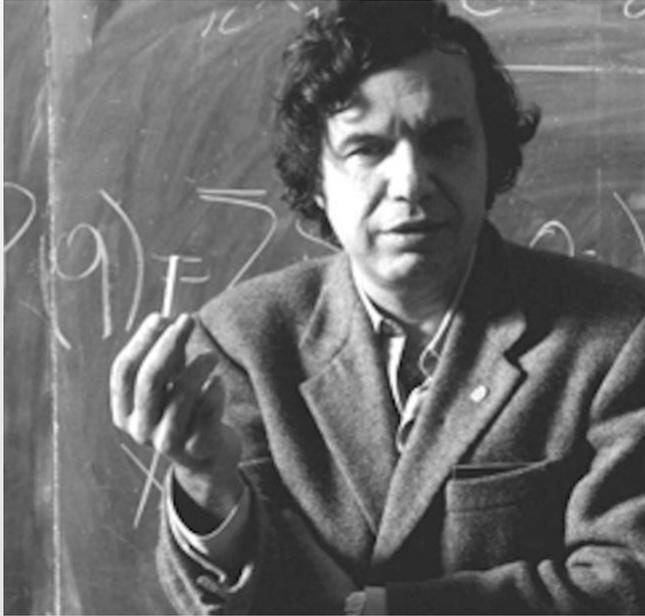




# EQUAZIONE DI LANGEVIN RISOLTA CON IL METODO MONTECARLO

CALCOLO DELLA CORRELAZIONE  
ATTRAVERSO IL METODO DI PARISI

A. Bernardi, G. Nasuti, J. Pedroni



# CONSIDERAZIONI TEORICHE

# EQUAZIONE DI LANGEVIN

L'equazione di Langevin è un'equazione differenziale stocastica della forma:

$$dx = b(x(t))dt + \sqrt{\sigma(x(t))}dw(t)$$

dove  $x(t)$ , e  $b(x(t))$  sono in generale funzioni a valori in  $R^n$ ,  $\sigma(x(t))$  è una funzione reale positiva (in genere una costante) e  $w(t)$  è il moto browniano standard in  $n$  dimensioni.

# EQUAZIONE DI LANGEVIN

Inoltre è dimostrabile che il processo  $x(t)$  è caratterizzato da una densità di probabilità  $P(x, t)$  la cui evoluzione temporale è dettata dall'equazione di Chapman-Kolmogorov:

$$\frac{\partial P(x, t)}{\partial t} = \frac{1}{2} \Delta(\sigma(x)P(x, t)) - \nabla \cdot (b(x)P(x, t))$$

Scegliendo accuratamente, tra una famiglia di funzioni dette «drift»,  $b(x)$  in modo che sia un campo irrotazionale e  $\sigma$  costante, possiamo trovare la soluzione che ci interessa per  $t$  molto grande.

# EQUAZIONE DI LANGEVIN

In particolare se poniamo  $b(x) = -\nabla\rho(x)$  l'equazione diventa:

$$\frac{\partial P(x, t)}{\partial t} = \nabla \left( \frac{1}{2} \sigma \nabla P(x, t) - P(x, t) \nabla \rho(x) \right)$$

che si annulla per  $P(x, t) = e^{-2\frac{\rho(x)}{\sigma}}$ .

# EQUAZIONE DI LANGEVIN

Possiamo ora costruire una EDS che ha come soluzione stazionaria una particolare forma di  $P(x)$ :

$$P(x) = \frac{e^{-\beta H(x)}}{Z}$$

Ovvero con  $b(x) = -\nabla H(x)$  e  $\sigma = \frac{2}{\beta}$ .

La soluzione che abbiamo costruito è la distribuzione di Gibbs, che ci permette di valutare medie statistiche in Meccanica Statistica Classica e in Meccanica Quantistica nella formulazione Euclidea (tempo immaginario).

# EQUAZIONE DI LANGEVIN

Dunque è possibile costruire un processo aleatorio  $x(t)$  convergente per tempi grandi ad una distribuzione stazionaria scelta e questo permette di utilizzare una EDS come algoritmo di una simulazione Montecarlo.

Prima di poter procedere con il calcolo computazionale occorre discretizzare l'equazione di Langevin:

$$dx_i = b(x_i)d\tau + \sqrt{\sigma}dw$$

L'indice  $i$  rappresenta la discretizzazione del tempo lungo un reticolo unidimensionale,  $\tau$  è il tempo di evoluzione dell'equazione di Langevin, mentre  $dw$  è il contributo browniano, ovvero, ad ogni passo, un valore estratto da una distribuzione gaussiana centrata in 0.

# CALCOLO DELLA CORRELAZIONE

Possiamo ora utilizzare quanto visto in Meccanica Quantistica (formulazione Euclidea) per studiare funzioni di correlazione di processi diffusivi e quindi avere informazioni sullo spettro energetico. Infatti, considerando la funzione:

$$\mathbb{E}[x(t)x(s)] = \langle E_0 | q(it)q(is) | E_0 \rangle = \sum_E |\langle E_0 | q | E \rangle|^2 e^{-\frac{E-E_0}{\hbar}|t-s|}$$

dove  $q$  è l'operatore posizione e la somma è estesa a tutto lo spettro dell'Hamiltoniana.

# CALCOLO DELLA CORRELAZIONE

Studiando la funzione per  $|t - s|$  grandi l'andamento è dominato dal primo livello eccitato  $\mathbb{E}[x(t)x(s)] \approx |\langle E_0 | q | E \rangle|^2 e^{-\frac{E-E_0}{\hbar}|t-s|}$  e quindi dalla pendenza della curva di correlazione in scala logaritmica si ottiene il gap di energia.

Ora occorre trovare un metodo per calcolare questa correlazione.

# METODO DI PARISI

Nel 1981 Parisi fornisce un metodo molto intuitivo per stimare la correlazione attraverso l'equazione di Langevin.

L'idea è quella di modificare la funzione di energia nel fattore di Gibbs come:

$$e^{-\beta H(x)} \rightarrow e^{-\beta[H(x)+\lambda x_j]}$$

che ad un valore fissato di  $j$  permette di calcolare la correlazione connessa  $\langle x_i x_j \rangle_c$  come:

$$\langle x_i x_j \rangle_c = \frac{d}{d\lambda} \langle x_i \rangle |_{\lambda=0}$$

# METODO DI PARISI

L'equazione di Langevin ottenuta è quindi:

$$dx_i = [b(x_i) + \lambda \delta_{ij}]d\tau + \sqrt{\sigma}dw$$

in cui porremo  $\sigma = 1$ .

