#27 Add Qiskit Nature demonstrations for physics problems

Qiskit Advocate Mentorship Program - Fall 2021

Mentors: Steve Wood and Soham Pal

Mentees: Siddhartha and José Victor

Summary

Qiskit

Introduction and Project Description

Team

Goals

Project Timeline

“Nature isn’t classical, dammit, and if you want to make a simulation of nature, you’d better make it quantum mechanical, and by golly it’s a wonderful problem, because it doesn’t look so easy.”

Richard Phillips Feynman

Qiskit

Introduction and Project Description

Qiskit

Quantum Computing can solve a variety of problems:

- Simulation of physical systems
- Optimization problems
- Quantum chemistry
- Machine learning, Finance, etc

Feels like we are missing more demonstrations of physical problems

Qiskit Nature aims to solve problems in physics, chemistry and biology.

Although, most of its tutorials are for solving chemistry problems, and they are very much tied to the use of molecular Hamiltonians, and quantum chemistry frameworks like PySCF.

There are a bunch of physical problems whose solution reduces to find the ground state of some system. Thus, it can be solved using Qiskit variational methods like **VQE**. Our goal is to broaden the type of problems that can be implemented in Qiskit Nature.

Qiskit

Probing ground state properties of the kagome antiferromagnetic Heisenberg model using the Variational Quantum Eigensolver

Jan Lukas Bosse[1,2,*] and Ashley Montanaro[2,1,†]

[1] *School of Mathematics, University of Bristol*
[2] *Phasecraft Ltd.*

(Dated: October 5, 2021)

Finding and probing the ground states of spin lattices, such as the antiferromagnetic Heisenberg model on the kagome lattice (KAFH), is a very challenging problem on classical computers and only possible for relatively small systems. We propose using the Variational Quantum Eigensolver (VQE) to find the ground state of the KAFH on a quantum computer. We find efficient ansatz circuits and show how physically interesting observables can be measured efficiently. To investigate

Cloud Quantum Computing of an Atomic Nucleus*

E. F. Dumitrescu,[1] A. J. McCaskey,[2] G. Hagen,[3,4] G. R. Jansen,[5,3] T. D. Morris,[4,3] T. Papenbrock,[4,3,†] R. C. Pooser,[1,4] D. J. Dean,[3] and P. Lougovski[1,‡]

[1] *Computational Sciences and Engineering Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831, USA*
[2] *Computer Science and Mathematics Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831, USA*
[3] *Physics Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831, USA*
[4] *Department of Physics and Astronomy, University of Tennessee, Knoxville, TN 37996, USA*
[5] *National Center for Computational Sciences, Oak Ridge National Laboratory, Oak Ridge, TN 37831, USA*

We report a quantum simulation of the deuteron binding energy on quantum processors accessed via cloud servers. We use a Hamiltonian from pionless effective field theory at leading order. We design a low-depth version of the unitary coupled-cluster ansatz, use the variational quantum eigensolver algorithm, and compute the binding energy to within a few percent. Our work is the first step towards scalable nuclear structure computations on a quantum processor via the cloud, and it sheds light on how to map scientific computing applications onto nascent quantum devices.

Team

Qiskit

Steve

Slack: @Steve Wood

Soham

Slack: @Soham Pal

Siddhartha

Slack: @Siddhartha Morales

Mexican

Master student at IFT/ICTP UNESP

Interests: Theoretical physics and machine learning

José Victor

Slack: @José Victor Soares Scursulim

Brazilian

Master Student at Federal University of Espírito Santo (UFES)

Interests: Quantum Simulation, Quantum Error Correction and Quantum Machine Learning

Goals

- Demonstrate how to use Qiskit Nature tools and features in physics problems
  - Develop tutorials for https://arxiv.org/abs/1801.03897 and https://arxiv.org/abs/2108.08086.

