	n	$\Delta(Pol)$	$E(Pol)$
10	1.388645e-04	4.166667e-04	10	1.541035e+00	1.546667e+00
20	3.472070e-05	1.041667e-04	20	3.860018e-01	3.866667e-01
40	8.680460e-06	2.604167e-05	40	9.654698e-02	9.666667e-02
80	2.170133e-06	6.510417e-06	80	2.413966e-02	2.416667e-02
160	5.425343e-07	1.627604e-06	160	6.035096e-03	6.041667e-03
320	1.356337e-07	4.069010e-07	320	1.508785e-03	1.510417e-03
640	3.390842e-08	1.017253e-07	640	3.771970e-04	3.776042e-04
1280	8.477104e-09	2.543132e-08	1280	9.429930e-05	9.440104e-05
2560	2.119278e-09	6.357829e-09	2560	2.357483e-05	2.360026e-05
5120	5.298190e-10	1.589457e-09	5120	5.893707e-06	5.900065e-06
10240	1.324540e-10	3.973643e-10	10240	1.473427e-06	1.475016e-06
20480	3.311029e-11	9.934107e-11	20480	3.683568e-07	3.687541e-07
40960	8.277934e-12	2.483527e-11	40960	9.208927e-08	9.218852e-08
81920	2.073119e-12	6.208817e-12	81920	2.302198e-08	2.304713e-08

Come previsto, l'errore calcolato segue l'andamento ricavato analiticamente pur rimanendo sempre al di sotto di quest'ultimo. $E(f)$ infatti è solo il limite superiore per l'errore $\Delta(f)$. Nel caso della polinomiale inoltre i due errori risultano particolarmente vicini.

1.4.2 Newton-Cotes 2

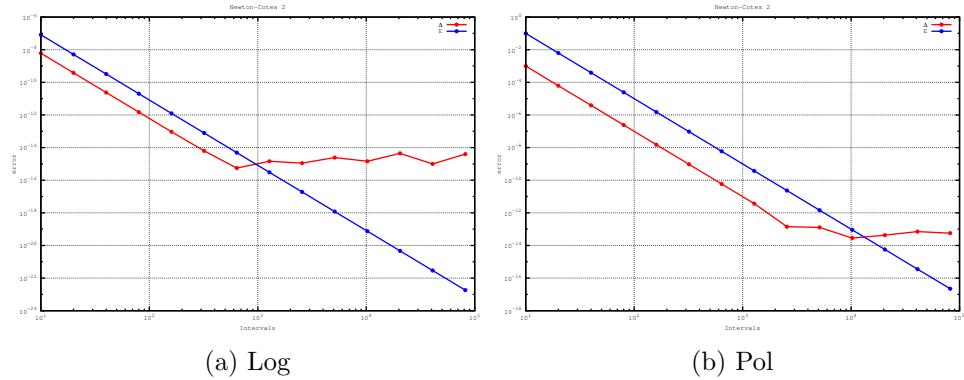


Figura 1.2: Confronto errore NC2 calcolato numericamente (rosso) e previsto analiticamente (blu).

n	$\Delta(Log)$	$E(Log)$	n	$\Delta(Pol)$	$E(Pol)$
10	6.101824e-09	8.230467e-08	10	9.908609e-04	1.000533e-01
20	3.816787e-10	5.144042e-09	20	6.203492e-05	6.253333e-03
40	2.386025e-11	3.215026e-10	40	3.878841e-06	3.908333e-04
80	1.491696e-12	2.009391e-11	80	2.424535e-07	2.442708e-05
160	9.325873e-14	1.255870e-12	160	1.515373e-08	1.526693e-06
320	6.328271e-15	7.849185e-14	320	9.471108e-10	9.541829e-08
640	5.551115e-16	4.905740e-15	640	5.920242e-11	5.963643e-09
1280	1.443290e-15	3.066088e-16	1280	3.666401e-12	3.727277e-10
2560	1.110223e-15	1.916305e-17	2560	1.421085e-13	2.329548e-11
5120	2.442491e-15	1.197691e-18	5120	1.278977e-13	1.455968e-12
10240	1.443290e-15	7.485566e-20	10240	2.842171e-14	9.099798e-14
20480	4.440892e-15	4.678479e-21	20480	4.263256e-14	5.687374e-15
40960	9.992007e-16	2.924049e-22	40960	7.105427e-14	3.554608e-16
81920	3.996803e-15	1.827531e-23	81920	5.684342e-14	2.221630e-17

L'errore per il metodo NC2 si comporta come previsto finché non raggiunge valori dell'ordine di grandezza di 10^{-14} per i quali la precisione finita del calcolatore limita la precisione del calcolo introducendo del rumore nei dati.

1.4.3 Quadratura Gaussiana 5

n	$\Delta(Log)$	n	$\Delta(Pol)$
10	2.220446e-16	10	0.000000e+00
20	2.220446e-16	20	0.000000e+00
40	1.110223e-16	40	0.000000e+00
80	0.000000e+00	80	0.000000e+00
160	5.551115e-16	160	0.000000e+00
320	1.110223e-15	320	1.421085e-14
640	2.220446e-16	640	0.000000e+00
1280	1.332268e-15	1280	1.421085e-14
2560	1.110223e-15	2560	1.136868e-13
5120	2.664535e-15	5120	1.136868e-13
10240	1.110223e-16	10240	1.421085e-14
20480	4.440892e-15	20480	4.263256e-14
40960	9.992007e-16	40960	8.526513e-14
81920	3.996803e-15	81920	5.684342e-14

Nel caso della quadratura gaussiana di quinto grado l'errore decresce molto più rapidamente, tanto che già con poche decine di intervalli il risultato può essere considerato sufficientemente preciso. Dopo di chè i dati sono troppo disturbati dal rumore per poterne ricavare un andamento funzionale. Si nota inoltre che per il polinomio l'errore è esattamente zero come previsto dalla teoria.

1.4.4 Monte Carlo

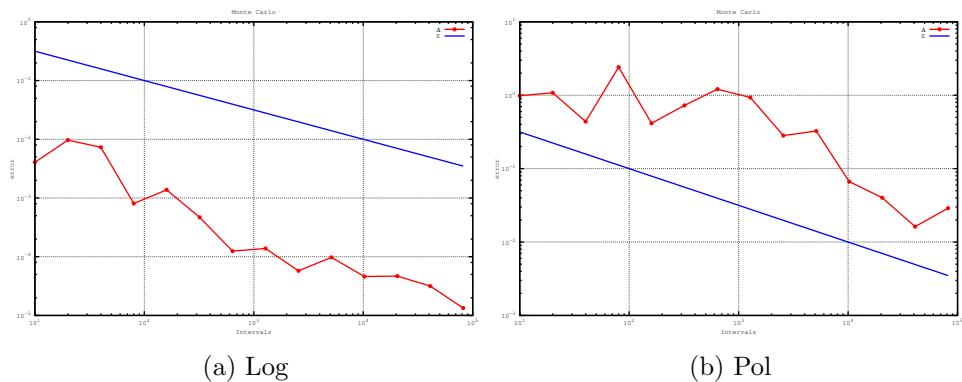


Figura 1.3: In rosso l'errore MC in funzione del numero di elementi del sample ed in blu la funzione $1/\sqrt{N}$.

Per analizzare il comportamento dell'errore dei metodi Molte Carlo si studia ora il caso di un integrale di volume bidimensionale. Il dominio di integrazione è l'insieme $\Omega = \{(x, y) \in \mathbb{R} \mid x^2 + y^2 < 1 \wedge x, y > 0\}$ mentre la funzione integranda non è altro che la costante 1. Si utilizza la convenzionale funzione di misura $d\mu = dx dy$. La misura di Ω allora è facile da calcolare essendo l'area di un quarto di circonferenza goniometrica:

$$\int_{\Omega} d\mu = \frac{\pi}{4}$$

La distribuzione dell'errore MC in funzione del numero di elementi del sample deve verificare la relazione $\sigma_{MC} \simeq 1/\sqrt{N}$.

In figura 1.4 sono rappresentati in scala bilogaritmica i dati ottenuti da 50000 esecuzioni dell'algoritmo implementante il metodo Monte Carlo. In ascissa sono riportati i numeri delle dimensioni del sample mentre sulle ordinate sono ripotati i rispettivi errori.

Come previsto, la distribuzione ottenuta risulta centrata sulla curva $1/\sqrt{N}$ che è proprio l'errore stimato teoricamente.

