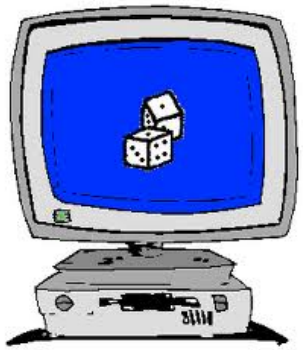
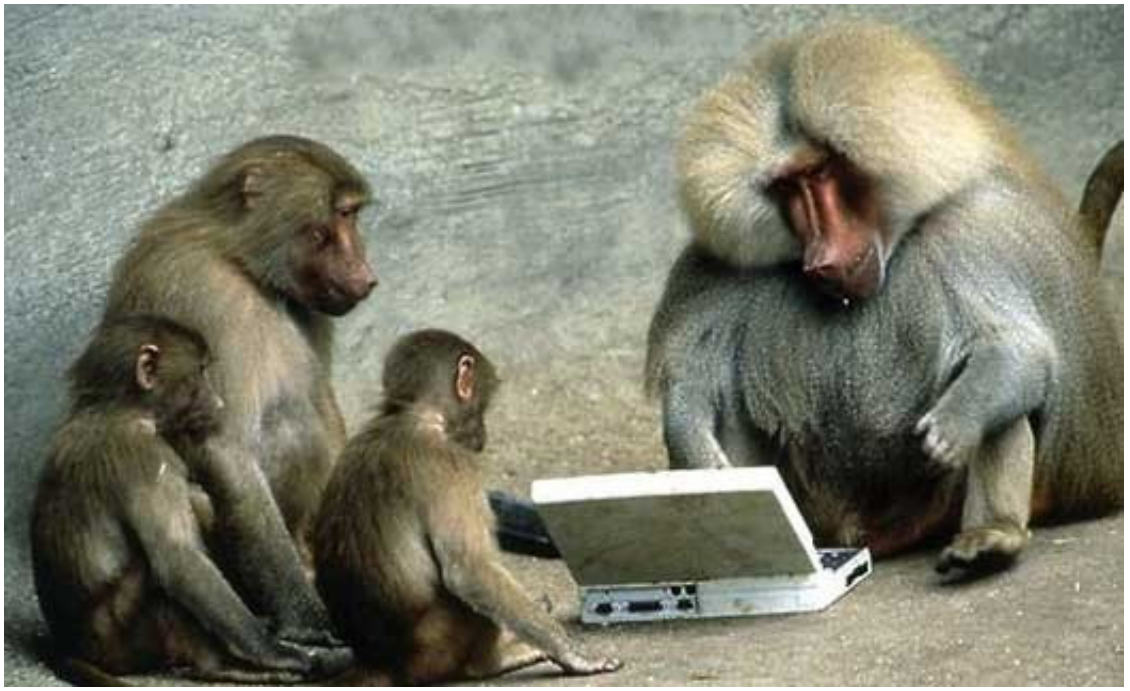
Coursework introduction

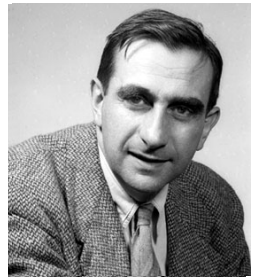
- Statistical mechanics describes the microscopic effects of thermal fluctuations.
- Most models of interest cannot be solved exactly 😔. Calculation of the partition function Z is only possible in simple cases such as when particles are weakly interacting.
- Systems with strong interactions require a **computational approach** that incorporates random thermal fluctuations



Simulation: aping nature with computers



Metropolis Monte Carlo



Edward Teller

THE JOURNAL OF CHEMICAL PHYSICS VOLUME 21, NUMBER 6 JUNE, 1953

Equation of State Calculations by Fast Computing Machines

NICHOLAS METROPOLIS, ARIANNA W. ROSENBLUTH, MARSHALL N. ROSENBLUTH, AND AUGUSTA H. TELLER,
Los Alamos Scientific Laboratory, Los Alamos, New Mexico

AND

EDWARD TELLER,* *Department of Physics, University of Chicago, Chicago, Illinois*

(Received March 6, 1953)

A general method, suitable for fast computing machines, for investigating such properties as equations of state for substances consisting of interacting individual molecules is described. The method consists of a modified Monte Carlo integration over configuration space. Results for the two-dimensional rigid-sphere system have been obtained on the Los Alamos MANIAC and are presented here. These results are compared to the free volume equation of state and to a four-term virial coefficient expansion.

