Project 4: Ising Annealing

July 26, 2020

Motivation

If we go back to a couple of weeks ago to Projects 1 and 2, you were able to simulate molecules in a couple of different ways: using machine learning (Restricted Boltzmann machines) and using the Variational Quantum Eigensolver (VQE). In Project 1, you were able to see that machine learning gives a huge advantage in data-driven problems in terms of outputting a compressed version of something quite large (exponential scaling – 2^{100} – versus a few hundred numbers). Not only this, you saw that if we needed to collaborate with an experimentalist in a lab, machine learning doesn't demand much from experimentalists (we don't need that much data from them). In Project 2, you saw that solving chemistry problems via simulation on a quantum circuit also has compression benefits, but we didn't need any input data to actually solve our problem like in Project 1. Rather, the *variational method* just requires knowledge of the Hamiltonian, and a tunable *trial solution* to our problem that we then change based on how good of an energy it gives.

At the end of the day, VQE and machine learning are both extremely useful in different settings, but at the heart of each is an *optimization problem*.[1] Sometimes this optimization problem can be extremely difficult to tackle. This week, we'll re-cast the problem into one that can be explored with the well-known simulation method called Monte Carlo (MC). At the end of a Monte Carlo simulation we do not obtain a compressed version of something much larger (like the wavefunction). Rather, we use a Monte Carlo procedure to produce samples, correctly distributed, from a thermal state.

Thermal annealing exploits thermal fluctuations to discover the groundstate of a system. In a Monte Carlo simulation the temperature can be slowly decreased. If this decrease is slow enough the system remains close to equilibrium, and at the end of annealing the typical configurations encountered will be those typical of the system in its groundstate.[2]

Monte Carlo simulations can be *very, very* efficient and fast.[3] This week, your tasks will involve the use of a single-spin-flip Metropolis-Hastings (MH) MC simulation. Let's dive into your tasks!

Your Tasks

This week, you will be tasked with performing various MC simulations on the Ising model,

$$H = -J \sum_{\langle i,j \rangle} \sigma_i \sigma_j, \tag{1}$$

[1] In machine learning we must optimize the cost function and in VQE we must optimize the energy.

[2] Of course, in addition to thermal fluctuations, one can envision the use of quantum fluctuations to accelerate the exploration of the configuration space in a similar manner

[3] You now see the daily struggles of computational scientists: which algorithm or method do I use to solve my problem?!

where σ_i are classical spin-1/2 variables (-1 or 1), and $\sum_{\langle i,j \rangle}$ denotes summation over nearest neighbour pairs. You will be walked through a simulation we've provided along with an annealing procedure. Then, you'll be given slight variations to the Ising model where you must come up with an annealing procedure yourself!

Task #1

Navigate to the `Task_1.ipynb` notebook. We've given you a full solution to a MC simulation employing the MH algorithm for a 2D ferromagnetic ($J > 0$) Ising Model on a square lattice with periodic boundary conditions. Code blocks 1 to 5 outline a MC simulation for this system at temperature $T = 1.0$. After this, we now wish to perform a simulation at a temperature $T_{\text{final}} = 0.01$ using an annealing procedure starting at $T_{\text{initial}} = 100$. Here, we employed an annealing procedure that decreases T_{initial} exponentially (see code block 7). Run this notebook and make sure that you understand what is going on under the hood (see `abstract_ising.py` and `ising_animator.py`).

Task #2

Above, we used an exponential annealing "schedule". However, there are many other functional forms that you could use to define the annealing procedure (e.g. linear). In this task you should explore different possibilities, to see what works best (by giving the lowest energy at the end of the anneal). Navigate to `Task_2.ipynb`. Here's what we'd like you to do!

1. Having understood Task #1, we're now going to task you with finding an annealing procedure for the random-bond Ising model in 1D,

$$H = J \sum_{\langle i,j \rangle} B_{ij} \sigma_i \sigma_j, \tag{2}$$

   where $B_{ij} = \pm 1$ are selected randomly at the start. Do this for various system sizes (10, 20, 50, 100).

2. The fully-connected random bond Ising model

$$H = J \sum_{i<j} B_{ij} \sigma_i \sigma_j, \tag{3}$$

   is even a little harder to perform a good MC simulation for low temperatures. Find an annealing procedure for this model, as well.

We've given you the required classes in order to get your started on these models in `Task_2.ipynb`. In your annealing procedures, start with as high of a starting temperature as needed and get to as low of a temperature as you can!

Task #3

This is where things get fun! We'll be applying what we've learned so far to find the ground state energy of the Hydrogen molecule at various separation lengths.