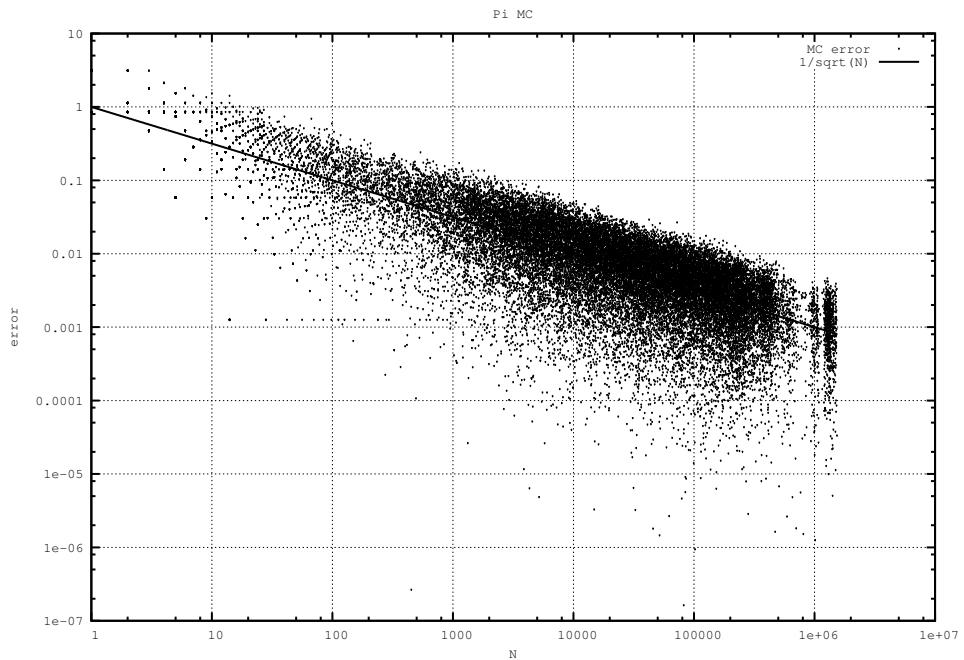


Figura 1.4: Distribuzione dell'errore MC in funzione del numero di elementi del sample. La linea continua rappresenta la funzione $1/\sqrt{N}$.

Capitolo 2

Algoritmo Metropolis

Il Metropolis è l'algoritmo più influenziale fra quelli appartenenti alla classe dei metodi Monte Carlo. Supportato da un'ampia teoria sulle catene di Markov [1], questo algoritmo costituisce uno strumento fondamentale per la scienza della computazione.

In questa sezione si propone di sviluppare un algoritmo Metropolis per simulare un oscillatore armonico quantistico e confrontare i risultati numerici con la teoria.

Il metodo Metropolis [2] nasce dalla necessità pratica di dover generare numeri casuali distribuiti con una densità di probabilità $p(x_1, \dots, x_d)$, che non necessariamente fattorizza. Sia il vettore $\phi = (x_1, \dots, x_d)$ uno stato dell'ensemble, che vogliamo generare. L'algoritmo consiste nel partire da uno stato iniziale ϕ_0 , e sostituire iterativamente uno stato vecchio con uno nuovo, in maniera tale da ottenere la distribuzione corretta nel limite di un gran numero di iterazioni. La distribuzione che viene raggiunta all'equilibrio è indipendente dallo stato iniziale ϕ_0 . Una volta che tale distribuzione è stata raggiunta, l'applicazione ripetuta dell'algoritmo mantiene il sistema all'interno dello stesso ensemble. In altre parole, la distribuzione desiderata costituisce l'unico punto fisso dell'algoritmo.

Due condizioni importanti devono essere soddisfatte per poter applicare il Metropolis: Ergodicità e bilancio dettagliato. La condizione di bilancio dettagliato afferma che le probabilità di transizione $W(\phi_1 \rightarrow \phi_2)$ e $W(\phi_2 \rightarrow \phi_1)$ rispettano l'equazione

$$p(\phi_1)W(\phi_1 \rightarrow \phi_2) = p(\phi_2)W(\phi_2 \rightarrow \phi_1)$$

L'ergodicità invece richiede che ogni stato dell'ensemble possa essere raggiunto in un numero finito di steps.

Dato uno stato iniziale ϕ_1 , un'iterazione dell'algoritmo consiste delle seguenti istruzioni:

- generare con distribuzione piatta un nuovo candidato ϕ' nell'intorno di ϕ di raggio δ

- calcolare $\Delta S = -\log(p(\phi')/p(\phi_1))$
- se $\Delta S < 0$ impostare il nuovo stato $\phi_2 = \phi'$
- se $\Delta S > 0$ accettare il nuovo candidato con probabilità $p(\phi')/p(\phi_1)$, altrimenti mantenere ϕ_1
- passare all'iterazione successiva

L'algoritmo Metropolis tuttavia presenta alcuni svantaggi, tra i quali il fatto che quando si raggiunge la distribuzione d'equilibrio, gli stati successivi sono in genere molto correlati. Se si vuole ottenere un unbiased sample di stati ϕ_i si possono quindi trascurare un numero τ_d di stati prima di passare a quello successivo. Il numero di steps τ_d viene denominato tempo di decorrelazione ed è dell'ordine di ξ^2 , dove ξ è una tipica distanza di correlazione del sistema. Ciò è dovuto al fatto che il Metropolis aggiorna gli stati localmente in maniera casuale. L'algoritmo consiste quindi nell'eseguire un random walk attraverso lo spazio delle configurazioni, il che richiede ξ^2 passaggi per effettuare uno spostamento di una distanza ξ in una certa direzione.

2.1 Oscillatore Armonico

Il sistema è costituito da una particella che si muove in uno spazio unidimensionale e in un reticolo temporale finito di passo a e lunghezza N con condizioni di periodicità al contorno. Le variabili del sistema sono quindi le coordinate x_t della particella ai vari istanti di tempo t , con la condizione $x_N \equiv x_0$. La particella interagisce inoltre con un potenziale armonico della forma

$$V(x) = \frac{m}{2} \omega^2 x^2$$

In meccanica quantistica la stima di un osservabile può essere estratta dalle funzioni di correlazione del sistema [5]. Si calcola quindi il correlatore delle variabili l -esima e k -esima $\langle x_l x_k \rangle$, definito dalla formula

$$\langle x_l x_k \rangle = \frac{\int Dx x_l x_k e^{-S_E}}{\int Dx e^{-S_E}}$$

Calcolare il correlatore significa calcolare un *path integral* [4] usando come ampiezza di probabilità il fattore e^{-S_E} . Il ruolo dell'algoritmo Metropolis è dunque quello di generare configurazioni del sistema con distribuzione di ampiezza di probabilità e^{-S_E} .

Dopo aver ottenuto una stima per il correlatore delle variabili x_l e x_k per ogni combinazione di (l, k) , si ricavano infine i valori di $\Delta E = (\tilde{E}_1 - \tilde{E}_0)$ e dell'elemento di matrice $W = \langle \tilde{E}_0 | \hat{x} | \tilde{E}_1 \rangle$ invertendo la relazione

$$\langle x_l x_k \rangle = 2 |\langle \tilde{E}_0 | \hat{x} | \tilde{E}_1 \rangle|^2 \exp \left(-\frac{Na}{2} (\tilde{E}_1 - \tilde{E}_0) \right) \cosh \left[a \left(\frac{N}{2} - |l - k| \right) (\tilde{E}_0 - \tilde{E}_1) \right].$$

2.1.1 Azione e Termalizzazione

Partendo da uno stato ϕ_0 casuale, è necessario lasciare del tempo affinchè il sistema si porti alla *pdf* di equilibrio. Tale processo, denominato termalizzazione, richiede in questo caso non più di 500 cicli di Metropolis, eseguiti i quali sarà possibile estrarre le configurazioni con la distribuzione desiderata. Come condizione iniziale si possono inizializzare tutte le variabili a zero (cold start) oppure a valori random (hot start).

