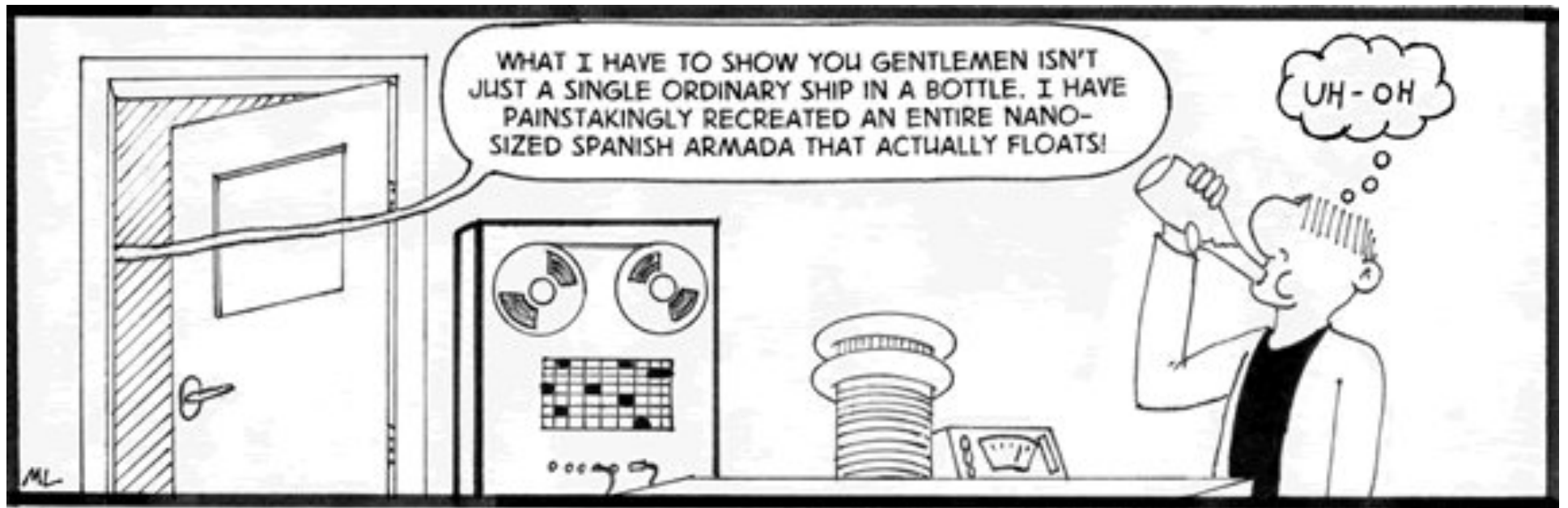


## Joking on nano-matter ....



# **Molecular Simulations**

Introduction

Computer Simulations allows us to:

- study properties of many particle systems.
- calculate most of the properties that can be measured in an experiment

We will study to types of simulations:

Monte Carlo (MC)

Molecular Dynamics (MD)

Experiments measure quantities averaged over the time of the experiment.

Since we usually aim to use computer simulations as the numerical counterpart of the experiment, we will learn how to compute averages using the language of statistical mechanics.

However CS allow us to calculate MORE than what can be measured.

#### Example

in an MD simulation of a system of water molecules, we can calculate the velocity of an individual molecule at a given time. However, that quantity can not be measured in a typical experiment.

In MC we sample the phase space by generating many states of the system in a **stochastic** (random) manner.

MD simulations compute the motions of individual molecules.

It's a modern realization of an old idea: the behavior of a system can be computed if we have a set of initial conditions plus the forces of interaction. It is a **deterministic** mechanical view of nature.

## Deterministic vs Predictable.

**Deterministic:** Output is causally connected to the input.

**Calculable:** are deterministic situations in which an algorithm permits us to compute the output if the inputs are known.

**Predictable:** calculable situations where the algorithm can be numerically implemented to actually obtain the output.

Calculable situations may be unpredictable because of the large number of input needed or extremely high precision is needed.

**Unpredictable:** We can identify many systems as unpredictable: stock markets, weather. When the components of the system interact non-linearly the behavior may be unpredictable.

“Whether it is going to rain in two weeks” is deterministic but unpredictable.

CS are performed on models, not real things.

The art of simulations is bound to the art of modeling. In computer simulations we use a machine to compute the behavior of a many body “model”. It’s not guaranteed that the computed behavior is representative of the model, or that the model mimics reality.

Simulated Model:

Model of molecular interactions  
Model for system-environment interactions

Model for molecular interactions:

force law or energy function, shape  
and electron clouds, symmetry,  
rigid or flexible or spherically  
symmetric (isotropic),  
pairwise additivity:  $U = \sum u_{ij}$ .

Model for system-environment interactions:

Include boundary conditions force  
with substrate.

Molecular-scale simulation:

- 1- Construction of a model
- 2- Calculation of trajectories
- 3- Analysis of trajectories to obtain property values.

