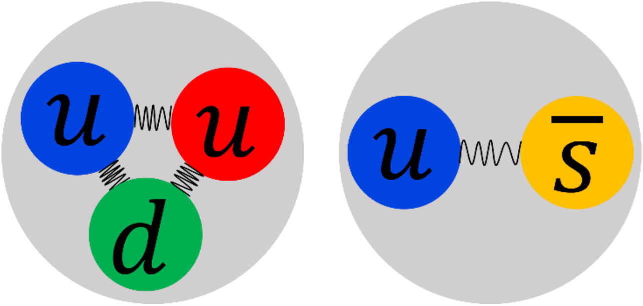
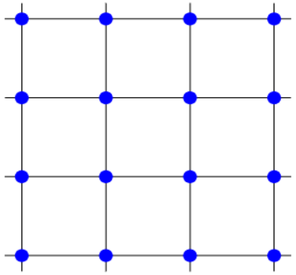
Roulette Wheels and the Fundamental Forces of the Universe

Have you ever wondered how scientists use computer simulations to better understand the dynamics of galaxies, evolutionary genetics, or the interaction of subatomic particles? In many cases, the answer is Markov Chain Monte Carlo methods (MCMC) which are, as the name suggests, to some extent based on randomness. In this article, I want to give you and introduction to one member of this group of algorithms, known as Hamiltonian Monte Carlo (HMC) and showcase an example of its application in particle physics, the context in which it was originally developed (here is the original paper).



Quark content of a Proton and a Kaon



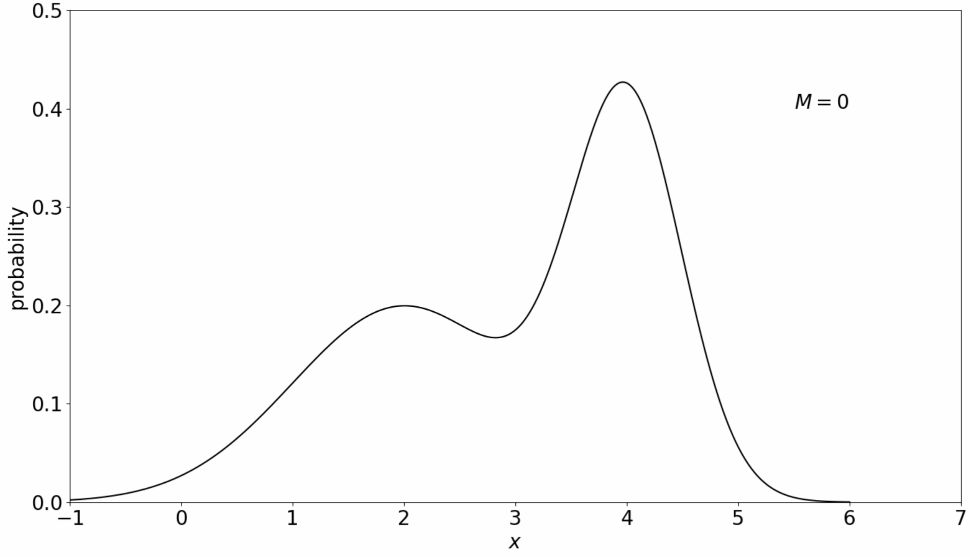
2-dimensional space lattice

The strength of quantum chromodynamics (QCD or simply the strong force) causes our already complicated theoretical calculation techniques to break down completely when studying hadrons. These are particles like protons or neutrons, but we are usually interested in their more exotic siblings such as Kaons. To make any progress, we have to turn to computers for aid, but since these can only work with discrete variables, we reduced the continuous space-time to a grid of points forming a lattice. Once the computations are finished, the results are extrapolated back to the continuum to make real-world predictions. To reduce the amount of extrapolation, a large and finely spaced lattice is required.

Since the simulations of QCD on a lattice are still incredibly complicated, it is common to consider theories with similar physics while being computationally simpler. These are usually called toy models. I studied such model and discover that a simple modification of the HMC algorithm can increase the simulation efficiency by up to a factor of 300!
But before we get to that, we need to address the elephant in the room: “What are Markov Chain Monte Carlo methods and how do they work?.”