THE purpose of this paper is to describe a general method, suitable for fast electronic computing machines, of calculating the properties of any substance which may be considered as composed of interacting individual molecules.

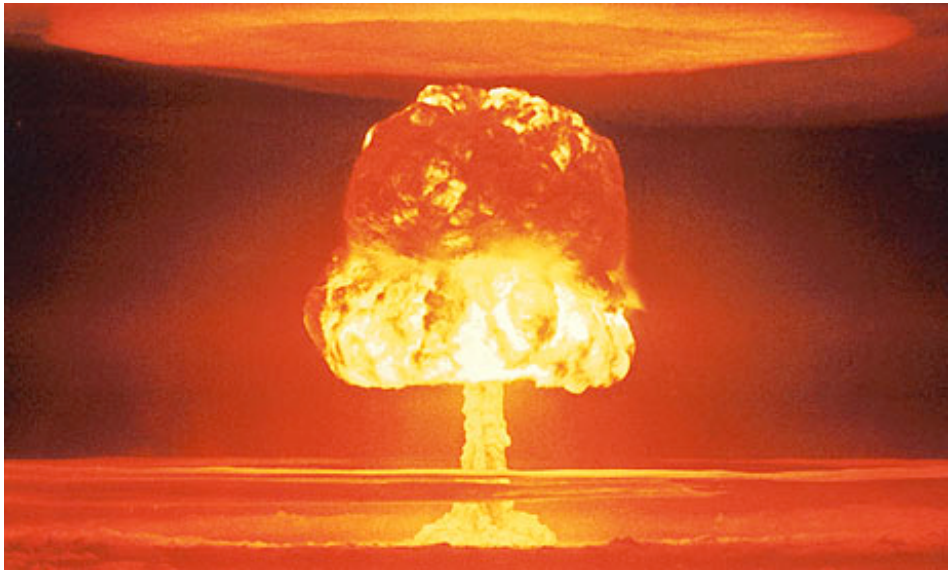
dimensional calculation for the rigid-sphere system. Work on the two-dimensional case with a Lennard-Jones potential is in progress and will be reported in a later paper. Also, the problem in three dimensions is being investigated.

the particles B, of which there is one in each of the squares which comprise the complete substance. If we have a potential which falls off rapidly with distance, there will be at most one of the distances AB which can make a substantial contribution; hence we need consider only the minimum distance d_{AB}.