In MD, the trajectories are positions connected in time, calculated solving the equations of motion.

In MC, the positions are NOT connected in time, are generated stochastically, and depend only on the previous configuration (Markov chain).



MC:

Typically a fixed number of molecules in a fixed volume at a fixed temperature.

The goal is to performed statistical-mechanical ensemble averages.

Ensemble averages are calculated by accumulating the integrand at randomly generated values of the atomic positions.

$$\langle A \rangle = \frac{1}{Z} \int \exp[-\beta U(r^N)] A(r^N) dr^N$$

MD:

For systems in equilibrium, typically in NVE ensemble (isolated system, constant energy)

The positions are changing following Newton's law for several thousand times.

$$\langle A \rangle = \lim_{t \rightarrow \infty} \frac{1}{t} \int A(t) dt$$

Ergodic hypothesis:

time average = ensemble average

MC approach = MD approach

## Monte Carlo vs Molecular Dynamics.

MC easier to program.

MC can be used in cases where the force is difficult to derive from the potential energy, for example for hard sphere potential, or ab-initio calculated potentials.

For computation of simple equilibrium properties MC and MD are equally effective, both require  $\sim$  same computing time.

MD is better to evaluate heat capacities, compressibilities, transport coefficients.

MD is used in NVE, for adsorption problems where the ensemble is Grand Canonical we must use MC.

MD usually done for system containing 100-1000 particles interacting with relatively short range forces: interaction should be small when the particles are separated by half the size of the box.

Also, the characteristic relaxation time must be small enough so we can generate several relaxation times in one simulation.

**Basic idea:**

MC: essence of technique involves use of probability

MD: deterministic

Many intermediate methods:

Force biased Monte Carlo (Mostly stochastic... )

Simulation of proteins in water:

A combination of Molecular Mechanics (deterministic), Monte Carlo and Molecular Dynamics are used to simulate different aspects of the system.

**Picture:** atoms/molecules are classical particles moving in a field created by other molecules, surfaces, solvents, etc.

**Goal:** to learn about the behavior of molecules on the atomic (microscopic) level and relate this behavior to the experimentally measurable properties (macroscopic level).

**Results:**

**Configuration:** positions (and velocities) of a collection of molecules.

If I know the **positions** of a collection of molecules as a function of time, I know **EVERYTHING** about the **static (or thermodynamic)** properties of the system: (**P, S, E, phase equilibria, etc.**)

If I know the **positions and velocities** of a collection of molecules as a function of time, I know **EVERYTHING** about the **dynamic** properties of the system as well. (mean square displacements, **T**, diffusion, spectroscopy, response time.

**Important Underlying Assumption:** that I know how the particles interact with themselves and their surroundings: Assume I know  $U_{\text{pot}}$ .

## **Overview of Computer Simulation**

Select a model

Initialize system

Generate configurations (positions and momenta (MD))

Calculate properties of interest

Start by discussing how properties are determined from configurations.

Then we are able to explain the features that must be present in the model.

### **Determining properties**

**Mechanical State:** (microscopic state)

positions (and momenta) of particles at an instant in time.

This is the information obtained in a simulation.

**Thermodynamic State:** (equation of state) macroscopic state

Characterized by a few variables, depends on **ensemble**

Canonical (NVT)

Microcanonical (EVT)

Grand canonical ( $\mu$  VT)

Isobaric, isothermal (NPT)

Once these variables are defined, state functions are determined:

density

chemical potential

shear viscosity

heat capacity

diffusion coefficient

structure factor

**State functions are characterized by thermodynamic variables, NOT by the positions and momenta that describe the instantaneous mechanical state.**

Statistical Mechanics: provides connection between the thermodynamic state and the mechanical state.

Assumption:

$$\begin{aligned} A_{obs} &= \langle A \rangle_{time} = \langle A(\Gamma(t)) \rangle = \lim_{t_{obs} \rightarrow \infty} \frac{1}{t_{obs}} \int_0^{t_{obs}} A(\Gamma(t)) dt \\ &= \langle A \rangle_{stat} = \frac{1}{Z} \int \exp[-\beta U(r^N)] A(r^N) dr^N \end{aligned}$$

$A_{obs}$  = observed value of a property

$\langle A \rangle_{stat}$  = ensemble average of the property (MC)

$\langle A \rangle_{time}$  = time averaged value of the property (MD)

$\Gamma(t)$  = generalized coordinates (positions and momenta) as a function of time



### **Can this method be practical?**

Can enough phase space be explored to give adequate time averages?

Can the behavior of  $\sim 10^{23}$  molecules be simulated by 100-1000 particles?

Can thermodynamic consistency be achieved?

Do you get the same results if different initial conditions (positions/momenta) are used?

### **REAL QUESTION:**

IF I know the coordinates (positions/momenta) of  $\sim 200$  particles, can I calculate the thermodynamic (and dynamic) properties of the system?