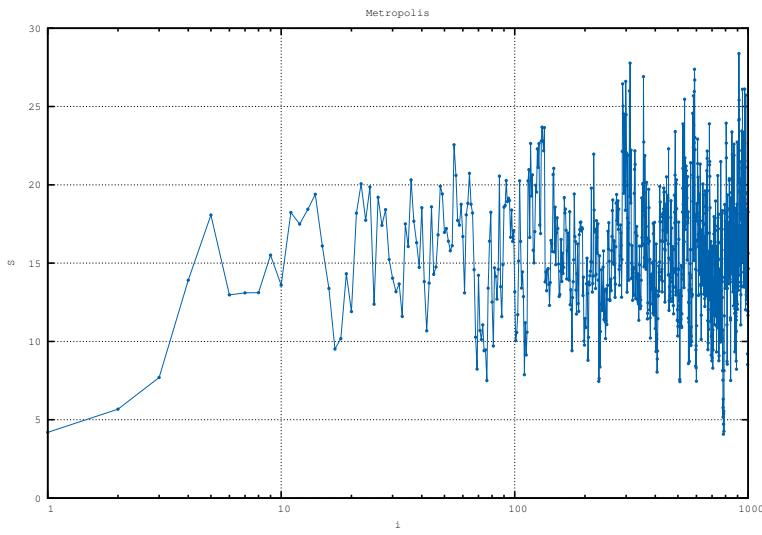


Figura 2.1: Cold Start

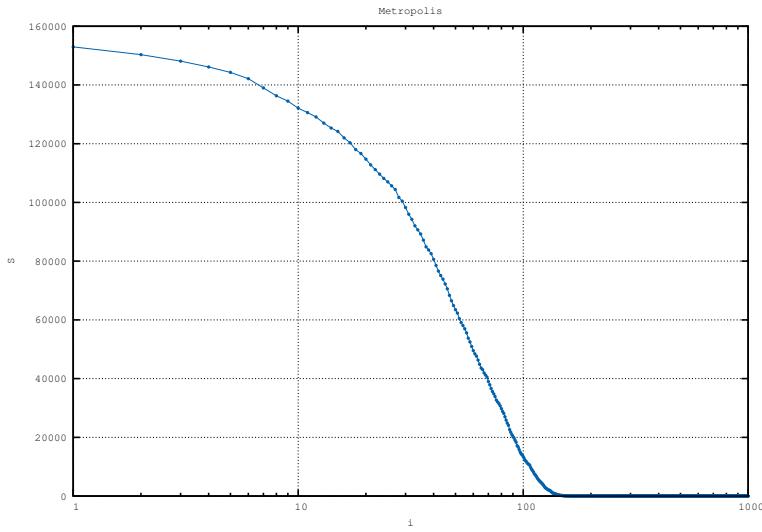


Figura 2.2: Hot Start

Per il calcolo dell'azione euclidea si usa la formula

$$S = a \sum_{i=0}^{N-1} \mathcal{L}(x_i, x_{i+1})$$

dove

$$\mathcal{L}(x_i, x_{i+1}) = \frac{m}{2} \left(\frac{x_i - x_{i+1}}{a} \right)^2 - \frac{1}{2} V(x_i) - \frac{1}{2} V(x_{i+1})$$

Tuttavia, per calcolare $\Delta S = S' - S$ si può tenere presente il fatto che ad ogni estrazione solo una variabile di sistema viene modificata e quindi tutti i termini delle due sommatorie che non contengono quella variabile si elidono nella differenza. Pertanto si utilizza la formula ridotta

$$\begin{aligned} \Delta S_i &= a[\mathcal{L}(x_{i-1}, x'_i) + \mathcal{L}(x'_i, x_{i+1}) - \mathcal{L}(x_{i-1}, x_i) - \mathcal{L}(x_i, x_{i+1})] \\ &= \frac{m}{a}[(x_i - x'_i)(x_{i+1} + x_{i-1}) + (x'^2_i - x_i^2)] + aV(x'_i) - aV(x_i) \end{aligned}$$

È facile verificare che l'algoritmo e i risultati possono essere resi indipendenti dal fattore di reticolo a se si ridefiniscono propriamente i parametri del problema, il che è equivalente a porre $a = 1$.

2.1.2 Autocorrelazione

Si studia ora la correlazione tra configurazioni generate a step consecutivi dell'algoritmo per ottenere una stima del tempo di decorrelazione τ_d del sistema. Tale dipendenza fra elementi della catena viene anche denominata *autocorrelazione*.

La formula per il correlatore di due stime dell'osservabile O calcolate da stati estratti a istanti di tempo distanti τ è

$$R(\tau) = \frac{\langle O_i O_{i+\tau} \rangle - \langle O_i \rangle^2}{\langle O_i^2 \rangle - \langle O_i \rangle^2} = \frac{\frac{1}{n-\tau} \sum_{i=0}^{n-\tau} O_i O_{i+\tau} - \langle O_i \rangle^2}{\langle O_i^2 \rangle - \langle O_i \rangle^2} \simeq e^{-\frac{\tau}{\tau_d}}$$

Il grafico dell'autocorrelazione $R(\tau)$ per l'osservabile $O(|l - k|) = \langle x_l x_k \rangle$ evidenzia la perdita esponenziale di correlazione nel segnale.

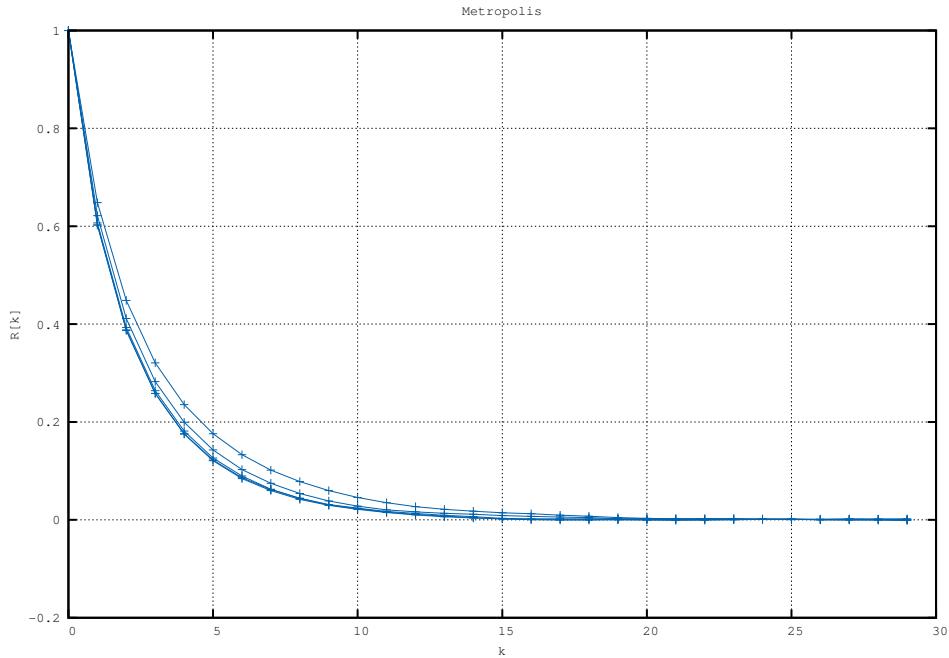


Figura 2.3: Autocorrelazione dell’osservabile $\langle x_l x_k \rangle$ per tutti i valori di $|l-k|$

Il tempo di decorrelazione τ_d è dell’ordine delle unità (~ 3) e quindi, per $\tau > 5\tau_d \sim 15$ il segnale può essere considerato praticamente scorrelato (fig.2.4).

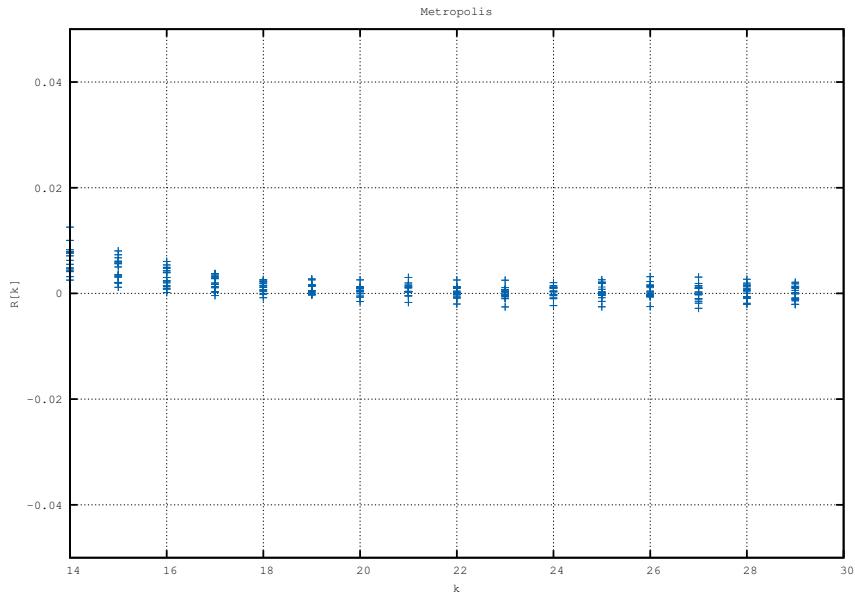


Figura 2.4: Autocorrelazione per $|l-k| > 15$.

