

Relazioni di Laboratorio di Fisica Computazionale

Luca Cassia - [MAT. 728341]

Spring 2012

Indice

1	Integrazione Numerica	2
1.1	Quadrature Gaussian	2
1.2	Monte Carlo	2
1.3	Esempi Numerici	2
1.4	Differenziazione Numerica	3
2	Algoritmo Metropolis	4
2.1	L'Oscillatore Armonico Quantistico	4
2.2	Azione e Termalizzazione	4
2.3	Autocorrelazione	7
2.4	Correlazione	7
2.5	Calcolo di ΔE	7
2.6	Calcolo dell'Elemento di Matrice	7
3	Metodo Runge-Kutta per Equazioni Differenziali Ordinarie	10
3.1	Il Pendolo Caotico	11
3.2	Oscillatore di Van Der Pol	11
3.3	Attrattore di Lorenz	13
4	Metodo implicito per PDE	14
4.1	Equazione di Schrödinger	14

Capitolo 1

Integrazione Numerica

1.1 Quadrature Gaussiane

1.2 Monte Carlo

1.3 Esempi

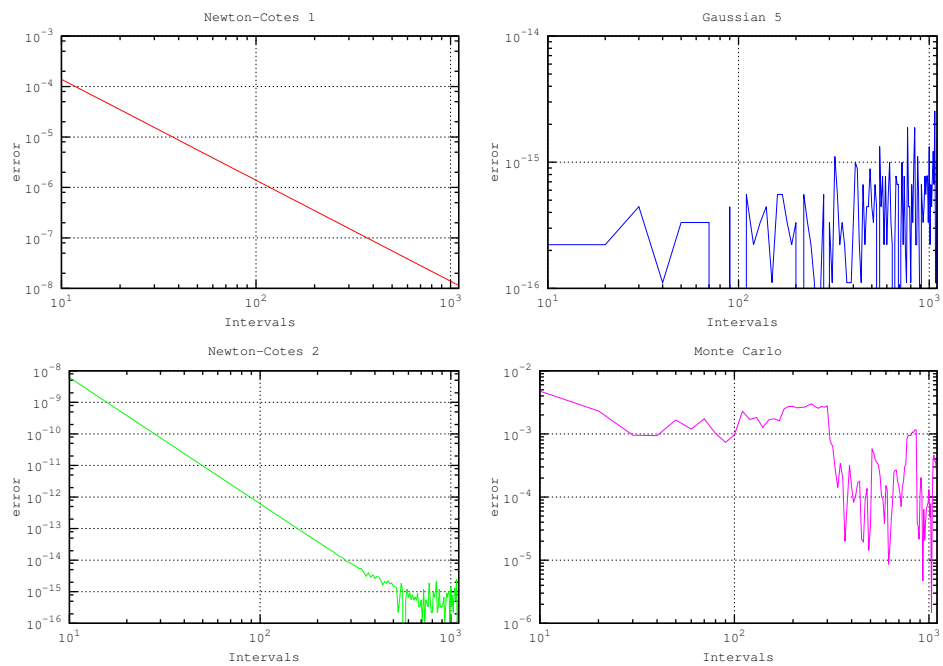


Figura 1.1: Errore calcolato

1.4 Differenziazione Numerica

Capitolo 2

Algoritmo Metropolis

Il Metropolis è l'algoritmo più influenziale fra quelli appartenenti alla classe dei metodi Monte Carlo. Supportato da una profonda teoria, 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.

2.1 L'Oscillatore Armonico Quantistico

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. La particella inoltre interagisce con un potenziale armonico della forma $V(x) = \frac{m}{2}\omega^2 x^2$.

La relazione che si vuole verificare è quella per il correlatore delle variabili l -esima e k -esima

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

Nel caso in esame l'insieme delle variabili x_t del sistema viene rappresentato da un array monodimensionale `x[i]` di `double`, dove l'indice `i` indica la posizione nel reticolo temporale, e dove si è posto $x_N \equiv x_0$.