Avendo un termine che dipende da  $\lambda$  possiamo pensare di sviluppare la soluzione come:

$$x_i(\lambda, t) = x_i^{(0)}(t) + \lambda x_i^{(1)}(t) + O(\lambda^2)$$

# METODO DI PARISI

Quello che otteniamo è un sistema di due EDS accoppiate. Posto il drift come  $b_i(x) = -\frac{\partial H}{\partial x_i}$  troviamo:

$$\left\{ \begin{array}{l} dx_i^{(0)}(t) = b_i(x)dt + dw_i(t) \\ dx_i^{(1)}(t) = \sum_k \frac{\partial b_i(x)}{\partial x_k} x_k^{(1)} dt + \delta_{ij} dt \end{array} \right.$$

# METODO DI PARISI

Dal teorema di fluttuazione-dissipazione si può concludere che:

$$\langle x_i x_j \rangle = \frac{d}{d\lambda} \langle x_i \rangle \Big|_{\lambda=0}$$

Da cui:

$$\langle x_i x_j \rangle = \lim_{\tau \rightarrow \infty} \frac{1}{\tau} \int_0^{\tau} x_i^{(1)}(t) dt$$

Ovvero la correlazione  $\langle x_i x_j \rangle_c$  è data semplicemente dalla media di  $x_i^{(1)}$ , che a livello computazionale non è altro che una media discreta.

# METODO DI PARISI

Da quanto visto finora sappiamo che l'evoluzione del sistema assume, una volta integrata, questa forma:

$$x(\tau + d\tau) - x(\tau) = \int_{\tau}^{\tau+d\tau} b[x(t)]dt + \sqrt{\sigma}[w(\tau + d\tau) - w(\tau)]$$

Possiamo dunque computare con il metodo dei trapezi questo integrale, discretizzandolo come:

$$x(\tau + d\tau) - x(\tau) = \frac{b[x(\tau + d\tau)] + b[x(\tau)]}{2}d\tau + \sqrt{\sigma}[w(\tau + d\tau) - w(\tau)]$$

# IMPLEMENTAZIONE

Dato che il drift  $b[x(\tau + d\tau)]$  che compare nella discretizzazione è funzione di quello che si vuole calcolare occorre integrare al metodo dei trapezi un preliminare calcolo fatto con il metodo di Eulero:

$$x(\tau + d\tau) - x(\tau) = b[x(\tau)]d\tau + \sqrt{\sigma}[w(\tau + d\tau) - dw(\tau)]$$

Questo porta ad un'incertezza di ordine  $O(d\tau^2)$ .

# IMPLEMENTAZIONE

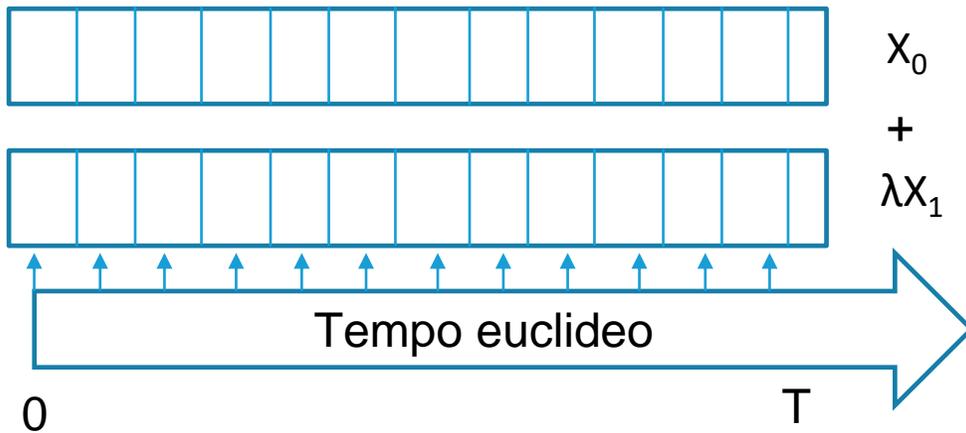
Prendiamo una generica Hamiltoniana unidimensionale discretizzata:

$$H = \sum_{k=0}^{N-1} \frac{m(x_{k+1} - x_k)^2}{2a^2} + \sum_{k=0}^N V(x_k)$$

dove  $a = \frac{T}{N}$  è il passo del tempo Euclideo. Calcoliamo ora il drift al tempo fissato  $k = i$ :

$$b(x_i) = -\frac{\partial H}{\partial x_i} = \frac{m(x_{i+1} - 2x_i + x_{i-1}))}{a^2} - V'(x_i)$$

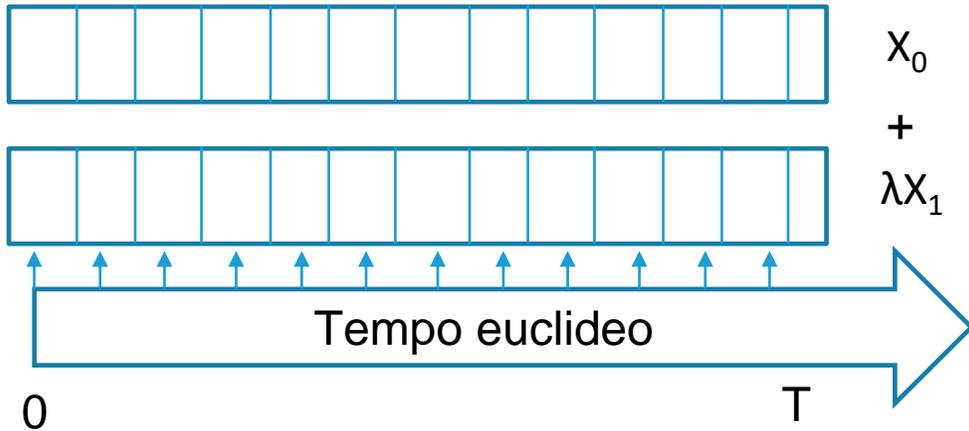
# IL PROGRAMMA



## Le Variabili utilizzate

Variabile	Utilizzo
N	numero di siti indipendenti fatti evolvere
T	tempo totale euclideo ( $a=T/N$ )
Tlang	tempo totale di evoluzione di Langevin
Steps	numero di suddivisioni del tempo di Langevin ( $d\tau=T\text{lang}/\text{steps}$ ) Intervallo che prendiamo come riferimento per risolvere gli integrali con il metodo di eulero e dei trapezi.
g	costante di accoppiamento del potenziale
skips	intervallo di campionamento della correlazione (per evitare fenomeni di autocorrelazione)
therm	numero di passi fatti prima di prendere misure
exps	numero di esperimenti indipendenti
trim	intervallo scelto per il fitting (per tempi troppo alti la correlazione cambia comportamento)

