

# Résolution d'une équation stochastique

## Contexte

On veut simuler un retour vers l'équilibre en température interne de molécules de gaz diatomiques. L'équation à résoudre est la suivante

$$\partial_t f = \frac{2}{\tau} \nabla_E \cdot ((E - RT) + \partial_E(ERTf))$$

où  $f(t, E)$  représente le nombre de nombre de molécules avec l'énergie  $E$  à l'instant  $t$ . et  $\tau$  un temps caractéristique du phénomène  $R = 280 \text{SI}$  la constante du gaz parfait et  $T$  la température interne moyenne du gaz.

# Algorithme de résolution

## Algorithme

- ① Tirer au sort 500000 énergies  $E_p$  de 100000J à 200000J
- ② calculer  $T_{int} = moyenne(Energie)/R$
- ③ Entre  $t$  et  $t + dt$  l'algorithme est :

$$E_p^{n+1} = \frac{1}{1 + 2\frac{dt}{\tau}} \left( E_p^n + \frac{RTdt}{\tau}(1 + \sigma^2) + 2\sqrt{\frac{dt}{\tau}RTE_p^n}\sigma \right)$$

où  $\sigma$  est un nombre aléatoire gaussien et  $dt$  le pas de temps.

## Cas-test

- $\tau = 5s$
- $dt = 1/100s$
- $N_p = 500000$  : nombre de particules
- $T = 300, R = 280$

# Programmation

## Codage

- Coder les différentes étapes.
- Stocker à chaque itération  $T_{int}$  pour suivre son évolution
- A la fin écrire le tableau d'évolution de  $T_{int}$
- Créer un histogramme de taille 100 des énergies internes

# Optimisation

## Utilisation de gprof

- Compiler en option -pg -O0  
ex : *gcc -Wall -pg test\_gprof.c test\_gprof\_new.c -o test\_gprof*
- Executer le cas-test
- Analyse grossière :  
*gprof test\_gprofmon.out > analysis.txt*
- Analyse fine :  
*gprof -l test\_gprofmon.out > analysisfine.txt*