

1

Introduction

1.1 Computer experiments

Traditionally, physical sciences are characterized by the interplay between experiment and theory. The experiment produces numeric results by measurements upon the investigated system. The theoretical treatment implies elaboration of a model to describe the system, usually in the form of mathematical equations, which has then to be validated by its ability to describe the system behavior. Most physical problems of practical (and also academic) interest have to be simplified considerably in order to render them analytically solvable. Thus, as a result of the undergone approximations the solvable problems can significantly deviate from reality.

The appearance of high speed computers and the continuous increase of their performances has dramatically changed this picture by introducing a new approach in between experiment and theory: the *computer experiment*. A computer experiment relies on a theoretical model, too, but, in order to produce the desired results, the actual calculations are performed by a computer, which executes the operations prescribed by an *algorithm* (and coded as a program in a particular programming language). In this way, a higher degree of complexity can be introduced and more realistic systems can be investigated. Rather than attempting to obtain analytical expressions which

describe the system behavior by resorting to (sometimes severe) approximations, in a computer experiment it is possible to examine the original system.

One of the features making computer experiments distinctive from other computations is the usage of *stochastic models* (implying random variables). This makes microscopic hazard play a key role, just like in real experiments. The role attributed to the computer is different, too—it is being used as a *virtual laboratory* in which the system is studied, rather than for merely performing deterministic calculations. From this perspective, a numerical experiment is also referred to as *computer simulation*. The computation of observable quantities has many of the attributes of real measurements and the results emerging from such simulations are affected by statistical fluctuations and should be, therefore, treated by the statistical methods used in the analysis of real experiments. The *reproducibility* of the results of a computer simulation has, thus, inherently statistical character.

Molecular simulation is the generic term used for both *Monte Carlo* (MC) and *molecular dynamics* (MD) computer simulations. The primary motivation for molecular simulations is to provide exact results for statistical mechanical problems, which otherwise, with a few notable exceptions, would not be analytically solvable unless drastic approximations are introduced and the system is assumed large (essentially infinite). Molecular simulations can be regarded as *computational statistical mechanics* and they allow basically to determine macroscopic properties by evaluating exactly a microscopic theoretical model of molecular behavior and calculating averages of the physical properties of interest.

Monte Carlo simulations. The MC method is a stochastic strategy that is useful for studying systems in thermal equilibrium, for which dynamics is intrinsically absent. In order to sample the configuration space, the dynamics of the system has to be prescribed, usually in the form of an “*update algorithm*” for the microstates. In order to improve the accuracy of the simulated properties of the system in the finite allocated time, it is essential to sample with higher probability the microstates that make the most significant contributions to the averages associated with the measurable quantities. Such an approach is called *importance sampling* and the *Metropolis sampling algorithm* is one of the most famous.

Typical applications of the Monte Carlo method regard:

- Dense gases and liquids,

- Phase transitions in systems of interacting variables (spin systems), modelled, for example, by the Ising model,
- Strongly correlated quantum mechanical systems, based, for example, on the Hubbard model,
- Lattice gauge theories in particle physics, attempting the description of fundamental interactions,
- Tests of universality and finite-size scaling,
- Theories of early universe.

Molecular dynamics simulations. In contrast to the MC method, which relies on transition probabilities between microstates, in MD simulations Newton's equations of motion (Euler's equations for rigid molecules) are solved in order to generate new configurations. Sampling occurs in the whole phase space, along a continuous path. Consequently, MD simulations allow for time-dependent properties to be investigated. Relativistic effects are not considered and quantum effects, if necessary, can be incorporated as semi-classical corrections.

Typical applications relying on MD simulations are:

- Fundamental studies: molecular chaos, kinetic theory, diffusion, transport properties
- Phase transitions: phase coexistence, order parameters, critical phenomena,
- Collective behavior: space/time correlation functions, vibrations, spectroscopy,
- Complex fluids: glasses, molecular and ionic liquids, films and monolayers,
- Polymers: relaxation and transport properties,
- Solids: defects, fractures, structural transformations, mechanical properties,
- Fluid dynamics: laminar flow, boundary layers, unstable flows etc.

1.2 Relationship of MD and MC to statistical mechanics

Statistical mechanics associates with a measurable physical quantity $A(t)$ the *time average*:

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

According to the *ergodic hypothesis*, the time average for a single system is identical to the ensemble average at fixed time. An ensemble is an imaginary collection of an infinite number of identical copies of the studied system.

Within the *canonical ensemble*, having as prototype a system kept under conditions of constant temperature ($T = \text{const}$), the ensemble average is

$$\langle A \rangle = \frac{\int_{\text{phase space}} A(\mathbf{r}_N) e^{-\beta E(\mathbf{r}_N)} d\mathbf{r}_N}{\int_{\text{phase space}} e^{-\beta E(\mathbf{r}_N)} d\mathbf{r}_N},$$

\mathbf{r}_N — generalized coordinates in phase space, $\beta = 1/k_B T$

$E(\mathbf{r}_N)$ — system energy.

Corresponds to a set of measurements over an ensemble of independent systems.

MD simulation — evolution of a single system in phase space.

Time average:

$$\langle A \rangle = \frac{1}{M} \sum_{m=1}^M A(\mathbf{r}_N(t_m)),$$

M number of measurements during the evolution.

Corresponds rather to the microcanonical ensemble ($E = \text{const.}$).

MC simulation — samples only the configuration space, eliminating momenta.

Ensemble average — integrals over phase space replaced by sums over discrete microstates:

$$\langle A \rangle = \frac{\sum_s A_s e^{-\beta E_s}}{\sum_s e^{-\beta E_s}}.$$

Generating microstates **randomly**:

- **uniform probability** — waste of effort (most of the states contribute little),

- **importance sampling** — the art of MC simulations — generating microstates with probability $e^{-\beta E_s}$:

$$\langle A \rangle = \frac{1}{S} \sum_{s=1}^S A_s,$$

S — total number of sampled microstates.

The Metropolis sampling algorithm

1. Establish an initial microstate.
2. Make a random trial change in the microstate
(for example, choose a spin (particle) at random and flip (displace) it).
3. Compute the change in the energy of the system due to the trial change:

$$\Delta E \equiv E_{\text{trial}} - E_{\text{old}}.$$

4. If $\Delta E \leq 0$ accept the new microstate and go to step 6.
5. If $\Delta E > 0$ accept the new microstate with probability $w = e^{-\beta \Delta E}$:
 - (a) compute the quantity $w = e^{-\beta \Delta E}$,
 - (b) generate uniformly a random number $r \in [0.1]$,
 - (c) if $r \leq w$ accept the new microstate, otherwise retain the previous one.
6. Determine the values of the desired physical quantities.
7. Repeat steps 2 to 6 to obtain a sufficient number of microstates.
8. Periodically compute averages over microstates.

Bibliografie

- [1] Rapaport, D.C., *The Art of Molecular Dynamics Simulation*, Cambridge University Press, Cambridge, 1995.
- [2] Gould, H. and Tobochnik, J., *An Introduction to Computer Simulation Methods. Theory, Algorithms and Object-Oriented*, Addison-Wesley Publishing Company, Reading, 1996.

- [3] Sadus, R.J., *Molecular Simulation of Fluids. Applications to Physical Systems. Second Edition*, Elsevier, Amsterdam, 1999.
- [4] Ercolessi, F., *A molecular dynamics primer*, <http://www.sissa.it/furio/>