# IL PROGRAMMA



Infatti se per prova poniamo  $N=4$ :

$$\begin{pmatrix} -2 & 1 & 0 & 1 \\ 1 & -2 & 1 & 0 \\ 0 & 1 & -2 & 1 \\ 1 & 0 & 1 & -2 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ w \\ z \end{pmatrix} = \begin{pmatrix} -2x+y+z \\ w+x-2y \\ -2w+y+z \\ w+x-2z \end{pmatrix}$$

$$x(\tau + d\tau) - x(\tau) = \int_{\tau}^{\tau+d\tau} b[x(t)]dt + \sqrt{\sigma}[w(\tau + d\tau) - w(\tau)]$$

$$\text{Con } b(x(t)) = \frac{m(x_{i+1} - 2x_i + x_{i-1}))}{a^2} - kx_i - gx_i^3$$

Applicando il metodo di Parisi e poi Eulero si ottiene che:

$$x_{0eu} = x_0 + D^2 x_0 - x_0 - 4 * g * x_0 * x_0 * x_0 * \tau + \text{noise} \quad (k=1)$$

Matrice che rappresenta il termine cinetico dell'Hamiltoniana

$$\begin{bmatrix} -2 & 1 & 0 & \dots & 0 & 1 \\ 1 & -2 & 1 & 0 & \dots & 0 \\ 0 & 1 & -2 & \ddots & 0 & \vdots \\ \vdots & 0 & \ddots & \ddots & 1 & 0 \\ 0 & \vdots & \dots & 1 & -2 & 1 \\ 1 & 0 & \dots & 0 & 1 & -2 \end{bmatrix}$$

Termine di rumore browniano  
Noise =  $D * \text{randn}(N, 1)$

$D = \sqrt{\frac{2\tau N \hbar}{2\pi T}}$  Coefficiente di diffusione

```

j = 0:therm
noise = D*randn(N,1);
x0 = x0 + (D2*x0 - x0 - 4*g*x0.*x0.*x0) * tau + noise;
x1 = x1 + (D2*x1 - x1 - 12*g*x0.*x0.*x1) * tau;
x1(src) = x1(src) + tau;
x2 = x2 + (D2*x2 - x2 - 12*g*x0.*x0.*x2) * tau;
x2(src) = x2(src) + 2*tau*x0(src);

```

```

for m = 1:skip
    noise = D*randn(N,1);
    x0src = x0(src);
    % method of trapezes
    x0 = x0 + 0.5*tau*((D2*x0 - x0 - 4*g*x0.*x0.*x0)+...
        (D2*x0eu - x0eu - 4*g*x0eu.*x0eu.*x0eu)) + noise;

    x1 = x1 + 0.5*tau*((D2*x1eu - x1eu - 12*g*x0eu.*x0eu.*x1eu) +...
        (D2*x1 - x1 - 12*g*x0.*x0.*x1));
    x1(src) = x1(src) + tau;

    x2 = x2 + 0.5*tau*((D2*x2eu - x2eu - 12*g*x0eu.*x0eu.*x2eu-12*g*x1eu.*x1eu.*x0eu) +...
        (D2*x2 - x2 - 12*g*x0.*x0.*x2-12*g*x1.*x1.*x0));%modifica
    x2(src) = x2(src) + tau*(x0src + x0(src));
end

Xc1 = [Xc1, x1];           % correlazione < x(s) . x(t) >
Xc2 = [Xc2, 2*x2.*x0];   % correlazione < x(s)^2 . x(t)^2 >

end;

Xcorr1 = mean(Xc1)';
Xcum1 = [Xcum1, Xcorr1]; % average on steps*exps/skip...
Xcorr2 = mean(Xc2)';
Xcum2 = [Xcum2, Xcorr2]; % average on steps*exps/skip...

```

```

function [s, f, Xf] = fitcorr(Xc, setup, interv)
% Extract the slope of the correlation produces in LangCluster
% E(x(t)x(s)) ~f*exp(-s|t-s|)
% using fminsearch; Xf returns the fit
% © E. Onofri, U.Parma, LabTeo-2005

N = str2num(setup{1}); % extract parameters
T = str2num(setup{2});

hbar = 1. ;
t = linspace(0,T,N)';

opt = optimset('TolFun', 1e-4,...
    'TolX', 1e-4,...
    'MaxIter', 1e6,...
    'MaxFunEval', 1e6,...
    'Display','on');

ti = t(interv);

z = fminsearch(@corr , [1., 1.], opt, ti, Xc, T);

s = z(1);
f = z(2);

Xf = f*(exp(-s*t)+exp(-s*(T-t)));

```

### Termalizzazione:

Il sistema viene fatto evolvere (tramite l'algoritmo di Eulero) un certo numero di volte per assicurarsi che si sposti dalla condizione iniziale di vettore nullo



### Processo di Langevin:

Il sistema viene fatto evolvere in maniera più precisa (metodo dei trapezzi+Eulero) i dati vengono poi salvati nella matrice Xc1.

Vengono salvati in questa matrice le posizioni del sistema dopo un numero di evoluzioni  $\tau$  pari a skip.



### Analisi dei dati:

Dopo aver raccolto tutti i dati si può procedere al calcolo della correlazione e di conseguenza al calcolo del energy-gap.

Visto che la relazione è esponenziale si effettua prima una regressione lineare, in modo tale che dal metodo dei minimi quadrati si possa risalire al coefficiente della retta (Energy-gap)



# OSCILLATORE ARMONICO

Al fine di testare la precisione del metodo utilizzato e di ottimizzare i parametri di quest'ultimo, abbiamo fatto preliminarmente alcune run studiando un'hamiltoniana di oscillatore armonico, di cui sappiamo quindi calcolare in modo preciso l'energy gap.