2.1.3 Funzioni di Correlazione

Si calcola numericamente l'integrale $\int Dx x_l x_k e^{-S_E}$ applicando i metodi Monte Carlo e Metropolis in congiunzione. Affinchè le configurazioni utilizzate siano statisticamente indipendenti, si raggruppano i sample in *bin* e si utilizza l'insieme delle medie su ogni *bin* come sample di configurazioni per il calcolo MC. In questo modo, se la dimensione dei *bin* è molto maggiore di τ_d , la media sul *bin* attenua considerevolmente gli effetti di correlazione e perciò il nuovo sample può essere considerato statisticamente unbiased.

La media di un generico osservabile O calcolata in questo modo è analiticamente uguale a quella calcolata per mezzo del sample completo di ogni elemento estratto. La varianza sui valori medi, tuttavia, differisce dalla varianza semplice di un termine proporzionale alla covarianza fra le variabili.

Δt	$C(\Delta t)$	$Var(C(\Delta t))$	Δt	$C(\Delta t)$	$Var(C(\Delta t))$
0	0.480083	0.001332	16	-0.000183	0.000845
1	0.178045	0.000712	17	-0.000091	0.000644
2	0.065869	0.000509	18	-0.000131	0.000495
3	0.024213	0.000448	19	-0.000041	0.000443
4	0.008814	0.000431	20	0.000096	0.000433
5	0.003284	0.000431	21	0.000157	0.000428
6	0.001328	0.000432	22	0.000073	0.000427
7	0.000660	0.000439	23	-0.000037	0.000428
8	0.000205	0.000433	24	0.000205	0.000433
9	-0.000037	0.000428	25	0.000660	0.000439
10	0.000073	0.000427	26	0.001328	0.000432
11	0.000157	0.000428	27	0.003284	0.000431
12	0.000096	0.000433	28	0.008814	0.000431
13	-0.000041	0.000443	29	0.024213	0.000448
14	-0.000131	0.000495	30	0.065869	0.000509
15	-0.000091	0.000644	31	0.178045	0.000712

Figura 2.5: Valori del correlatore delle coordinate in funzione della loro separazione nel reticolo temporale.

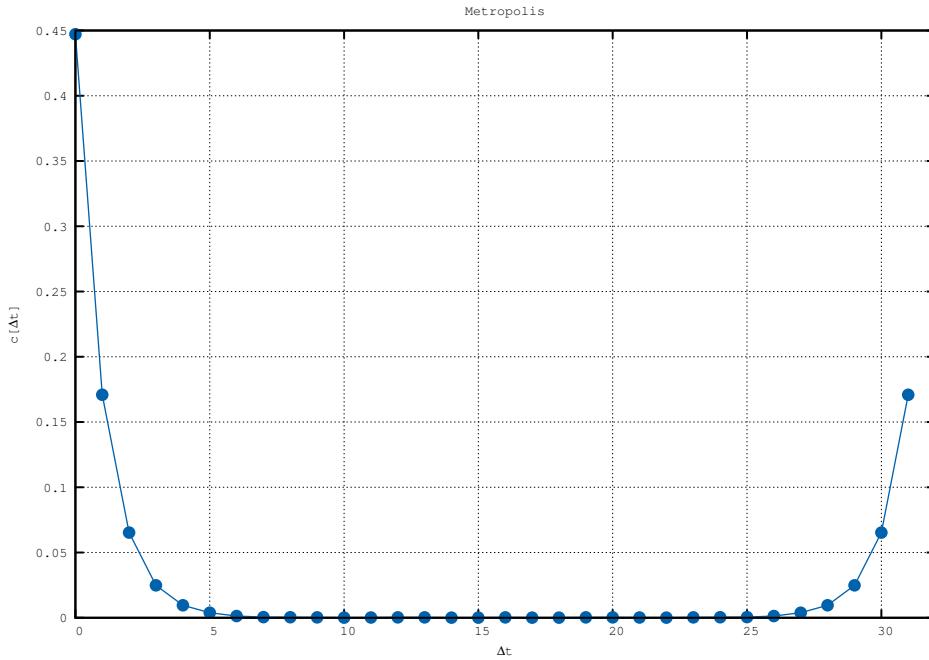


Figura 2.6: L’andamento del correlatore delle coordinate è quello di un coseno iperbolico centrato in $|l - k| = \frac{N}{2}$.

Sebbene il correlatore calcolato abbia un andamento complessivo in accordo con quello previsto dalla teoria, per valori di Δt compresi fra 8 e 24 il segnale viene perso e prevale il rumore.

2.1.4 Calcolo di Grandezze Quanto-Meccaniche Secondarie

Come anticipato nell’introduzione sull’oscillatore armonico, è ora possibile ricavare stime di grandezze quanto-meccaniche secondarie direttamente dai valori delle funzioni di correlazione $C(\Delta t) = \langle x_t x_{t+\Delta t} \rangle$ del sistema. Le varianze sui valori medi sono calcolate attraverso la tecnica *cluster jackknife*.

Sia o_i la stima semplice dell’osservabile O allo step i . La tecnica *cluster jackknife* consiste nel ricavare le grandezze derivate o^i (*clusterization*), definite come

$$o^i = \langle o \rangle + \frac{\langle o \rangle - o_i}{N_{bin} - 1}$$

dove N_{bin} indica il numero totale di *bin*. Nel caso in esame la grandezza O è il correlatore calcolato nella sezione precedente ed è clusterizzato con *bin* di ampiezza 100 ($\gg \tau_d$). Data una funzione $f = f(o)$, si definiscono $f^i = f(o^i)$ e $\langle f \rangle = f(\langle o \rangle)$. Si ottiene allora

$$\sigma_f^2 = \frac{N_{bin} - 1}{N_{bin}} \sum_i (f^i - \langle f \rangle)^2$$

Si ricavano dunque le formule per l'energia ΔE e l'elemento di matrice W . Per comodità si pone $\lambda \equiv N/2 - \Delta t$ e si utilizza il seguente trucco per ricavare ΔE dalla formula per il correlatore C :

$$\begin{aligned} \frac{C(\Delta t - 1) + C(\Delta t + 1)}{2C(\Delta t)} &= \\ &= \frac{\frac{1}{2}(e^{a\Delta E(\lambda+1)} + e^{-a\Delta E(\lambda+1)}) + \frac{1}{2}(e^{a\Delta E(\lambda-1)} + e^{-a\Delta E(\lambda-1)})}{2\frac{1}{2}(e^{a\Delta E\lambda} + e^{-a\Delta E\lambda})} = \\ &= \frac{(e^{a\Delta E} + e^{-a\Delta E})(e^{a\Delta E\lambda} + e^{-a\Delta E\lambda})}{2(e^{a\Delta E\lambda} + e^{-a\Delta E\lambda})} = \cosh(a\Delta E) \\ \Rightarrow a\Delta E &= \cosh^{-1}\left(\frac{C(\Delta t - 1) + C(\Delta t + 1)}{2C(\Delta t)}\right) \end{aligned}$$

Trascurando il fattore a e utilizzando i valori medi di C calcolati su i *cluster* si ottiene:

$$\Delta E = \cosh^{-1}\left(\frac{\langle C(\Delta t + 1) \rangle + \langle C(\Delta t - 1) \rangle}{2\langle C(\Delta t) \rangle}\right)$$

$$W = \sqrt{\langle C(\Delta t) \rangle \frac{\exp(\Delta E N/2)}{2 \cosh(\Delta E(N/2 - \Delta t))}}$$

La migliore stima di ΔE e W viene quindi calcolata come media sui valori dati dalle precedenti formule utilizzando i valori del correlatore per $0 \leq \Delta t \leq 6$.