2.2 Azione e Termalizzazione

Inizializzando il vettore della configurazione con numeri casuali è necessario lasciare del tempo al sistema per portarsi allo stato di equilibrio, dove, per l'ipotesi di ergodicità, tutte le configurazioni possibili sono equiprobabili. Tale processo, denominato termalizzazione, richiede di norma non più di 500 cicli di metropolis, eseguiti i quali sarà possibile estrarre le configurazioni con la giusta distribuzione di probabilità. Vi sono due modi di porre le condizioni iniziali:

- Azione “fredda” : le variabili sono inizializzate a zero.
- Azione “calda” : le variabili sono inizializzate a valori diversi da zero.

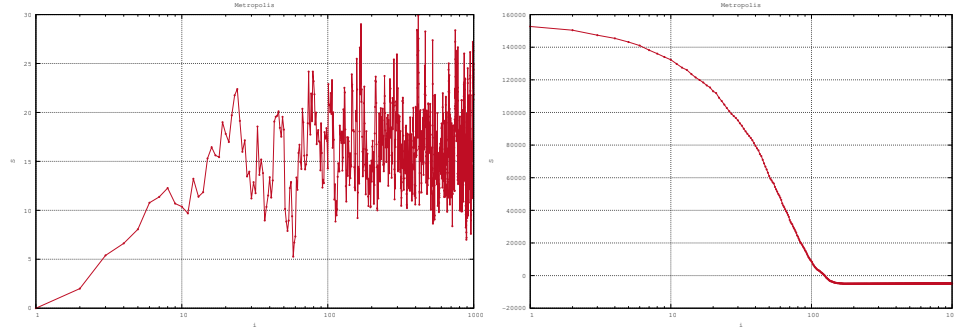


Figura 2.1: Azione Fredda e Calda

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

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

Per calcolare il $\Delta S = S' - S$ 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

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

Il codice corrispondente è

```

9 #define N 32
10
11 double M = 1;
12 double W = 1;
13 double D = 3;
14
15 double V(double x){
16     return M*W*W*x*x/2;
17 }
18
19 double dS(double *x, double y, int i){
20     return M*((x[i]-y)*(x[(i+1)%N]+x[(i-1+N)%N])+(y*y-x[i]*x[i]
        ))+V(y)-V(x[i]);

```

```

21 }
22
23 double action(double *x){
24     double s = 0;
25     int i; for(i = 0; i < N-1; i++)
26         s += M*(x[i+1]-x[i])*(x[i+1]-x[i])/2+(V(x[i])+V(x[i+1]))/2;
27     return s;
28 }

```

../main/metropolis.c

dove le variabili M e W corrispondono alla massa e alla pulsazione riscalate del fattore a mentre la y argomento della funzione $dS()$ corrisponde alla nuova variabile estratta x'_i . Si noti che le condizioni di periodicità al contorno del reticolo sono implementate attraverso l'algebra modulo N sugli indici del vettore x .

Il ciclo di termalizzazione viene eseguito all'interno del `main()`:

```

84     double S = action(x);
85     FILE* f = fopen("action.dat","w");
86     for(i = 0; i < 1000; i++)
87         fprintf(f,"%d\t%f\n", (i+1), S += metropolis(x));
88     fclose(f);
89     plot_action();

```

../main/metropolis.c

La funzione `metropolis()` esegue uno sweep sul vettore x delle variabili di sistema aggiornandole una alla volta con probabilità $\min(1, e^{-\Delta S_i})$, e restituendo infine la variazione di azione totale $\Delta S = \sum ds = \sum_{i=0}^{N-1} \Delta S_i$.

```

30 double metropolis(double *x){
31     double DS = 0, y, ds, r[2*N];
32     ranlxd(r, 2*N);
33     int i; for(i = 0; i < N; i++){
34         y = x[i]+D*(2*r[i]-1);
35         ds = dS(x,y,i);
36         if(r[N+i] < exp(-ds)){
37             x[i] = y;
38             DS += ds;
39         }
40     }
41     return DS;
42 }

```

../main/metropolis.c

La variabile x'_i è estratta con una distribuzione piatta in un'intorno sferico di x_i di raggio D .