# OSCILLATORE ARMONICO

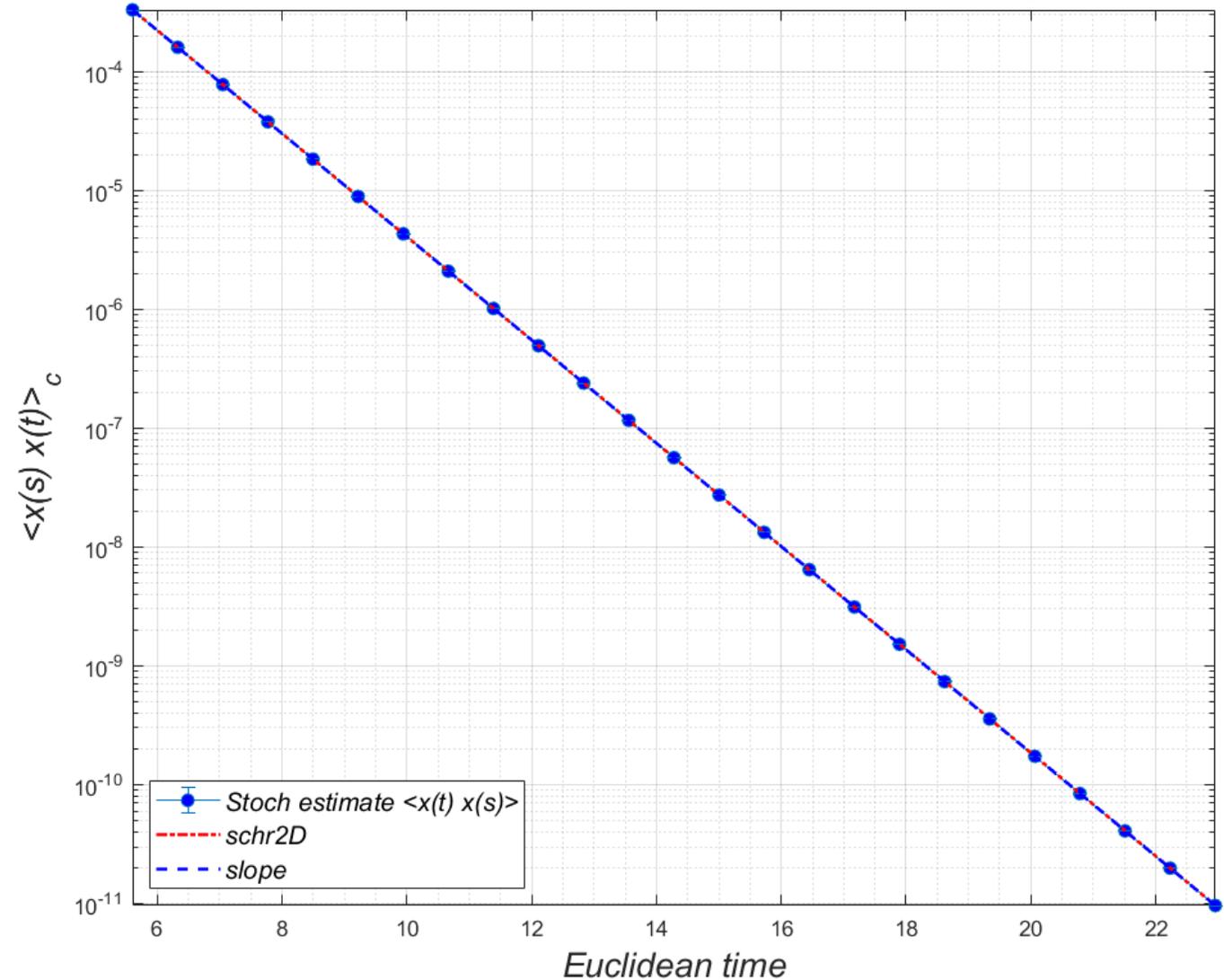
$\Delta E$  Metodo di Parisi

1.0001

$\Delta E$  Esatto

1.0000

Stochastic quantization - 2D-Armonic Oscillator -  $E_1 - E_0$  at  $g=1$



# OSCILLATORE ANARMONICO

Abbiamo quindi studiato un'hamiltoniana di oscillatore anarmonico unidimensionale con potenziale:

$$V(x) = \frac{x^2}{2} + gx^4$$

dove  $g$  è detta costante di accoppiamento.

L'obiettivo è stato quello di calcolare, per tre diverse costanti di accoppiamento,  $\Delta E$  al variare di  $\tau$  al fine di poter costruire un fit quadratico e ricavare il valore del gap energetico per  $\tau \rightarrow 0$ .

Quanto trovato è stato poi confrontato con i corrispondenti valori calcolati attraverso metodi perturbativi o risolvendo lo spettro della matrice Hamiltoniana.