I valori utilizzati per i parametri del sistema sono $N = 32$, $\hbar = 1$, $a = 1$, $\omega = 1$, $m = 1$, $\delta = 3$ con un numero di sweep pari a 10^6 . I valori ottenuti sono distribuiti gaussianamente attorno ai valori medi che possono essere quindi ricavati dal fit di gaussiane agli istogrammi dei sample:

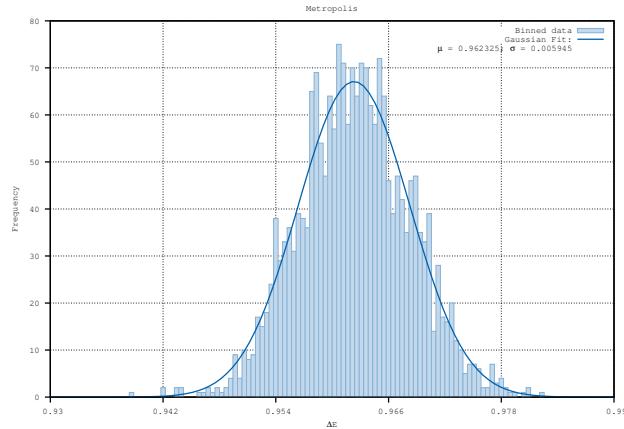


Figura 2.7: Distribuzione di ΔE a fissato numero di sweeps

Risultati dei fit:

$$\begin{aligned}\Delta E &= 0.962325 \pm 0.005945 \\ W &= 0.667925 \pm 0.007161\end{aligned}$$

Si nota inoltre che anche l'errore stimato dal *cluster jackknife* è distribuito normalmente intorno al valore dell'errore statistico ricavato dal fit.

Si studia ora l'andamento della varianza $\sigma_{jackknife}^2$ in funzione del numero di *sweeps*.

N_{sweeps}	$\sigma_{jackknife}^2$
1000	2.087426e-02
2000	1.201255e-02
4000	6.999672e-03
8000	5.147464e-03
16000	2.081578e-03
32000	9.692904e-04
64000	5.851014e-04
128000	2.368376e-04
256000	1.377607e-04
512000	6.941350e-05
1024000	3.300726e-05
2048000	1.669044e-05
4096000	8.221103e-06
8192000	4.123789e-06
16384000	2.054391e-06
32768000	1.029125e-06
65392000	5.224967e-07

Rappresentando i dati ricavati dalle simulazioni in un grafico in scala bilogaritmica appare evidente che la varianza è inversamente proporzionale al numero di *sweeps*.

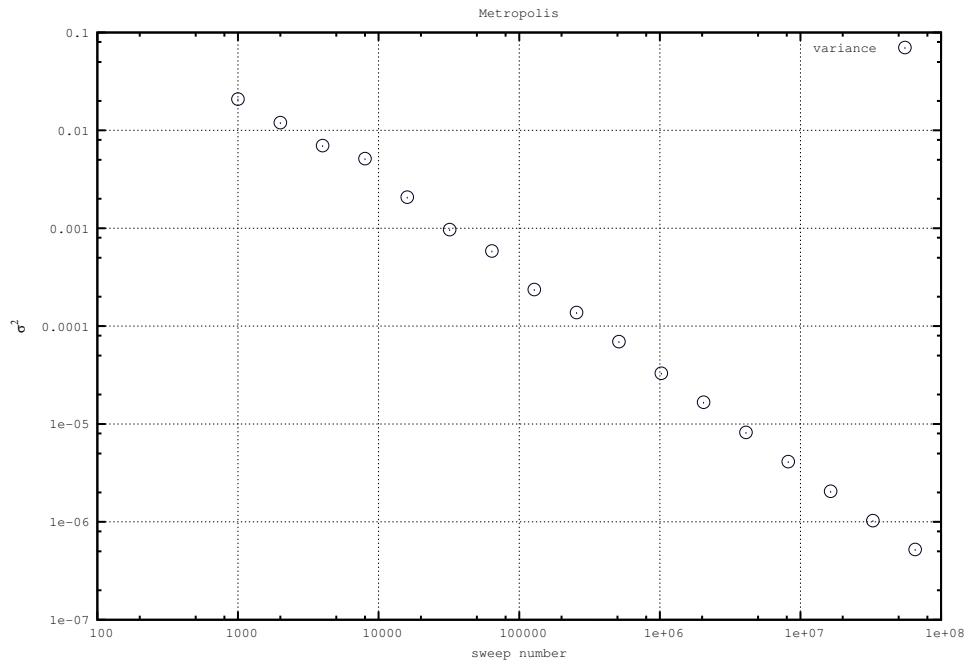


Figura 2.8: Varianza di ΔE in funzione del numero di sweeps

Interpolando linearmente i dati in scala bilogaritica si ottiene infatti un coefficiente angolare $m = -0.976 \sim -1$. Se ne deduce quindi che:

$$\sigma \sim 1/\sqrt{N_{\text{sweeps}}}$$

in accordo con la teoria degli errori per i metodi della famiglia dei Monte Carlo.

Capitolo 3

Metodo Runge-Kutta

In analisi numerica i metodi Runge-Kutta sono una famiglia di metodi iterativi impliciti ed esplicativi per la risoluzione approssimata di equazioni differenziali ordinarie (ODE). Il più comune di questi metodi è il cosiddetto “RK4” o anche Runge-Kutta del quarto ordine.

Sia dato il problema di Cauchy

$$\dot{y} = f(t, y), \quad y(t_0) = y_0$$

Si assume che il tempo sia discretizzato in istanti t_n equidistanziati di un intervallo h .

Il metodo RK4 per questo problema è allora dato dalle seguenti equazioni:

$$\begin{aligned} y_{n+1} &= y_n + \frac{1}{6} (k_1 + 2k_2 + 2k_3 + k_4) \\ t_{n+1} &= t_n + h \end{aligned}$$

dove y_{n+1} è l'approssimazione RK4 di $y(t_{n+1})$, e

$$\begin{aligned} k_1 &= hf(t_n, y_n), \\ k_2 &= hf(t_n + \frac{1}{2}h, y_n + \frac{1}{2}k_1), \\ k_3 &= hf(t_n + \frac{1}{2}h, y_n + \frac{1}{2}k_2), \\ k_4 &= hf(t_n + h, y_n + k_3). \end{aligned}$$

Il valore della funzione y all'istante t_{n+1} è uguale quindi al suo valore all'istante t_n incrementato della media ponderata di quattro incrementi k , dove ogni incremento è il prodotto della dimensione dell'intervallo h ed una stimata pendenza specificata dalla funzione f .

- k_1 è l'incremento basato sulla pendenza di f all'estremo sinistro dell'intervallo, calcolato in y_n (metodo di Eulero);
- k_2 è l'incremento basato sulla pendenza nel punto medio dell'intervallo, calcolato in $y_n + \frac{1}{2}k_1$;

- k_3 è ancora l'incremento basato sulla pendenza nel punto medio, calcolato però in $y_n + \frac{1}{2}k_2$;
- k_4 è l'incremento basato sulla pendenza all'estremo destro dell'intervallo, calcolato in $y_n + k_3$.

Si nota dalla formula per y_{n+1} che peso maggiore viene assegnato all'incremento al centro dell'intervallo. Inoltre, se $f = f(t)$, cioè non dipende da y , il metodo RK4 si riduce alla regola di integrazione di Simpson.

RK4 è un metodo del quarto ordine e quindi l'errore ad ogni step è dell'ordine di h^5 , mentre l'errore totale accumulato è dell'ordine di h^4 .

3.1 Simulazione di Sistemi Dinamici

Per mezzo del metodo RK4 è possibile simulare l'evoluzione di sistemi dinamici, in particolare di quelli non lineari o caotici, le cui soluzioni non sono note analiticamente.

Esempi di sistemi dinamici che presentano comportamenti non lineari sono i sistemi di Lorenz, Rossler, Chua e di Rabinovich-Fabrikant.