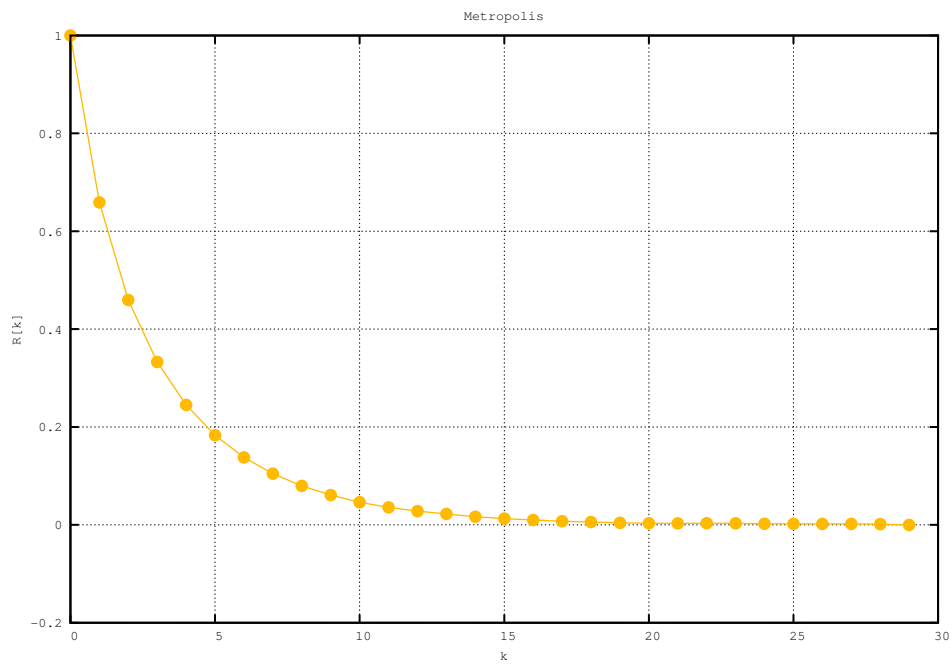


Figura 2.2: Autocorrelazione

2.3 Autocorrelazione

2.4 Correlazione

2.5 Calcolo di ΔE

2.6 Calcolo dell'Elemento di Matrice

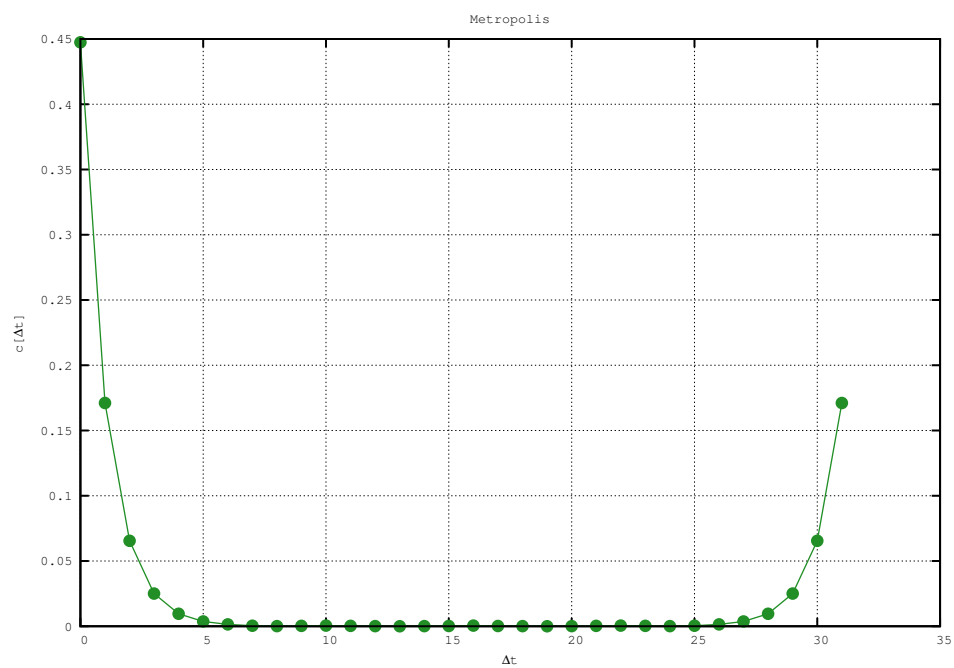