# OSCILLATORE ANARMONICO

**$\Delta E$  Metodo di Parisi**

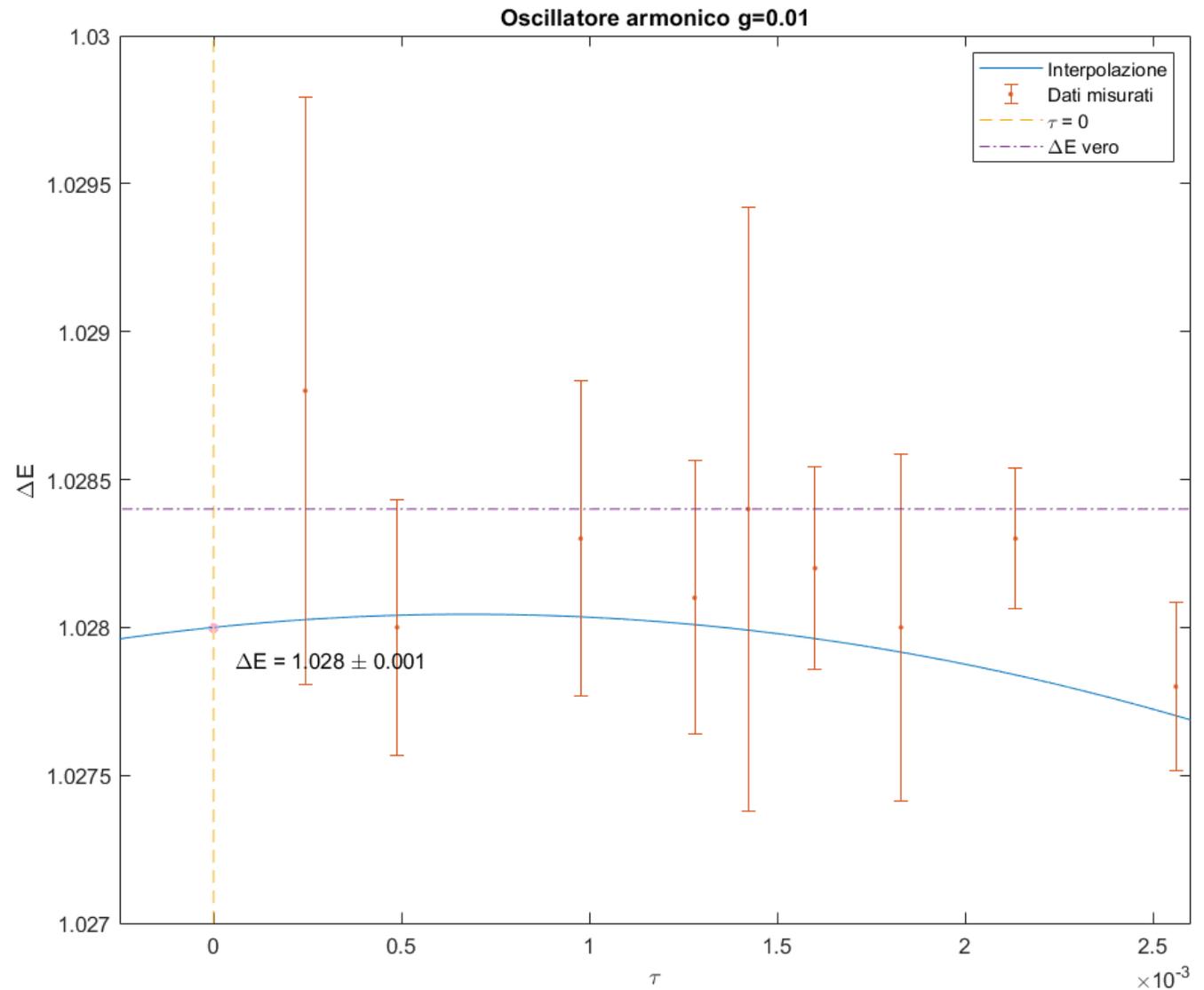
$1.028 \pm 0.001$

**$\Delta E$  Spettro**

1.0284

**$\Delta E$  Perturbativo**

1.0284



# OSCILLATORE ANARMONICO

$\Delta E$  Metodo di Parisi

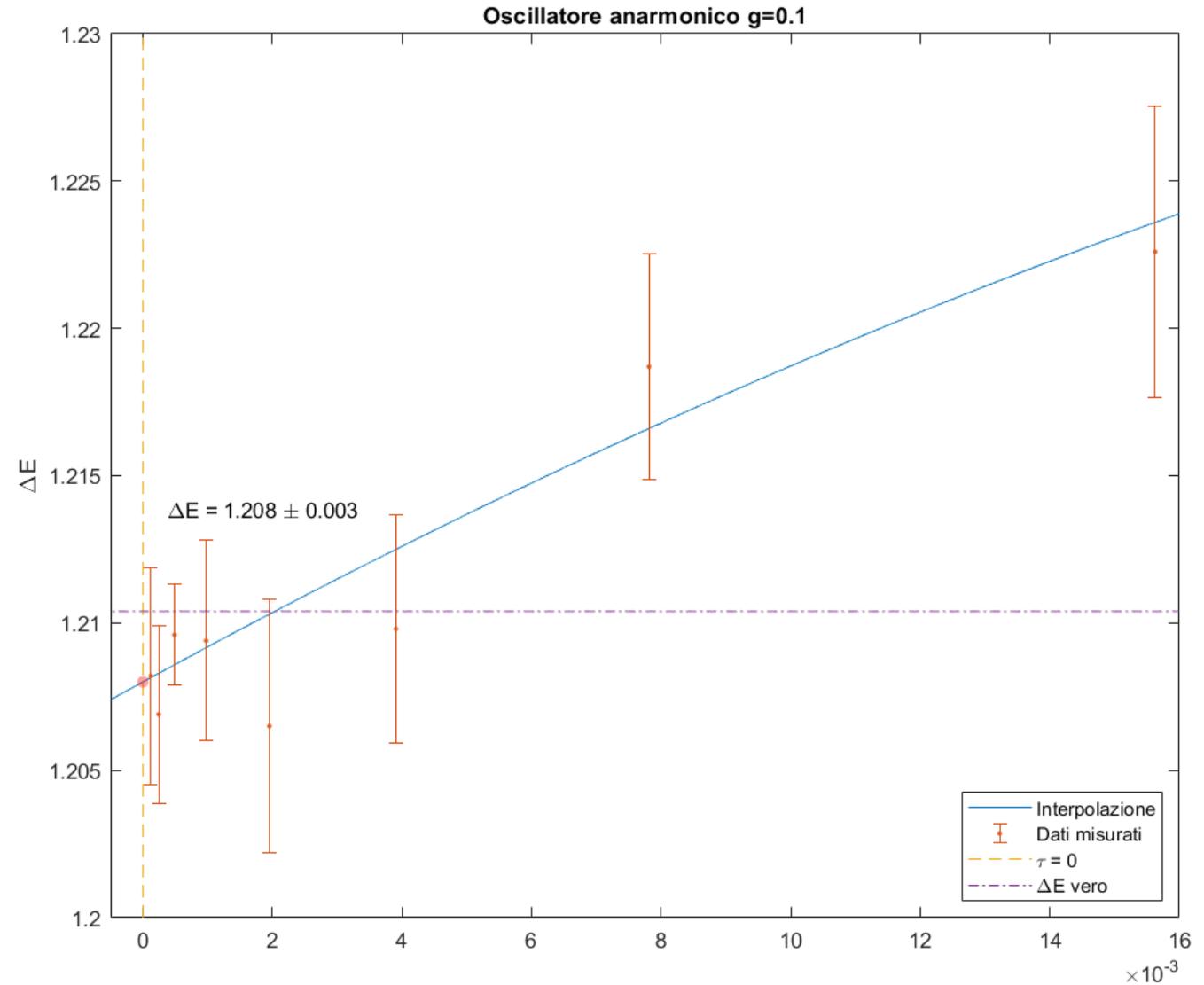
$1.208 \pm 0.003$

$\Delta E$  Spettro

1.2104

$\Delta E$  Perturbativo

1.2053



# OSCILLATORE ANARMONICO

$\Delta E$  Metodo di Parisi

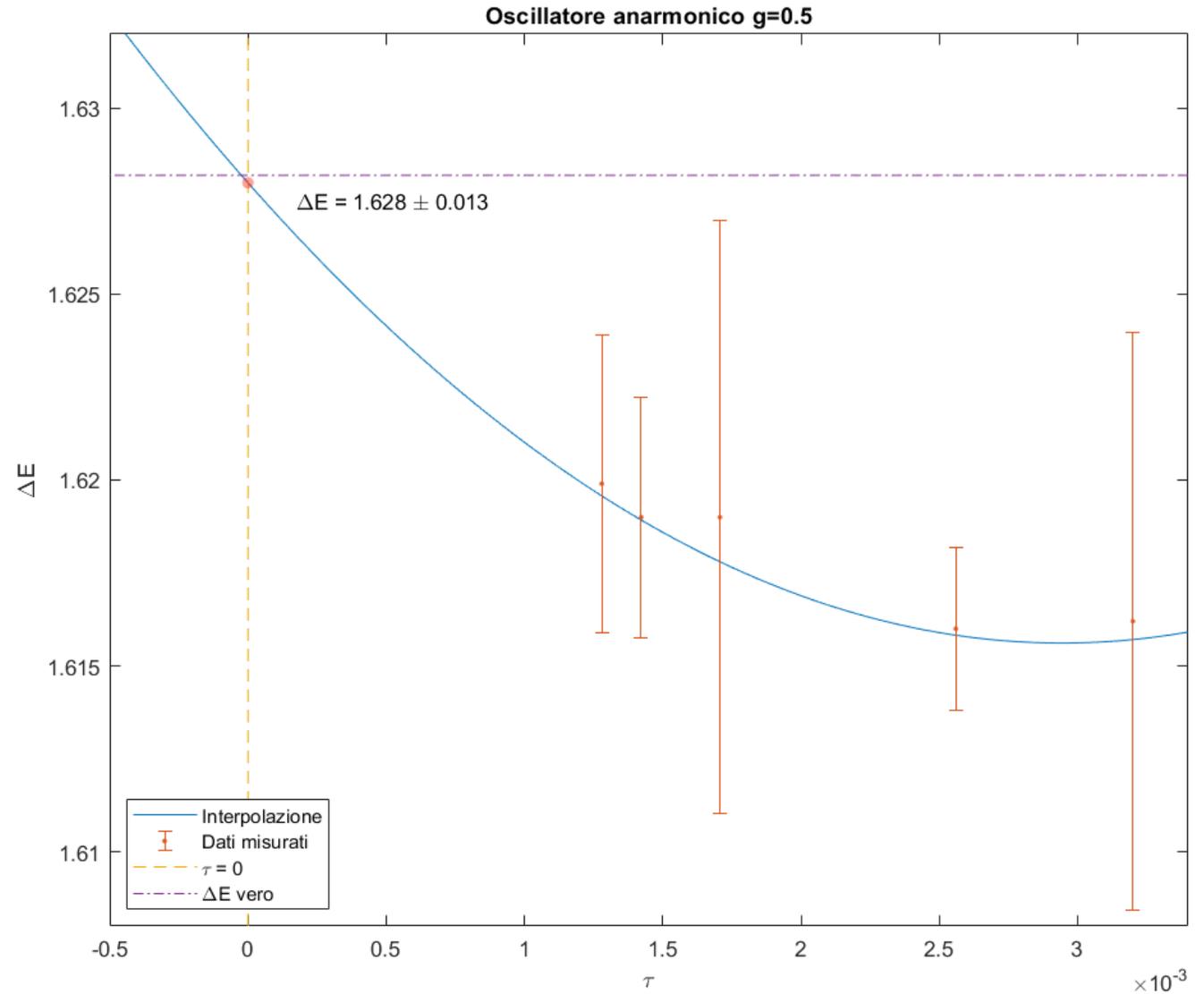
$1.628 \pm 0.013$

$\Delta E$  Spettro

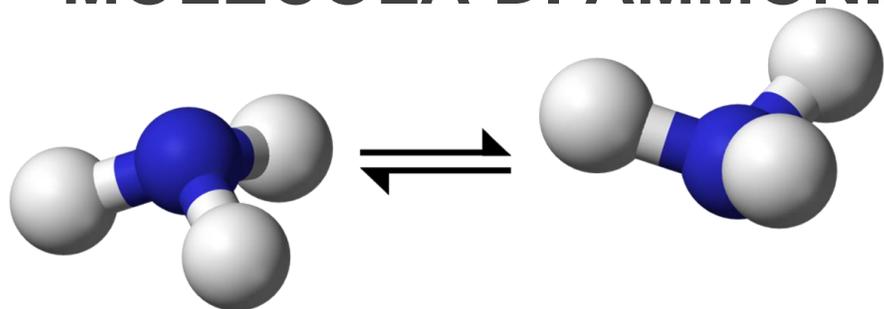
1.6282

$\Delta E$  Perturbativo

1.4796

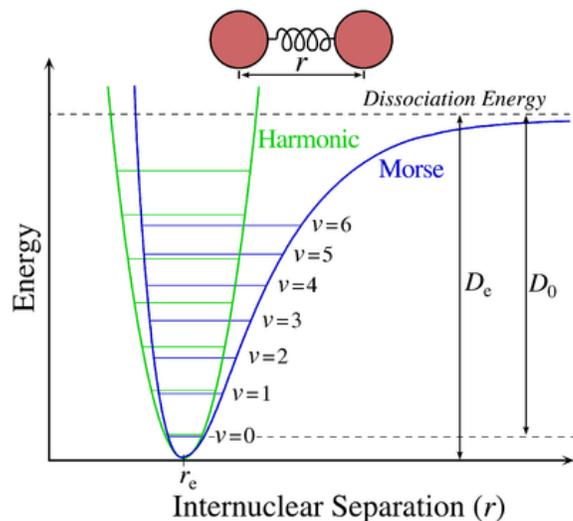


# MOLECOLA DI AMMONIACA



La molecola di ammoniaca nelle sue configurazioni dovute all'inversione piramidale.

Assumiamo che l'energia tra due atomi sia descritta dal potenziale di Morse:  $E = V_0 [1 - e^{-a(r-r_e)}]^2$



Variabile	descrizione
$V_0$ In figura chiamato $D_e$	Profondità del legame (atomi non legati)
$a$	Controlla la larghezza del potenziale