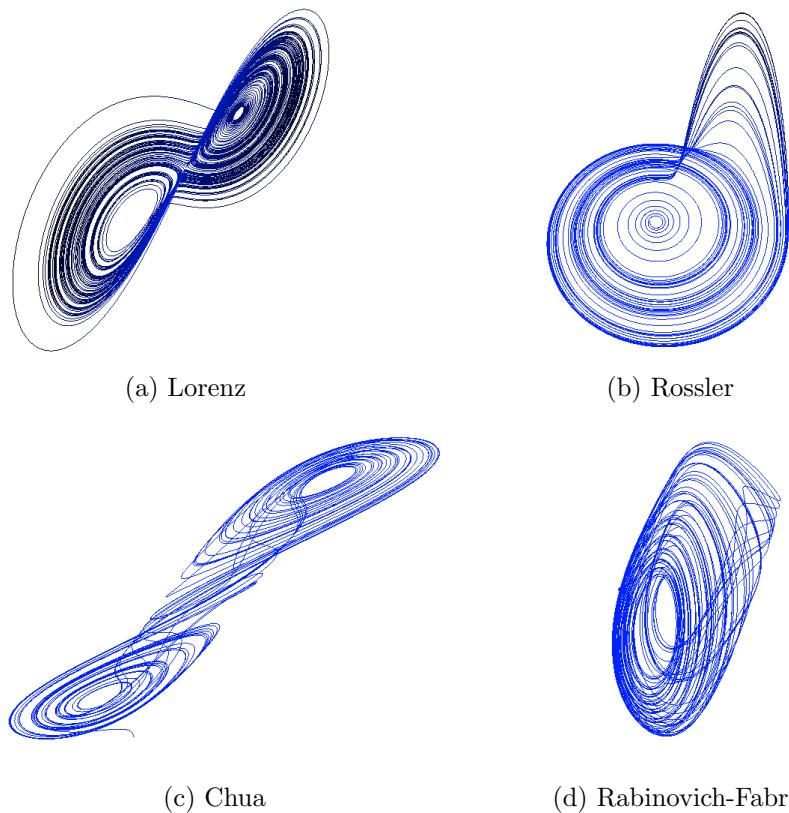


Figura 3.1: Soluzioni numeriche di sistemi dinamici non lineari

La precisione del metodo RK4 decresce considerevolmente con all'aumentare del numero di step della simulazione. L'errore teorico stimato infatti si accumula ad ogni step incrementando la discrepanza con la soluzione esatta. Ciò risulta evidente dal risultato della simulazione rappresentata in figura 3.2.

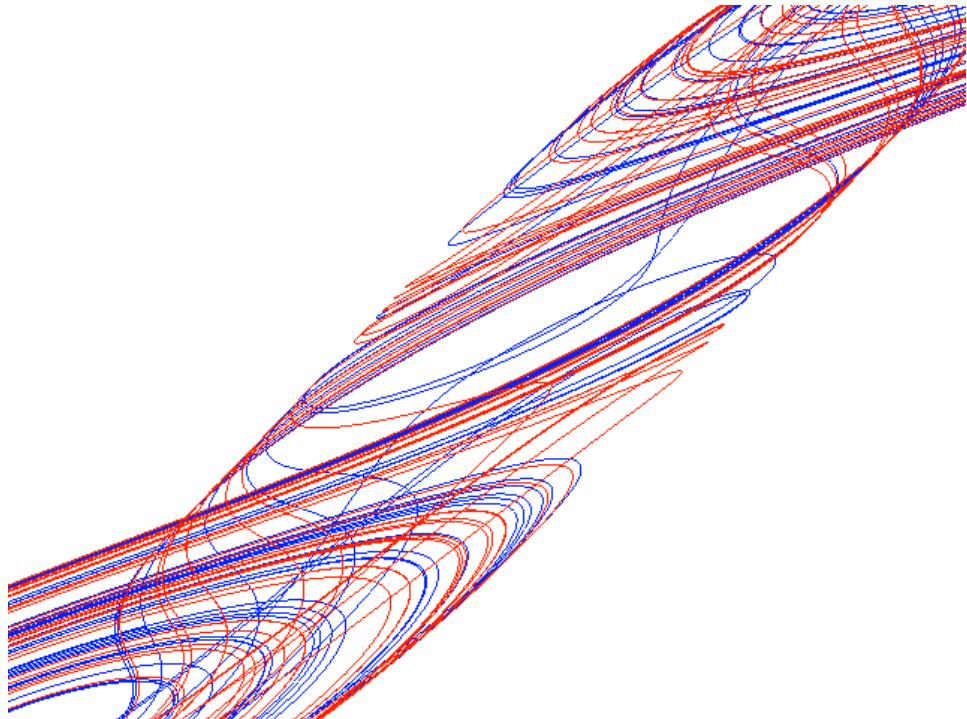


Figura 3.2: Due simulazioni del sistema di Chua con uguali condizioni iniziali e differenti h . La linea rossa rappresenta una soluzione numerica calcolata con h doppio rispetto a quello della soluzione in blu.

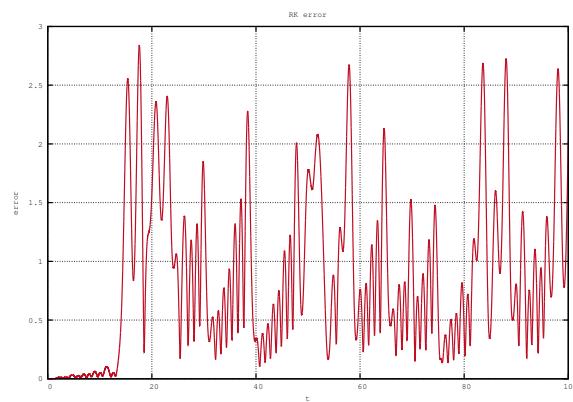


Figura 3.3: Discrepanza fra le soluzioni in funzione del tempo di simulazione

3.1.1 Pendolo Caotico

Si trovano ora soluzioni numeriche di un sistema non lineare e se ne studia il comportamento al variare dei suoi parametri.

L'equazione del moto di un pendolo non lineare forzato e caotico è:

$$\ddot{\theta}(t) = -\sin(\theta(t)) - q\dot{\theta}(t) + b \cos(\omega t)$$

È possibile ricondurre l'equazione precedente ad un sistema di equazioni differenziali del primo ordine effettuando la sostituzione $x = \theta$, $y = \dot{\theta}$:

$$\Rightarrow \begin{cases} \dot{x} = y \\ \dot{y} = -\sin(x) - qy + b \cos(\omega t) \end{cases}$$

Applicando ora il metodo RK4 si ottengono le seguenti formule per l'evoluzione delle variabili dinamiche x, y

$$x_{n+1} = x_n + hy_n$$

$$y_{n+1} = y_n + \frac{1}{6}k_{y,1} + \frac{1}{3}k_{y,2} + \frac{1}{3}k_{y,3} + \frac{1}{6}k_{y,4}$$

dove i $k_{y,i}$ sono i coefficienti definiti nel paragrafo iniziale e relativi alla seconda equazione del sistema. È facile verificare, invece, che i coefficienti $k_{x,i}$ sono banalmente tutti uguali a hy_n , dal momento che $\frac{\partial \dot{x}}{\partial x} = 0$.

Di seguito sono raffigurate le traiettorie nello spazio delle fasi (x, y) dell'oscillatore di alcune soluzioni, con uguali condizioni iniziali e $h = 10^{-3}$.

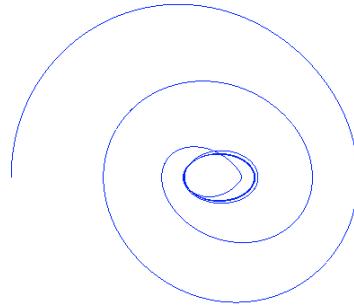
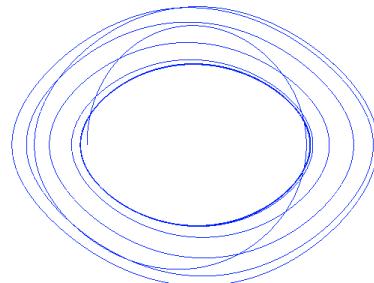
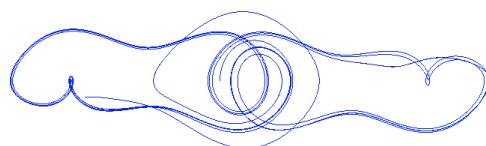
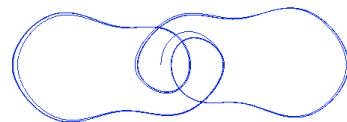


Figura 3.4: Soluzione per $q = 0.3$, $b = 0.1$, $\omega = 2/3$

Figura 3.5: Soluzione per $q = 0.3, b = 0.5, \omega = 2/3$ Figura 3.6: Soluzione per $q = 0.3, b = 0.9, \omega = 2/3$ Figura 3.7: Soluzione per $q = 0.3, b = 1.4, \omega = 2/3$

Quello che si nota è che il sistema presenta in tutti e quattro i casi un ciclo limite. Aumentando di poco l'intensità della forzante aumentano anche le dimensioni di tale curva. Quando però b diviene troppo grande, come nel terzo e quarto caso (fig.3.6, 3.7), il ciclo limite assume una topologia totalmente differente.

