CS 190C
Science Education in Computational Thinking

Case Studies in Computational Physics

Physicists explain and predict the structure and behavior of physical systems by building models of them using a relatively small number of fundamental definitions and principles, things like Newton's laws of motion, Maxwell's laws of electromagnetism and so on.

They are constantly engaged in the business of testing and refining these fundamental ideas about the way in which the Universe works.

Increasingly, computational thinking plays a crucial role in the process of building working physical models.

Mark Haugan, February 25, 2008

1

Computational Thinking Facilitates the Construction of Successful Physical Models

Physical modeling imposes competing demands.

- Models must be complex

They must account for all the physics that can significantly influence the structure or behavior of the system of interest and, yet,

- Models must be simple and / or clever

We must actually be able to compute relevant aspects of structure and behavior our model of the system of interest.

Only if we are able to determine relevant properties of the model system can we possibly use our results to explain or predict the corresponding properties of the real physical system we've modeled!

2

The Challenge of Thermal Physics

We are going to use to process of modeling thermal properties of two systems, a gas and a magnetic film, to illustrate the constructive role that computational thinking can play in physics.

Macroscopic physical systems like a sample of gas or of a magnetic film consist of enormous numbers of interacting microscopic constituents. Recall, that Avogadro's number is

$$N_A \approx 6 \times 10^{23} \text{ atoms / mole.}$$

To understand the thermal properties of such systems we must account for the effects of interactions among all of their microscopic constituents. Given the complexity of the interactions and the overwhelming number of constituents in real systems, a direct approach has its limitations.

3

In principle, we could estimate the trajectory of each of the gas' particles. In practice, this is possible only for systems consisting of a relatively small number of particles.

Gas.py

For N particles:

We have N particle trajectories to predict, each of which depends on the N -1 trajectories of the other gas particles ($3N$ coupled differential equations to solve)

or,

in the ideal gas approximation, we may treat particle trajectories as having constant velocities between collisions, but we must still monitor N (N -1) pairs of particles for collisions and model each collision that occurs.

This sample program tracks 50 ideal gas molecules.

A liter same of such a gas at room temperature and pressure contains more than 2×10^{22} molecules!

4

Molecular dynamics simulations like the preceding one are sometime needed because they can give insight into the behavior of systems far from equilibrium. However, this particular simulation is being used to estimate the speed distribution of the molecules of an ideal gas in equilibrium.

A bit of physical and computational thinking will lead us to a more efficient and more accurate way to estimate equilibrium properties of macroscopic systems like a gas sample.

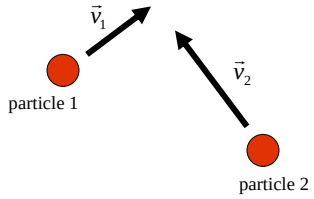
The fundamental physical principle which states that the energy of an isolated system is conserved, that is, the isolated system's energy does not change in time , plays an essential role in our new approach to modeling.

5

First, some physical thinking.

Consider a pair of molecules in a sample of ideal gas that are about to collide.

Before the collision the locations and velocities of these molecules partially characterize the gas' microscopic structure. Even when the gas is in equilibrium, that is when its macroscopic properties like pressure, volume and temperature are not changing, the gas' "microstate" is constantly changing. Collisions change molecular velocities seemly at random. Only one constraint is imposed on each of the resulting microstates. Their energies must all be the same because the energy of an isolated sample of gas is conserved.



The idea that the complicated interactions among the constituents of an isolated macroscopic system cause random transitions between all possible microstates of the system with the same energy is a powerful one.

6

The Fundamental Principle of Statistical Physics

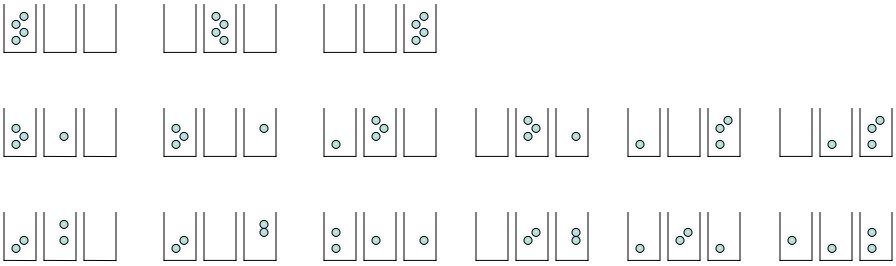
Each microstate (microscopic distribution of energy) corresponding to a given macrostate (total energy) of an isolated physical system is equally probable.

We can begin to explore the consequences of this principle by considering the microstates of a small system, one with only three simple microscopic constituents. They are simple because they can only have energies that are integer multiples of a fixed unit of energy.

Note: This kind of microscopic constituent may seem rather artificial, but there actually are systems approximately like this. They are called quantum harmonic oscillators.

7

The figure below depicts the microstates corresponding the macrostate of this system having 4 units of energy. Each unit of energy is represented by a blue dot. Each constituent is represented by a container.



Notice that there are more microstates with energy spread more evenly between the microscopic constituents. The probability of observing a microstate with one constituent having all of the energy is only 3/15 = 1/5.