$r$	E' la distanza fra gli atomi
$r_e$	Distanza di equilibrio

Per la molecola di ammoniaca andrà considerata la somma di tutti i potenziali generati dai vari atomi:

```
function b = Drift(x)
    xN = x(:,1:3);
    xH1 = x(:,4:6);
    xH2 = x(:,7:9);
    xH3 = x(:,10:12);

    F_NH1 = MP(xN ,xH1,L1,V0(1),nu(1));
    F_NH2 = MP(xN ,xH2,L1,V0(1),nu(1));
    F_NH3 = MP(xN ,xH3,L1,V0(1),nu(1));
    F_H12 = MP(xH1,xH2,L2,V0(2),nu(2));
    F_H13 = MP(xH1,xH3,L2,V0(2),nu(2));
    F_H23 = MP(xH2,xH3,L2,V0(2),nu(2));

    bN = F_NH1 + F_NH2 + F_NH3;
    bH1 = -F_NH1 + F_H12 + F_H13;
    bH2 = -F_NH2 - F_H12 + F_H23;
    bH3 = -F_NH3 - F_H13 - F_H23;

    b = -[bN, bH1, bH2, bH3];
end
```

In realtà la funzione mp restituisce già il gradiente dell'energia di Morse (sfruttando la linearità del gradiente).

$$-\nabla E = \frac{2V_0 a e^{a(r-r_e)} (1 - e^{a(r-r_e)})}{r} \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$



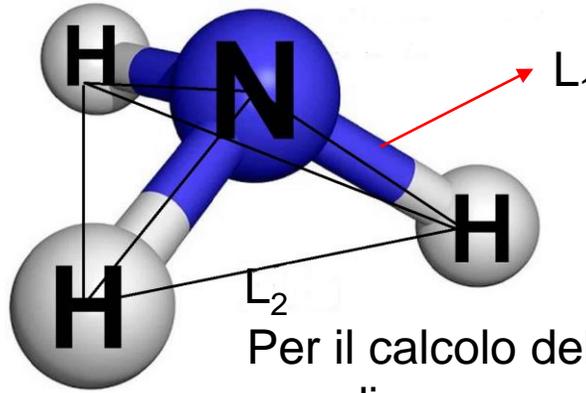
Termine di Drift

# IL PROGRAMMA

Condizioni iniziali:

Volume iniziale:

$$V_0 = \frac{L_2^2}{2} * \sqrt{3L_1^2 - L_2^2}$$



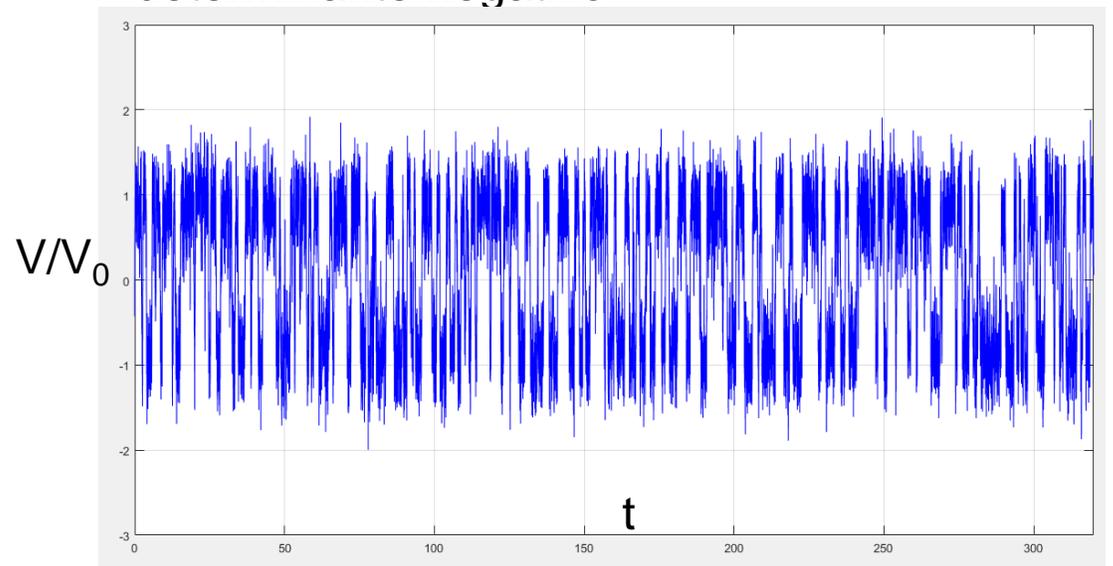
Origine del sistema di riferimento nel baricentro dei tre idrogeni (nella posizione iniziale)

Iniziamo l'evoluzione del sistema attraverso il metodo di Eulero.

```
for m = 1:skip
    x = KinEvol(x,W);
    x = x + tau * Drift(x);
    x = x + D*randn(N,12);
end

if(clustering=='y')
    [x,nf,nh] = cluster_flip(x, a, N); % flip ogni "skip" passaggi
    hits = hits + nh;
    accept = accept + nf; % numero di clusters flippati
    disp(['run:',num2str(run),...
        '-sweep:',num2str(sweep-1),...
        '-Accept-rate: ', num2str(accept/hits)]);
end
```

Per il calcolo del volume al variare del tempo prendiamo un secondo sistema di riferimento solidale all'atomo di azoto, indicando con  $e_i$  i vettori che identificano la posizione degli idrogeni, scriviamo la matrice  $V=(e_1, e_2, e_3)$ . Il  $V=\det(V)$ : se avviene l'inversione si otterrà un determinante negativo

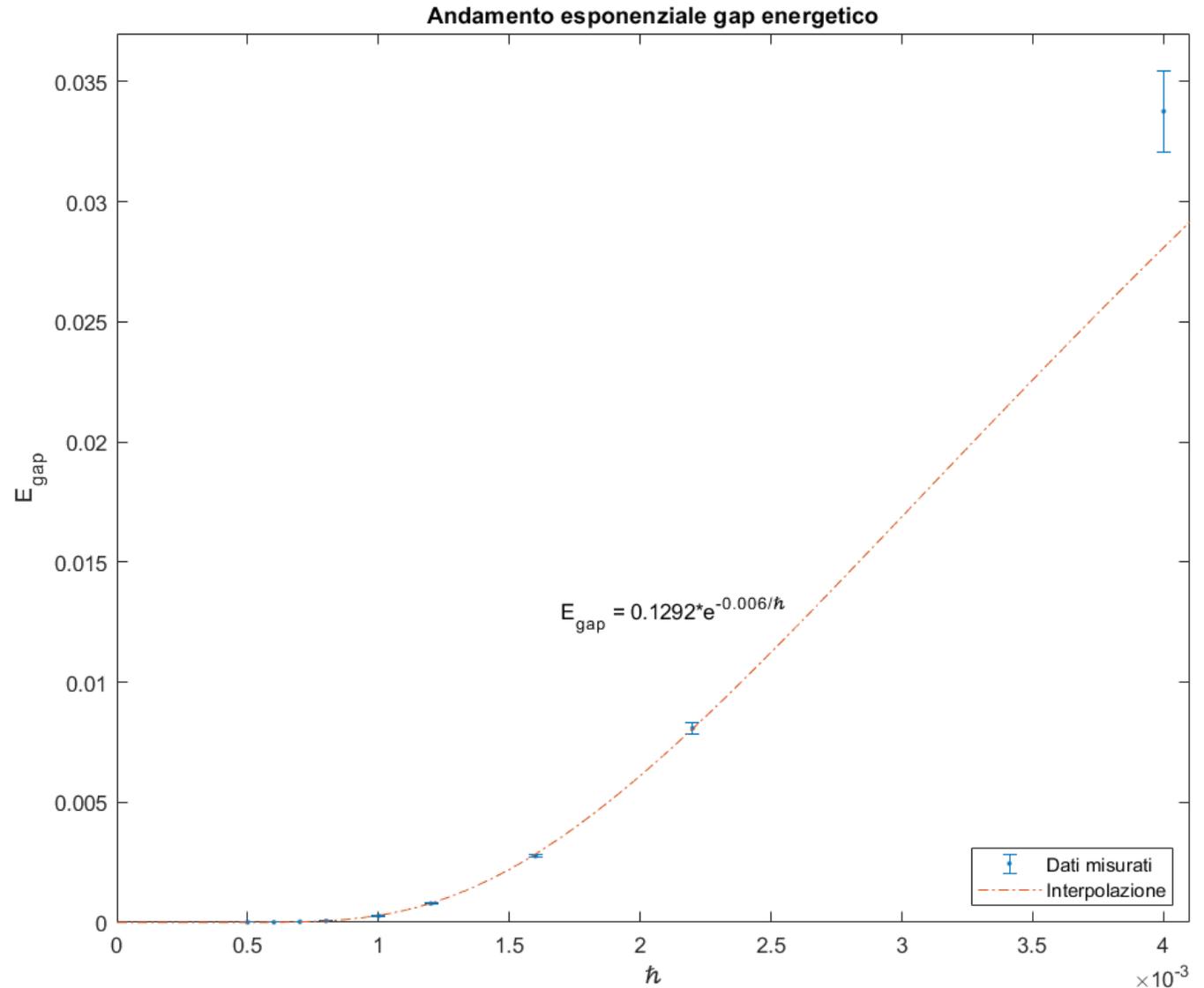


# MOLECOLA DI AMMONIACA

Questo grafico rappresenta l'energy gap in funzione di  $\hbar$  e come si può vedere l'andamento è quello descritto da:

$$E_{gap} = A * e^{-\frac{S}{\hbar}}$$

Per quanto riguarda la costante di normalizzazione  $A$  e l'azione  $S$ , queste sono state trovate come descritto di seguito.