There are several methods with which one can map the electronic structure Hamiltonian of the Hydrogen molecule to a classical Ising Hamiltonian [1, 2, 3, 4] (all of these papers are available on the arXiv). The Ising Hamiltonians we'll be handling here were produced using the *Iterative Qubit*

Coupled Cluster method [1]. This method in fact produces *Generalized Ising Hamiltonians*, that is, Ising Hamiltonians with k-local interactions (where $k > 2$). In the case of the Hydrogen molecule, we need only 4-spins to encode the ground state and hence, our Generalized Ising Hamiltonian has k-local interactions upto $k = 4$.

$$H = E_0 + \sum_i h_i \sigma_i + \sum_{ij} J_{ij} \sigma_i \sigma_j + \sum_{ijk} K_{ijk} \sigma_i \sigma_j \sigma_k + \sum_{ijkl} L_{ijkl} \sigma_i \sigma_j \sigma_k \sigma_l$$

The arrays $J_{ij}, K_{ijk}, L_{ijkl}$, encode the k-point interaction strengths between different Ising spins, h_i encodes the external magnetic field applied to the ith spin, and E_0 is a constant energy shift which is irrelevant for the Monte Carlo simulation but necessary for computing the energy of the Hydrogen molecule.

Our goal is to perform an annealed Monte Carlo simulation of this Hamiltonian in order to find the ground state energy of H_2.

1. In `Task_3.ipynb`, you'll find the function `read_generalized_ising_hamiltonian` which will provide you with the Hamiltonian parameters above. Below that, you'll find a skeleton for a `GeneralizedIsingModel` class which you must complete. With this, you'll perform a Monte Carlo simulation at a fixed temperature for 1000 steps.

2. Next, you'll devise an annealing procedure (or use one of the ones you came up with previously) to find the ground state of the system.

3. As we only have 4 Ising spins, we can compute the ground state energy exactly. Do this by iterating over all possible spin configurations, computing the energy for each, and find the ground state. Compare this energy to the one you obtained using annealing.

4. Last, you'll repeat the above for all the different Hydrogen separation lengths provided.

Further Challenges

1. In `Task_2.ipynb`, we also gave you some code to get you started on determining an annealing procedure for the fully-connected Mattis model. Try finding an annealing procedure for this model at various system sizes.

2. Many existing quantum annealers don't support k-local interactions for $k > 2$. Reduce the H_2 generalized Ising Model from Task #3 to a 2-local Hamiltonian (see [4]) and repeat Task #3 for this 2-local Hamiltonian. Compare the energies you get from annealing this Hamiltonian to the energies you got from applying annealing to the original k-local Hamiltonian. Does it seem to be more or less difficult to find the ground state with classical thermal annealing?

3. If you have a trustworthy 2-local Hamiltonian from the above challenge, compare your home-grown thermal annealing procedure to open-source commercial software (e.g. D-Wave Leap). Benchmark performance. Compare and contrast any differences in annealing approaches.

4. Map your favorite NP-complete Hamiltonian to an Ising Hamiltonian, and perform a thermally-annealed MC simulation to try to solve the problem. See [5] for some inspiration. Benchmark against commercially available open-source optimization software.

Acknowledgements

We thank the authors of [1] (particularly Ilya and Scott) for their help with Task #3 and for providing the Generalized Ising Hamiltonians for H_2.

References

[1] Ilya G. Ryabinkin, Robert A. Lang, Scott N. Genin, and Artur F. Izmaylov. Iterative qubit coupled cluster approach with efficient screening of generators. *Journal of Chemical Theory and Computation*, 16(2):1055–1063, 2020. PMID: 31935085.

[2] Ilya G. Ryabinkin, Tzu-Ching Yen, Scott N. Genin, and Artur F. Izmaylov. Qubit coupled cluster method: A systematic approach to quantum chemistry on a quantum computer. *Journal of Chemical Theory and Computation*, 14(12):6317–6326, 2018. PMID: 30427679.

[3] Rongxin Xia, Teng Bian, and Sabre Kais. Electronic structure calculatins and the ising machine. *arXiv preprint arXiv:1611.01068*, 2016.

[4] Rongxin Xia, Teng Bian, and Sabre Kais. Electronic structure calculations and the ising hamiltonian. *The Journal of Physical Chemistry B*, 122(13):3384–3395, 2017.

[5] Andrew Lucas. Ising formulations of many np problems. *Frontiers in Physics*, 2:5, 2014.

[6] M. E. J. Newman and G.T. Barkema. *Monte Carlo Methods in Statistical Physics*. Clarendon Press, 1998.