YES! Using MD and/or MC simulation techniques

# of particles is not much, but # of **CONFIGURATIONS** is huge

# Fundamentals of Theory of Probability



## DEFINITION of PROBABILITY

Let's consider RANDOM event with a countable set of possible outcomes:  
 $A_1, A_2, \dots, A_q$  (e.g. rolling a die)

We repeat the test  $N \gg 1$  times, getting the outcome  $A_k$   $N_k$  times.

**Definition** of the **probability** of event  $A_k$ :

$$p(A_k) = p_k = N_k/N \quad (N \gg 1)$$

Properties:

$$\sum_k p_k = 1$$
$$0 \leq p_k \leq 1$$

If  $A_k$  and  $A_j$  are mutually exclusive events then

$$P(A_k \text{ and } A_j) = 0 \quad \text{and} \quad P(A_k \text{ or } A_j) = p_k + p_j.$$

## Joint probability

Let's consider two events A and B with set of outcomes  $A_1, A_2, \dots$  and  $B_1, B_2, \dots$

Definition: Joint probability

$$p_{ij} = P(A_i \text{ and } B_j)$$

If A and B are independent then  $p_{ij} = p_i \times p_j$

If A and B are not independent, we define the conditional probability that  $B_j$  occurs given that  $A_i$  occurs:

$$p(j|i) = \frac{p_{ij}}{\sum_k p_{ik}} = \frac{p_{ij}}{p_{1i}}$$

$$\sum_j p(j|i) = 1.$$

# Expectation Values

$$\langle x \rangle = \sum_j p_j x_j$$

$$\langle g(x) \rangle = \sum_j p_j g(x_j) \quad \text{for any function } g$$

**Definition: n<sup>th</sup> momentum**

$$\langle x^n \rangle = \sum_j p_j x_j^n$$

$$\text{cumulant: } \langle (x - \langle x \rangle)^n \rangle = \sum_j p_j (x_j - \langle x \rangle)^n.$$

$$\text{case } n=2 \text{ (variance): } \text{var}(x) = \langle (x - \langle x \rangle)^2 \rangle = \langle x^2 \rangle - \langle x \rangle^2$$

If  $x$  and  $y$  are independent random variables,

$$\langle xy \rangle = \langle x \rangle \langle y \rangle.$$

Measure of the degree of independence:

$$\text{cov}(x, y) = \langle xy \rangle - \langle x \rangle \langle y \rangle$$

## Theorem of the central limit.

If random variables  $x_1, x_2 \dots x_n$  are all independent from each other and drawn from the same distribution, the average value

$$\bar{X}_n = \frac{1}{n} \sum_{i=1}^n x_i \text{ for } n \rightarrow \infty$$

will always be distributed according to the Gaussian Distribution

$$p_G(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left[ -\frac{(x - \langle x \rangle)^2}{2\sigma^2} \right]$$

Suppose the quantity  $A$  is distributed according to a Gaussian with mean value  $\langle A \rangle$  and width  $\sigma$ . An estimator of  $\langle A \rangle$  is

$$\bar{A} = \frac{1}{n} \sum_{i=1}^n A_i$$

and the standard error of this estimate is

$$\sigma / \sqrt{n}$$



## Markov Chain

Consider a stochastic process at discrete times  $t_1, t_2, \dots$  for a system with a finite set of possible states  $S_1, S_2, \dots$

We denote  $X_t$  the state at time  $t$ .

The conditional probability that at time  $t_n$  the state is  $S_i$ :

$$X_{t_n} = S_{i_n}$$

$$P_c = P(X_{t_n} = S_{i_n} \mid X_{t_{n-1}} = S_{i_{n-1}}, X_{t_{n-2}} = S_{i_{n-2}}, \dots, X_{t_1} = S_{i_1}),$$

Definition:

A process is called Markovian if this conditional probability is independent of all states BUT the predecessor, i.e.

$$P_c = P(X_{t_n} = S_{i_n} \mid X_{t_{n-1}} = S_{i_{n-1}})$$

For a Markov process the knowledge of the state at time  $t$  determines completely the future, independently of the past.

There is NO memory.

The sequence of states is called “Markov chain”, and the conditional probability  $P_c$  is interpreted as a **transition probability** to move from state  $i$  to  $j$

$$W_{ij} = W(S_i \rightarrow S_j) = P(X_{t_n} = S_j \mid X_{t_{n-1}} = S_i).$$

$$W_{ij} \geq 0$$

$$\sum_j W_{ij} = 1$$

We define the total probability that at time  $t$  the system is in state  $S_j$ :

$$P(S_j, t)$$

The **master equation** considers the change in time of this probability

$$\frac{dP(S_j, t)}{dt} = -\sum_i W_{ji} P(S_j, t) + \sum_i W_{ij} P(S_i, t).$$

This is a “continuity equation”: change=gain-loss, and total probability is conserved

$$\sum P_{eq}(S_j, t) = 1$$

In equilibrium, the probability doesn't change in time:

$$W_{ji} P_{eq}(S_j) = W_{ij} P(S_i).$$

This condition is called “**detailed balance**”.