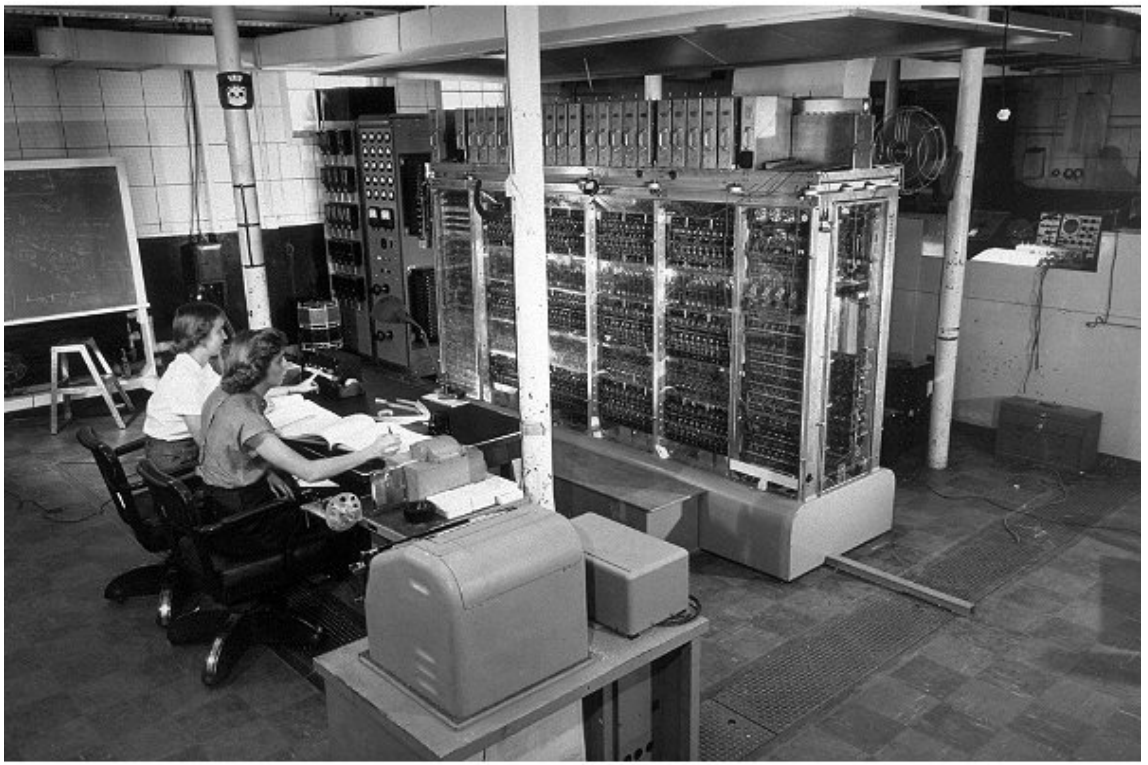
* Now at the Radiation Laboratory of the University of California, Livermore, California.

† We will use the two-dimensional nomenclature here since it is easier to visualize. The extension to three dimensions is obvious.

Los Alamos



Early days

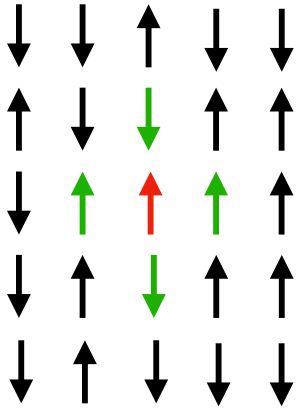


Mathematical Analyzer, Numerical Integrator, and Computer (MANIAC)

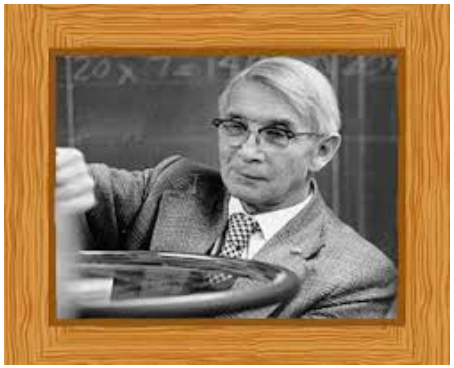
Coursework topic: two dimensional Ising model

Microstate $S = \{s_{1,1}, s_{1,2}, \ldots, s_{i,j}\}$

$s_{i,j} = \pm 1$



Each dipole (spin) interacts with its 4 nearest neighbours. Assume periodic boundary conditions.