Figura 2.3: Correlazione

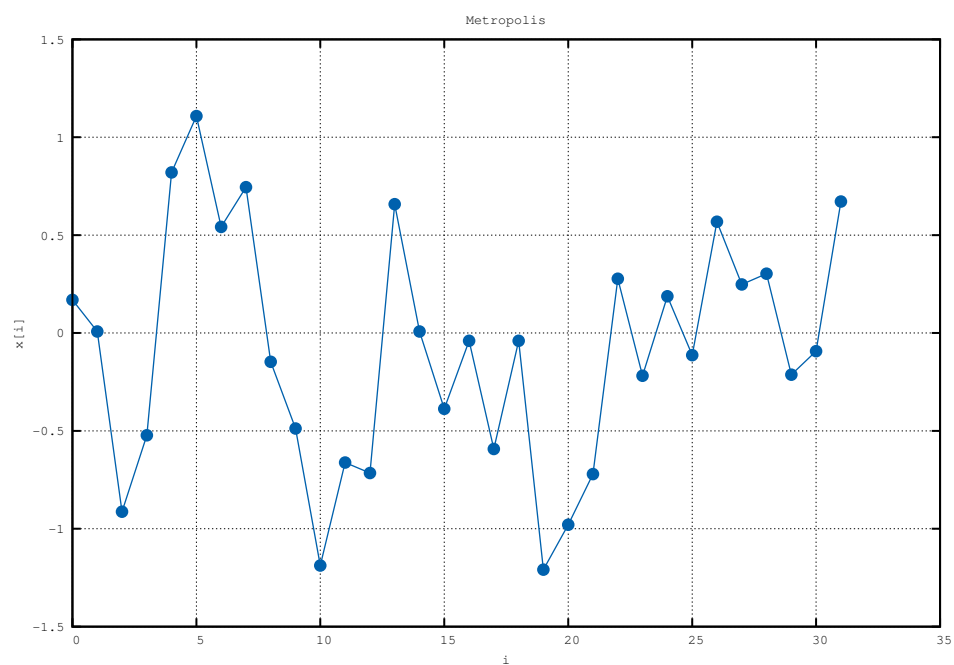


Figura 2.4: Configurazione

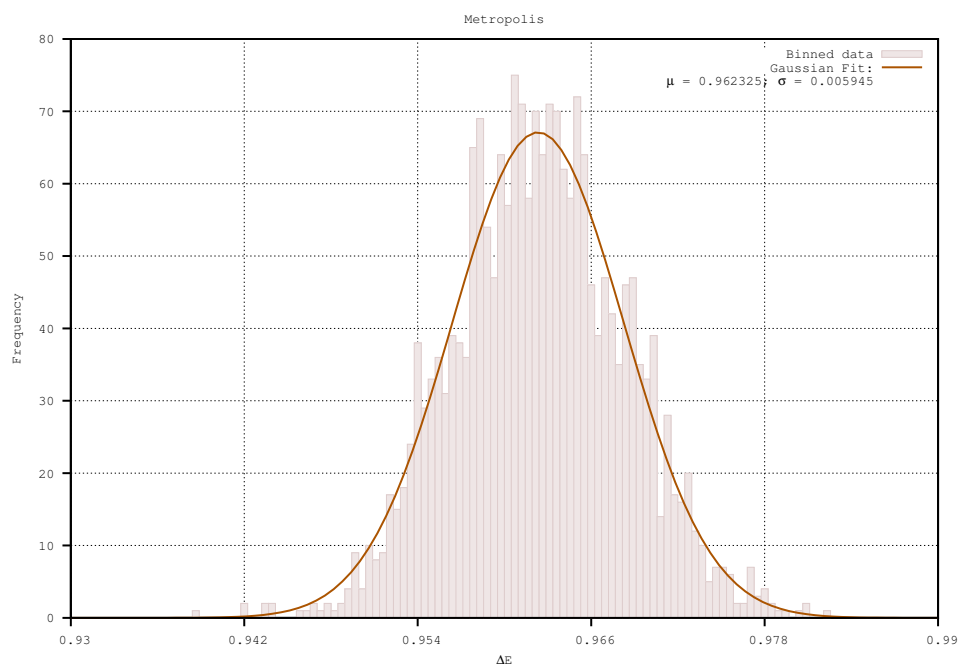
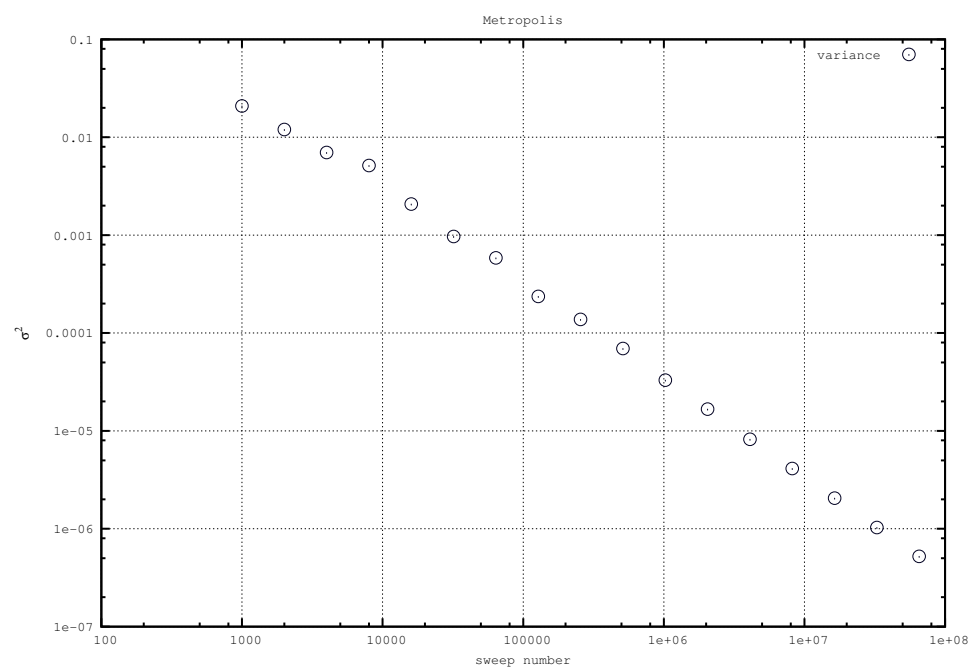