8

This tendency for energy to be spread evenly among system constituents becomes very pronounced for macroscopic systems. The equilibrium properties of macroscopic systems are determined by the properties of these highly probable microstates!

A bit of computational thinking at this point leads to the idea of using a Monte Carlo method to determine a system's equilibrium properties.

Recall that a Monte Carlo method consists of making trial choices at random from a large set of possibilities and then making deductions on the basis of probabilities estimated by tabulating the results of the trial choices. You did this, for example in your percolation project.

A bit more physical and computational thinking reveals an extremely efficient way to generate random microstates of a system with (very nearly) a given macroscopic energy.

9

It turns out that we can avoid entirely analyzing, as molecular dynamics simulations do, the complicated interactions that actually cause the system's microstate to change.

We do this by adding a fictitious microscopic constituent to the macroscopic system we are interested in. For historical reasons this new constituent is called a demon. In effect, the demon mediates interactions between the system's other constituents and allows us to easily generate system microstates that all have (very nearly) the same macroscopic energy. Just what we need to apply a Monte Carlo method.

Here's how the demon does this.

10

The Demon Algorithm

Begin with the macroscopic system of interest and the demon in a state in which their total energy is equal to the energy of the desired macrostate of the system.

Let the demon "interact" with a random constituent of the system of interest, by trying to make a random change in the state of the constituent. We would, for example, try to make a random change in the velocity of an atom if the system of interest happened to be a gas.

If the trial change decreases the energy of the constituent, the change is made and the extra energy is transferred to the demon.

If the trial change increases the energy of the constituent, the change is made only if the demon has enough energy to transfer to the constituent.

In this way, the energy of the system plus demon remains constant as new microstates of the system are generated.

11

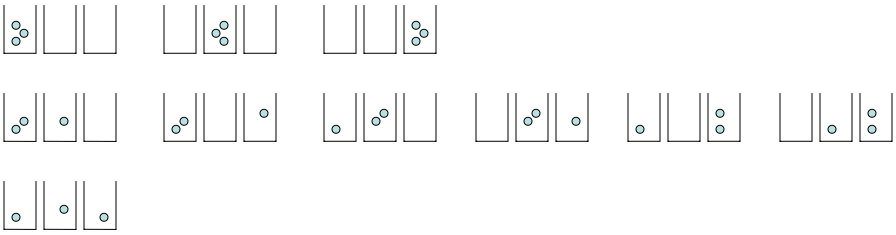
Since the Fundamental Principle of Statistical Physics predicts that a system's energy will most likely be shared equally among the demon and constituents of the original system, the demon will barely change the average energy of a macroscopic system having an enormous number of microscopic constituents.

To illustrate how the demon algorithm works, lets add a demon to the tiny system of three microscopic constituents in a macrostate with 4 units of energy, the system we considered before.

The 15 microstates of that system that we listed earlier correspond to the microstates of the combined system-demon system with zero demon energy!

12

If we try to make a change in one of those micro states that decreases the energy of a system constituent by one unit, we can do so by giving that unit to the demon. The original system would now be in one of these 10 microstates corresponding to a macrostate with only 3 units of energy.



Since there are 10 of these microstates and 15 of the ones with zero demon energy, it follows that if we make a long series of random microstate changes, the relative probability of finding the demon with zero as opposed to one unit of energy would be 15/ 10 = 3/ 2.

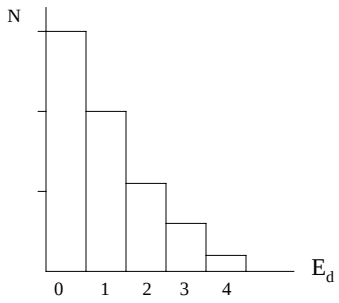
While the demon perturbs this tiny system significantly, the system is still more likely to be found with the original macroscopic energy rather than any specific lower energy.

13

Check for yourself that if the demon has 2 units of energy, there are 6 microstates of the combined system-demon system.

If the demon has 3 units of energy, there are 3 such microstates. If it has all 4 units of the energy, there is only 1 microstate of the combined system.

Together our microstate accounting results imply that if we use the demon algorithm to generate a large number of random microstates of our tiny system, the histogram below would represent the relative number of these microstates in which the demon has 0, 1, 2, 3 or 4 units of energy.



14

For a macroscopic system, the probability of generating microstates with demon energy E relative falls off exponentially relative to the probability of generating microstates with zero demon energy,

$$\frac{P(E_d = E)}{P(E_d = 0)} = e^{-E/kT},$$

where k is Boltzmann's constant and where T is the temperature of the macroscopic system.

This Boltzmann distribution governs the energy states of any microscopic constituent of a macroscopic thermal system in equilibrium.

Notice that by examining the distribution of demon energies when using the demon algorithm and the Monte Carlo method to predict the properties of a thermal system we can determine the temperature of the system as a function of its macroscopic energy.

In effect, we can use the demon as a thermometer!

15

Modeling an Ideal Gas using the Demon Algorithm

IdealDemon.py

- Possible initial states
- Generating new microstates consistent with the gas' macroscopic energy
- Accumulate the demon energy distribution
- Display the distribution of particle of velocity components in the equilibrium state

Approximations and Idealizations

- Particle energy independent of particle position
- Can treat a 3-d ideal gas a 3 independent 1-d ideal gases!

16