Ernst Ising

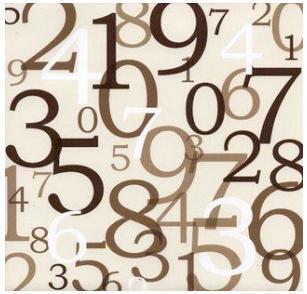
Bond energy

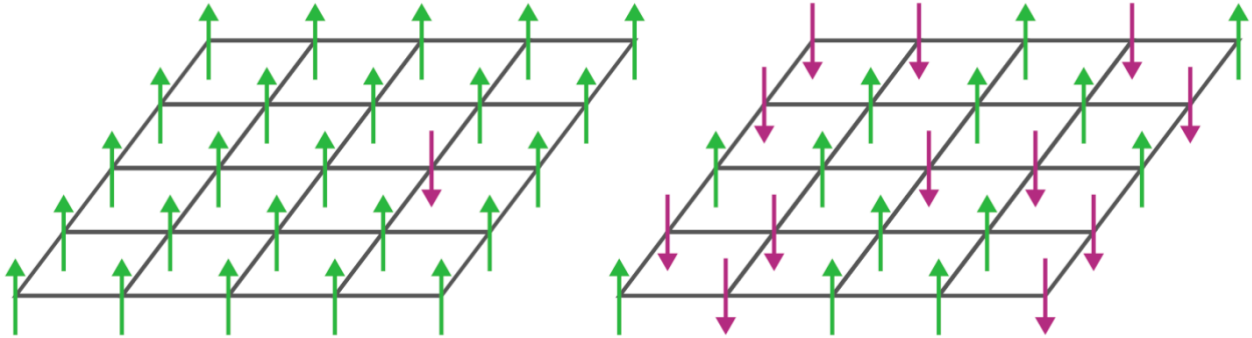
$$E(S) = -\frac{J}{2}\sum_{i,j} s_{i,j}\left(s_{i+1,j} + s_{i-1,j} + s_{i,j+1} + s_{i,j-1}\right)$$

Sum over each spin $s_{i,j}$

Sum over 4 neighbours of current spin $s_{i,j}$

Metropolis algorithm for 2d Ising model





- What Metropolis et al did was come up with a clever way of generating a set of microstates whose probabilities satisfy the Boltzmann distribution.
- This is done sequentially (`Markov chain'). Spins are `flipped' with a certain probability dependent on the associated energy change and the prescribed temperature.
- Process requires a pseudo random number generator

NB: You don’t need to understand the Ising model and Monte Carlo simulation in detail to do the coursework! However non-examinable background theory is given for those who are interested (see sec 2 of the assignment)

Coursework overview

- You will be supplied with a working Monte Carlo simulation of the two-dimensional Ising model written in Python.
- Program uses the Metropolis algorithm to generate a sequence of microstates at a user prescribed starting value of T. Calculates averages energy $\overline{E}$ and magnetisation $\overline{M}$ over this sequence.
- Repeats for a range of T and plots results.
- **Assignment**
  - run the simulation and measure various properties (think of it like a lab experiment using a prebuilt apparatus)
  - Use your Python knowledge to modify slightly what the program measures and plots.
  - Write a short skeleton report (2 sides of A4) summarising your observations
  - Submit report for grading via Blackboard (same weighting as other courseworks)
- **Further details will be released on Friday via Blackboard (see Assessment, submission and feedback tab)**

Setup

This should be completed in the week prior to the release of the assignment to make sure that any technical problems are resolved.

- On the PH20040 Blackboard page open the Resources and Tools tab
- Scroll down to Notable and open it (if off campus make sure you have the UoB VPN enabled)
- Select the Jupyter Notebook (Legacy) notebook server option
- When the notebook has opened click the +Gitrepo button
- Under enter Git Repository insert: https://github.com/nbwilding/Ising-coursework
- Press the “clone” button. This will download a notebook called Ising.ipynb into Jupyter
- Check that the program runs
- Familiarise yourself with the main features of the program. Pay attention to how to change the temperature range and system size L^2, and how the energy and magnetisation are calculated. Be aware that the program can take several minutes to run depending on the system size.

The assignment itself will be released on Friday 14th March with a deadline of Friday 21st March at 12:30.

Please contact me (nigel.wilding@bristol.ac.uk) if you have trouble with the setup described above, detailing the problem you encountered.