Capitolo 4

Metodo Implicito

Per sistemi di equazioni alle derivate parziali (PDE) esistono altri metodi oltre a quelli RK. In questa sezione si illustra e implementa il cosiddetto metodo implicito per risolvere equazioni agli operatori lineari.

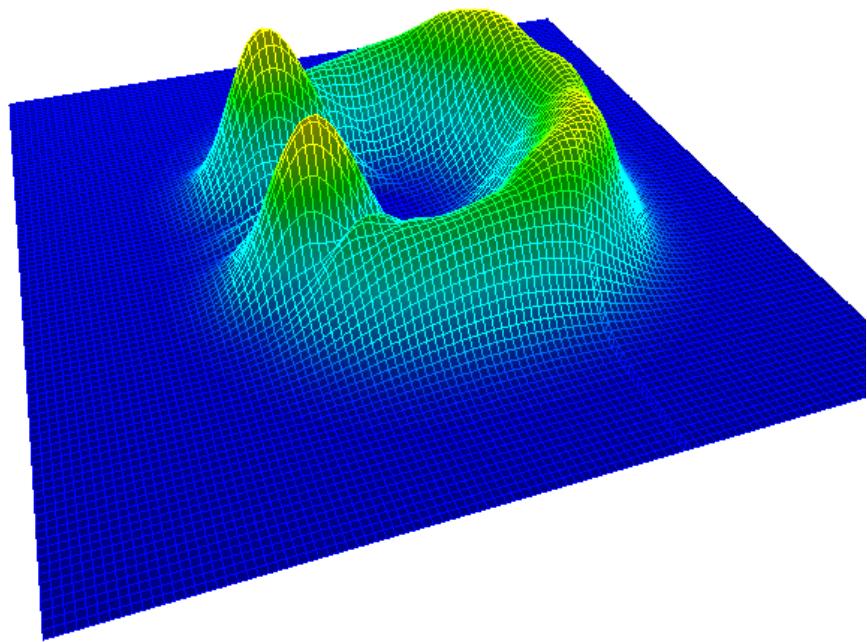


Figura 4.1: Scattering di un pacchetto gaussiano con una buca di potenziale. L'onda, arrivando da destra, scatterà contro la buca di potenziale posta al centro del reticolo senza riuscire ad attraversarla.

4.1 Equazione di Schrödinger

L'equazione che si intende risolvere è l'equazione di Schrödinger dipendente dal tempo

$$i\hbar \frac{\partial}{\partial t} \psi(x, t) = -\frac{\hbar^2}{2m} \nabla^2 \psi(x, t) + V(x) \psi(x, t)$$

A tal fine si discretizza lo spazio in modo tale che $x = na$, con $n = 0, \dots, N-1$, ed a passo del reticolo spaziale. $\psi(x, t)$ diventa quindi un vettore di valori complessi $\psi_n(t)$. Si pongono inoltre condizioni di periodicità al contorno di modo che $\psi_N \equiv \psi_0$.

Con questa dicretizzazione si sostituisce

$$\nabla^2 \psi(x) \rightarrow \frac{\psi_{n+1} + \psi_{n-1} - 2\psi_n}{a^2}$$

pertanto l'equazione diventa

$$i\hbar \frac{\partial}{\partial t} \psi_n = -\frac{\hbar^2}{2m} \frac{\psi_{n+1} + \psi_{n-1} - 2\psi_n}{a^2} + V_n \psi_n = \sum_n A_{n,m} \psi_m$$

La soluzione dell'equazione è la funzione esponenziale

$$\psi(t) = \exp\left(-\frac{i}{\hbar} At\right) \psi(0)$$

dove $U(t) = \exp(-\frac{i}{\hbar} At)$ rappresenta l'evolutore temporale del sistema. Sviluppando ora $U(t)$ intorno all'unità per $t \rightarrow 0$ e trascurando gli ordini superiori in t si ottiene la formula approssimata

$$\psi(t_0 + t) \simeq \psi(t_0) - \frac{i}{\hbar} At \psi(t_0)$$

Questa prima approssimazione tuttavia porta all'insorgere di numerosi problemi dovuti al fatto che l'iterazione descritta dal precedente passaggio non rispetta la condizione di unitarietà dell'operatore evolutore temporale. Bisogna quindi ricorrere ad una strategia più astuta.

Il **metodo implicito** [6] consiste nell'approssimare l'operatore esatto $U(t)$ attraverso la formula

$$\psi(t + dt) = (1 + \frac{i}{2\hbar} Adt)^{-1} (1 - \frac{i}{2\hbar} Adt) \psi(t)$$

Utilizzando il fatto che A è Hermitiano, è facile dimostrare che l'operatore della precedente formula è unitario proprio come $U(t)$. Per implementare questo algoritmo è necessario risolvere ad ogni step il sistema lineare

$$(1 + \frac{i}{2\hbar} Adt) \psi(t + dt) = (1 - \frac{i}{2\hbar} Adt) \psi(t)$$

Infine si moltiplicano entrambi i membri dell'equazione per l'aggiunto dell'operatore a primo membro, in modo tale da ottenere

$$(1 + \frac{1}{4\hbar^2} A^2 dt^2) \psi(t + dt) = (1 - \frac{i}{2\hbar} Adt)^2 \psi(t)$$

dove ora $(1 + \frac{1}{4\hbar^2} A^2 dt^2)$ è Hermitiano e si può quindi risolvere rispetto a $\psi(t + dt)$ utilizzando il *Conjugate Residual Method* [7].

L'idea che sta alla base di questo metodo è che se in un intervallo di tempo dt la funzione $\psi(t)$ evolve in $\psi(t + dt)$, allora facendo evolvere $\psi(t)$ in avanti nel tempo per $\frac{dt}{2}$ e $\psi(t + dt)$ indietro nel tempo sempre di $\frac{dt}{2}$ le due funzioni dovranno necessariamente coincidere.

Al fine di migliorare la precisione dell'approssimazione è anche possibile utilizzare sviluppi a ordini superiori in t al costo, tuttavia, di incrementare considerevolmente la complessità dell'algoritmo.

La formula per l'evolutore al secondo ordine è data da

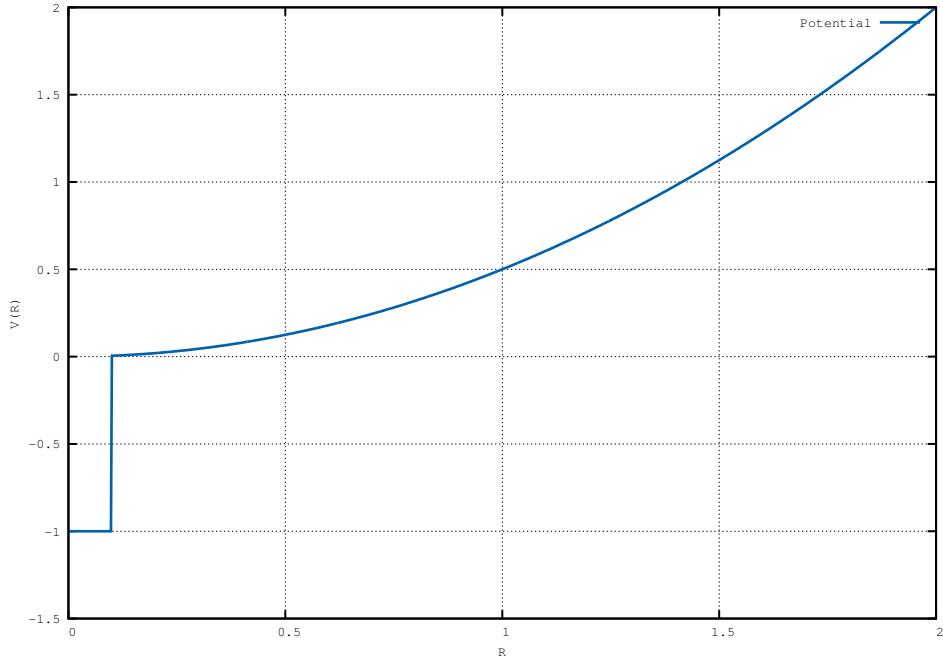
$$\psi(t + dt) = \left(1 + \frac{i}{2\hbar} Adt - \frac{1}{8\hbar^2} A^2 dt^2\right)^{-1} \left(1 - \frac{i}{2\hbar} Adt - \frac{1}{8\hbar^2} A^2 dt^2\right) \psi(t)$$

che svolgendo il procedimento descritto per lo sviluppo al primo ordine diventa

$$\left(1 + \frac{1}{64\hbar^4} A^4 dt^4\right) \psi(t+dt) = \left(1 - \frac{i}{2\hbar} Adt - \frac{3}{8\hbar^2} A^2 dt^2 + \frac{i}{16\hbar^3} A^3 dt^3 + \frac{1}{64\hbar^4} A^4 dt^4\right) \psi(t)$$

L'unitarietà dell'operatore è evidentemente preservata ad ogni ordine proprio grazie al modo in cui è stata costruita la formula per l'approssimazione dell'evolutore.