Figura 2.5: Istogramma

Figura 2.6: Varianza di ΔE

Capitolo 3

Metodo Runge-Kutta per Equazioni Differenziali Ordinarie

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

Sia dato il problema di Cauchy

$$y' = f(t, y), \quad y(t_0) = y_0.$$

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 Il Pendolo Caotico

Il sistema che si intende simulare è quello dell'oscillatore caotico forzato di equazione

$$\ddot{\theta} = \underbrace{\frac{g}{R}}_1 \sin(\theta) - \underbrace{\frac{k}{m}}_q \dot{\theta} + \underbrace{\frac{f}{mR}}_b \cos(\omega t).$$

È possibile rielaborare l'equazione precedente in un sistema di equazioni differenziali del primo ordine come di seguito

$$\begin{cases} \dot{\theta} = \phi \\ \dot{\phi} = \sin(\theta) - q\phi + b \cos(\omega t) \end{cases}$$

ed applicare ora la formula del metodo RK4 ad entrambe le equazioni.

Rappresentando la traiettoria del sistema nello spazio delle fasi si ottiene

3.2 Oscillatore di Van Der Pol

Analogamente al caso dell'oscillatore caotico, si applica il metodo RK4 all'equazione del pendolo di Van Der Pol, che in questo caso è

$$\frac{d^2x}{dt^2} - \mu(1 - x^2)\frac{dx}{dt} + x = 0$$

dove μ rappresenta l'intensità dello smorzamento. Il sistema associato è

$$\begin{cases} \dot{x} = y \\ \dot{y} = \mu(1 - x^2)\frac{dx}{dt} - x \end{cases}$$