Markov Chain Monte Carlo

Most problems can be phrased in a statistical manner. In fact, all of Quantum mechanics is probabilistic. We never know the state of a particle with certainty but can only assign a probability to it. For almost all systems, this probability is next to impossible to work out, but MCMC gives us a recipe for approximating it.
The basic idea is that you can estimate the probability distribution by a set of samples



Approximating a probability distribution (solid line) by making a histogram as increasingly many samples are collected.

drawn from it. The more samples you collect, the better the approximation as illustrated by the animation for some generic random variable x.

There are two steps for generating the chain of samples in which each element is influenced by the one immediately preceeding it. First, a candidate is proposed based on the current sample. Secondly, the candidate is appended to the chain if it is more likely to occur under the distribution than the current sample. In the opposite case, the current sample gets repeated, giving it a greater weight in the chain. There are a few technical points that the proposal and accept/reject rules need to fulfil to guarantee that the collected samples will represent the distribution well. You can find more details here.

Hamiltonian Monte Carlo

Now let's have a look at what the specific rules for HMC are. The current state of the system can be described by some number of independent parameters, known as the degrees of freedom. The key idea is to interpret these parameters as coordinates of a cluster of gas molecules, and by assigning them a random momentum, we can make them move. These multiparticle dynamics follow Newton's law, but describing them through the Hamiltonian, which you can imagine as the energy of the system, is much more efficient. If we let this evolution happen for some time and then map the resulting arrangement of the gas back to the parameter values of the system, we have produced a candidate sample. The accept or reject decision is then performed using the Metropolis update: If the candidate lowers the energy, we will always accept it. Should it instead increase it, we will only accept it in rare cases to assure that all possible configurations of the system are explored. An explicit mathematical expression for this decision is given in the below summary of the HMC steps:

1. Start the chain with some arbitrary sample.
2. Assign each component of the system a random momentum (usually in the form of a gaussian random variable).
3. Compute the Hamiltonian H_{old} of the system.
4. Evolve the molecular dynamics.
5. Compute the Hamiltonian H_{new} of the system in this new state.
6. Accept the new state as a sample with probability $min(1, e^{-H_{new}+H_{old})})$.
7. Return to step 2.

Critical Slowing Down

But there is a problem. To get an accurate representation of the probability distribution, all of the collected samples must be statistically independent. Considering the opposite extreme, in which the chain is just a repetition of a single sample, makes this clear as it would result in a highly inaccurate histogram approximation.
On average, two statistically independent samples are separated by τ steps in the chain (statisticians call this the integrated autocorrelation time), such that out of the total M simulation steps, only M/τ can be used as data points.

In lattice simulations, this problem becomes increasingly severe as a more finely spaced lattice is used. This means that an increasing amount of computing power is wasted on calculating samples which will not be used in the data analysis. For a lattice with zero spacing, it would take an infinite amount of time to get only one data point, which is known as critical slowing down. For the toy model I considered, HMC required

6569		1
simulation steps	for	data point

Luckily, we can do better than that.
The origin of critical slowing down lies in the drastically different evolution rates of the system's constituents during the molecular dynamics. These constituents are commonly called modes. Since the momenta introduce in step 2 are arbitrary, we can modify them in a very particular way to achieve a more equal evolution. We slow down rapidly evolving modes by assigning them a large mass, while slow modes are sped up through small masses. These manipulations make simulating the molecular dynamics more difficult, but with the help of a commonly used tool in physics, the Fourier transform, they can be easily disentangled. Because of this, the modified algorithm is called Fourier accelerated HMC and only required

21 simulation steps for **1** data point

Much better! In fact, better by a factor of 310! In other words, employing Fourier acceleration makes the simulation roughly 300 times faster without losing accuracy in the final data analysis. For a lattice even closer to the continuum, this improvement continues to grow.

By testing simulation techniques on these simpler toy models, we hope to gain insight into how to enhance QCD simulations and ultimately improve our understanding of how nature works on the smallest scales.

To learn more about my project, I invite you to check out the full simulation code and data here.