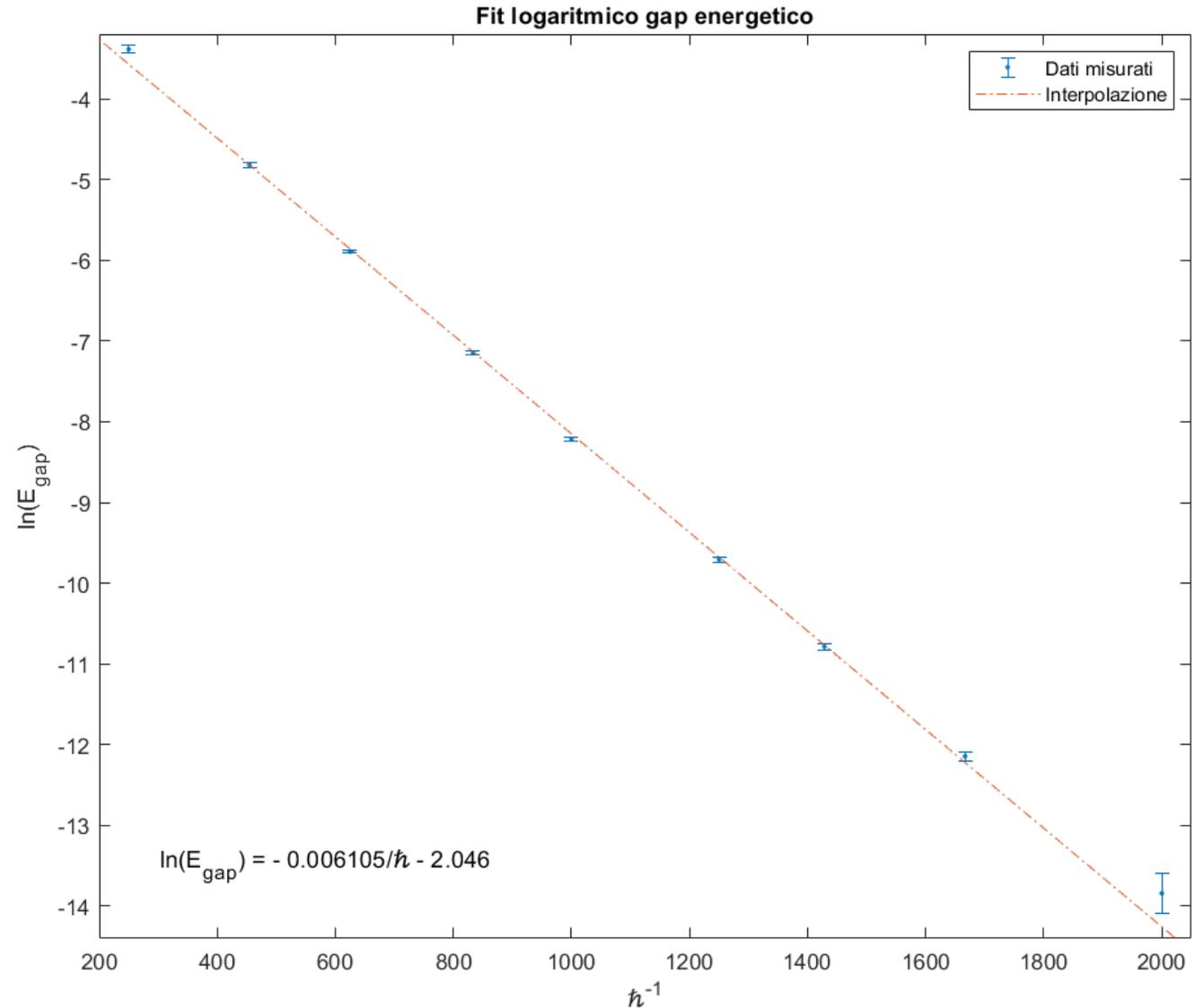


# MOLECOLA DI AMMONIACA

Questo grafico rappresenta l'interpolazione che abbiamo fatto per trovare le costanti prima citate.

Il fit è stato costruito dopo aver linearizzato in questo modo:

$$\ln(E_{\text{gap}}) = \ln(A) - S\hbar^{-1}$$



# MOLECOLA DI AMMONIACA

I dati ottenuti seguono abbastanza bene l'andamento previsto, specialmente per  $\hbar \rightarrow 0$ , limite della fisica classica, in cui l'energy gap va a zero e i livelli energetici diventano continui.

Quanto trovato è quindi in accordo con il modello teorico e con quanto sappiamo riguardo all'effetto tunnel dell'atomo di azoto e quindi all'inversione piramidale della molecola, poiché avendo essa un periodo di inversione

$$\bar{T} \propto \frac{\hbar}{\Delta E}, \text{ questo tende ad infinito nel limite classico.}$$

# BIBLIOGRAFIA

- “Lezioni di Metodi Probabilistici della Fisica”, E. Onofri;
- “Correlation functions and computer simulations”, G. Parisi, 1980;
- “Instantons in Field Thoery”, D. Oliver, S. Sciuto, R. J. Crewther, 1978;
- “Path Integral Monte Carlo applicati a Meccanica Quantistica”, N. Macellari, C. Meneghelli, 2006;
- “Correlazione con il metodo di Parisi”, T. Bilzi, M. Savi, A. Nervo, 2022.



## CODICI UTILIZZATI

- Per lo studio dell'oscillatore armonico abbiamo utilizzato 'LangParCheck';
- Per lo studio dell'oscillatore anarmonico abbiamo utilizzato 'LangPar1D' e 'schr1D';
- Per lo studio della molecola di ammoniaca abbiamo utilizzato 'MorseAmmonia' e 'runMA'.

Tutti i programmi MATLAB citati ci sono stati forniti dal professore E. Onofri.



**FINE DELLA PRESENTAZIONE**

Grazie per l'attenzione.