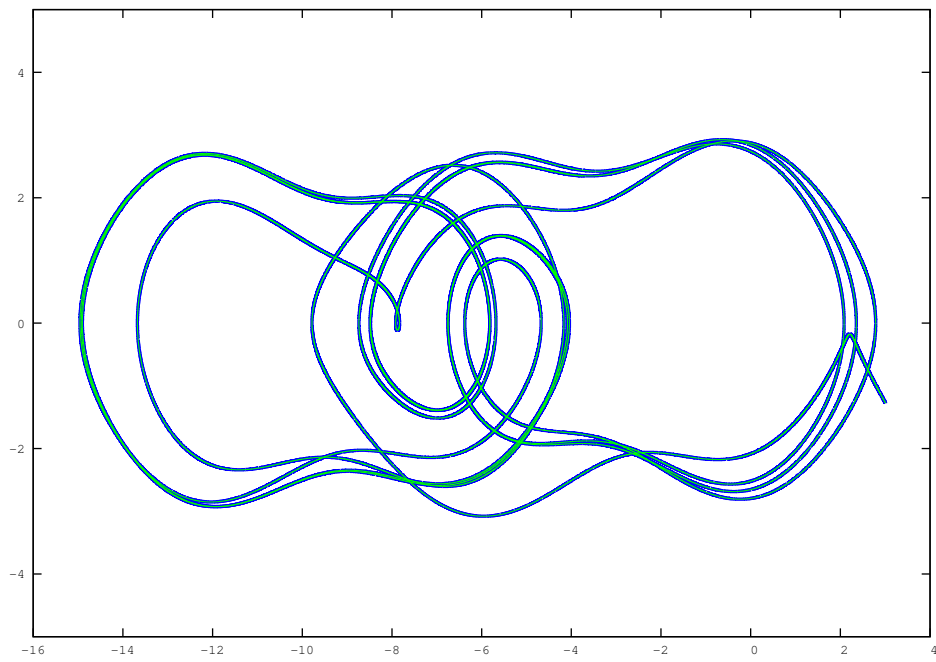


Figura 3.1: Traiettorie dell'oscillatore per $q = 0.3$, $b = 1.4$, $\omega = 0.6667$

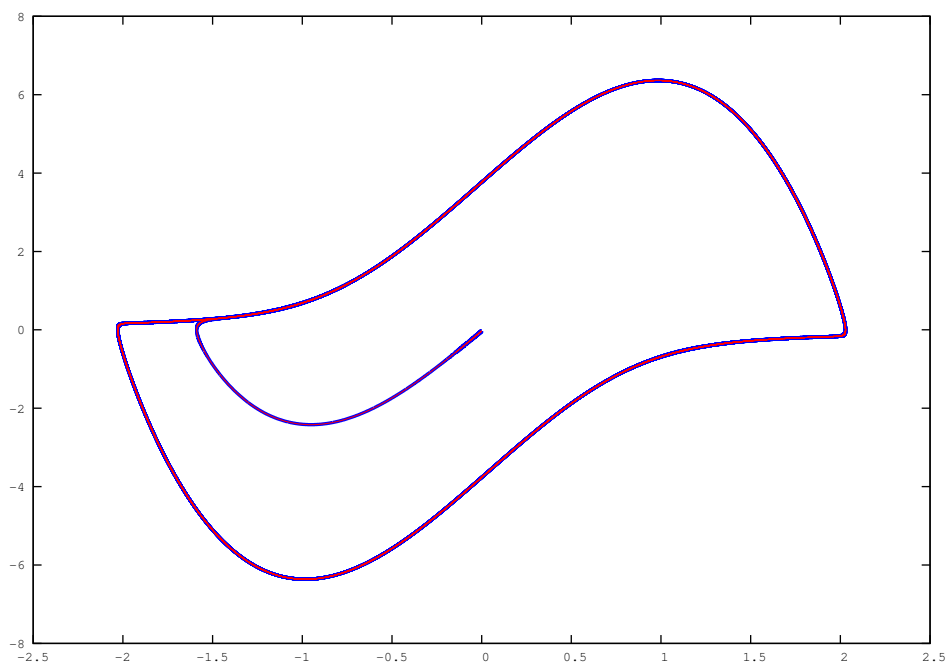


Figura 3.2: Traiettorie dell'oscillatore per $\mu = 4$

Anche in questo caso si può osservare un ciclo limite nello spazio delle fasi del sistema, ovvero una traiettoria chiusa verso la quale il sistema è portato ad evolvere.