4.1.1 Scattering



Si utilizza un potenziale centrale con buca profonda $-V_0$, del tipo mostrato in figura. La funzione d'onda viene inizializzata ad una gaussiana bidimensionale stazionaria decentrata, posta in un reticolo 256×256 con condizioni periodiche al contorno.

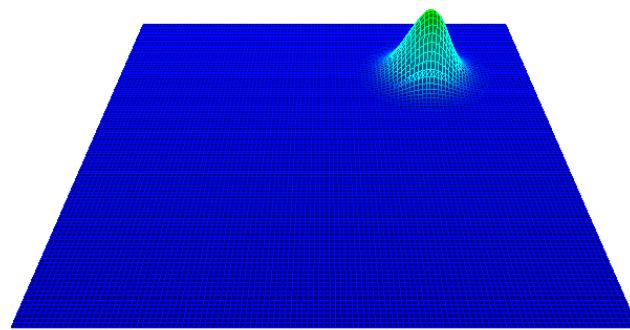


Figura 4.2: $\psi(t_0) = \mathcal{N}(\mu_x, \mu_y, \sigma^2)$

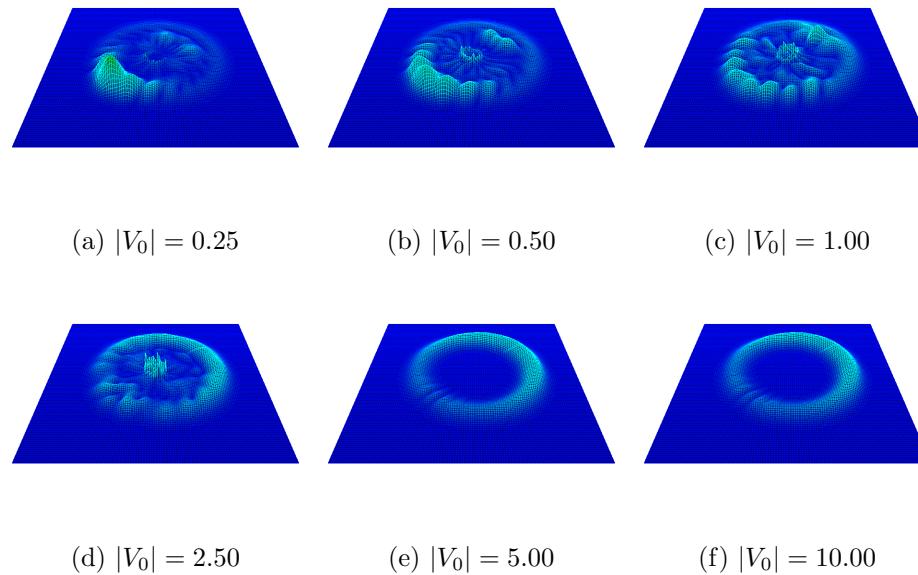


Figura 4.3: Soluzioni numeriche per differenti valori del potenziale V_0 . Tutti gli shot sono stati effettuati allo stesso tempo di esecuzione, cioè dopo che il pacchetto si è scontrato per la prima volta con la buca.

Per grandi valori di $|V_0|$ il pacchetto d'onda non riesce ad attraversare l'ostacolo e viene scatterato radialmente. Se inoltre le dimensioni della buca sono maggiori di quelle del pacchetto, quest'ultimo può addirittura venire completamente riflesso e distorto. Al diminuire di $|V_0|$ invece si può arrivare ad ottenere uno stato momentaneamente legato all'interno della buca, come in figura 4.4.

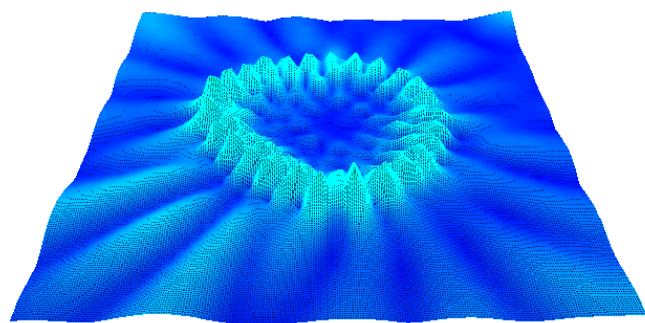


Figura 4.4: Dettaglio dello stato legato.

Come si nota in fig.4.5 la differenza fra simulazioni con diversi dt è minima. Per tempi di simulazione non troppo grandi, all'aumentare del parametro dt corrisponde una variazione equivalente ad un piccolo sfasamento temporale nelle soluzioni calcolate.

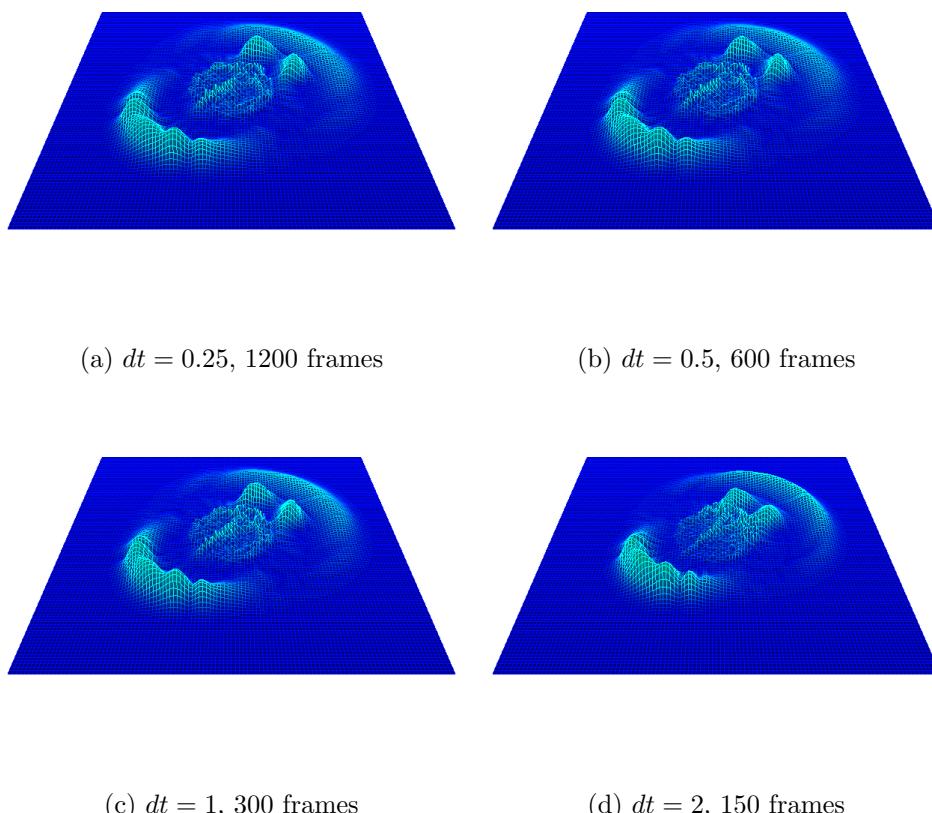


Figura 4.5: Soluzioni numeriche per differenti valori del parametro dt

Bibliografia

- [1] M. Hjorth-Jensen. “§9, random walks and the metropolis algorithm”, *Computational Physics*, 2009.
- [2] W. Krauth. *Introduction To Monte Carlo Algorithms*, 2004.
- [3] I. A. Stegun M. Abramowitz. “§25.4, integration”, *Handbook of Mathematical Functions*, 1972.
- [4] B. Freedman M. Creutz. *A Statistical Approach to Quantum Mechanics*, 1980.
- [5] C. Morningstar. *The Monte Carlo Method in Quantum Mechanics*, 2004.
- [6] C. Rebbi. *Lecture Notes on Advanced Computing in Physics*, 1996.
- [7] Y. Saad. “§6.8, the conjugate residual method”, *Iterative Methods for Sparse Linear Systems*, 2000.
- [8] S. Weinzierl. *Introduction to Monte Carlo Methods*, 2000.