3.3 Attrattore di Lorenz

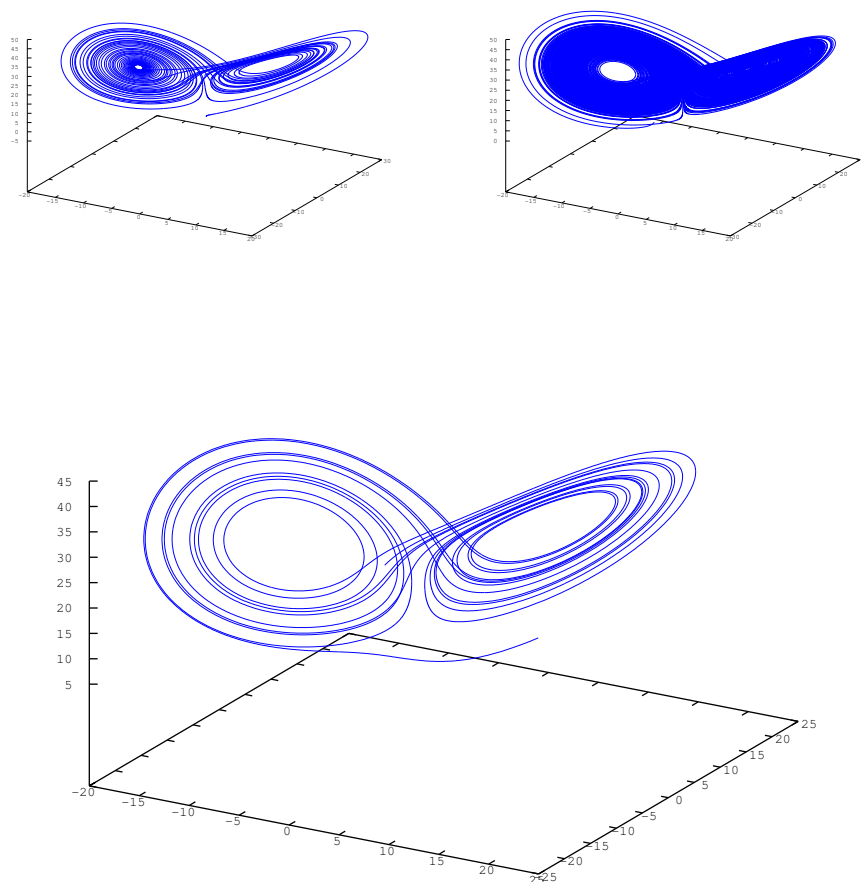


Figura 3.3: Sistema di Lorenz in tre dimensioni

Capitolo 4

Metodo implicito per PDE

4.1 Equazione di Schrödinger

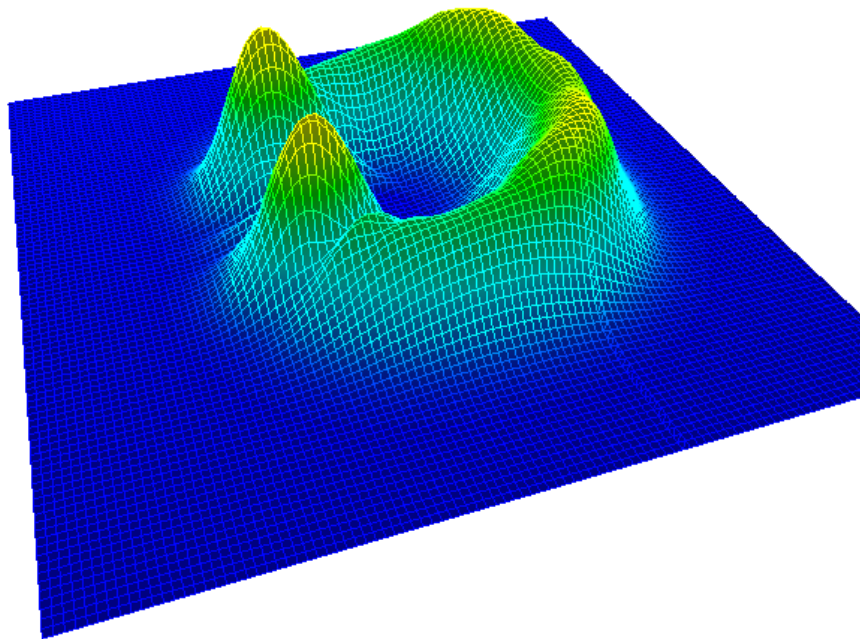


Figura 4.1: Scattering di un pacchetto gaussiano su una buca di potenziale