- Show that the scope of applications of Qiskit Nature can be expanded

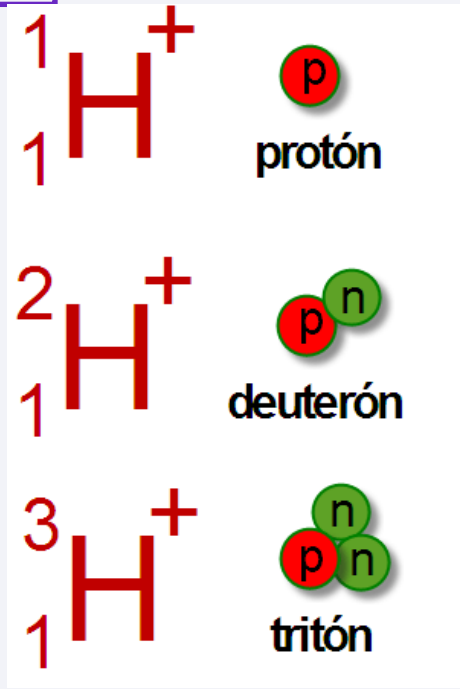
- Point features that could be implemented or modificated in **Qiskit Nature**

Deuteron

$$H_N = \sum_{n,n'=0}^{N-1} \langle n'|(T+V)|n\rangle a_{n'}^{\dagger} a_n.$$

Effective Hamiltonian → Second Quantization → Hamiltonian mapping → Variational Algorithms

$$\begin{aligned} H_2 &= 5.906709I + 0.218291Z_0 - 6.125Z_1 \\ &\quad - 2.143304\,(X_0X_1 + Y_0Y_1)\,, \\ H_3 &= H_2 + 9.625(I - Z_2) \\ &\quad - 3.913119\,(X_1X_2 + Y_1Y_2)\,. \end{aligned}$$

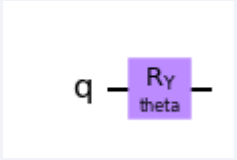


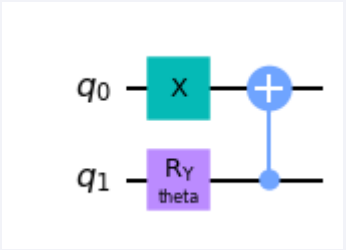
Deuteron

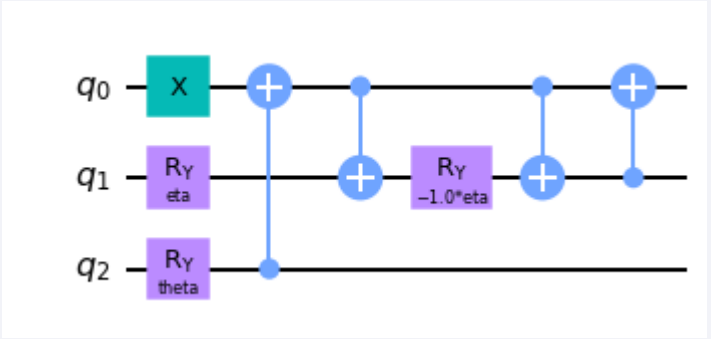
We want to calculate the energy of the ground state of deuteron using Qiskit Nature tools.

We can use Operator Flow to write the Hamiltonian of the problem.

Ansatz circuits and a piece of code that shows the results of VQE








```
#defining the Hamiltonians
h1 = (0.218291*Z)-(0.218291*I)
h2 = (5.906709*I^I) + (0.218291*I^Z) - (6.125*Z^I) - (2.143304*X^X) - (2.143304*Y^Y)
```

```
op1 = ~StateFn(h1) @ StateFn(wavefunction)
op2 = ~StateFn(h2) @ StateFn(wavefunction2)
```

```
vqe = VQE(wavefunction, optimizer=cobyla, quantum_instance=quantum_instance)
result = vqe.compute_minimum_eigenvalue(h1)
print('Result:', np.round(result.optimal_value,3))
```

```
Result: -0.437
```

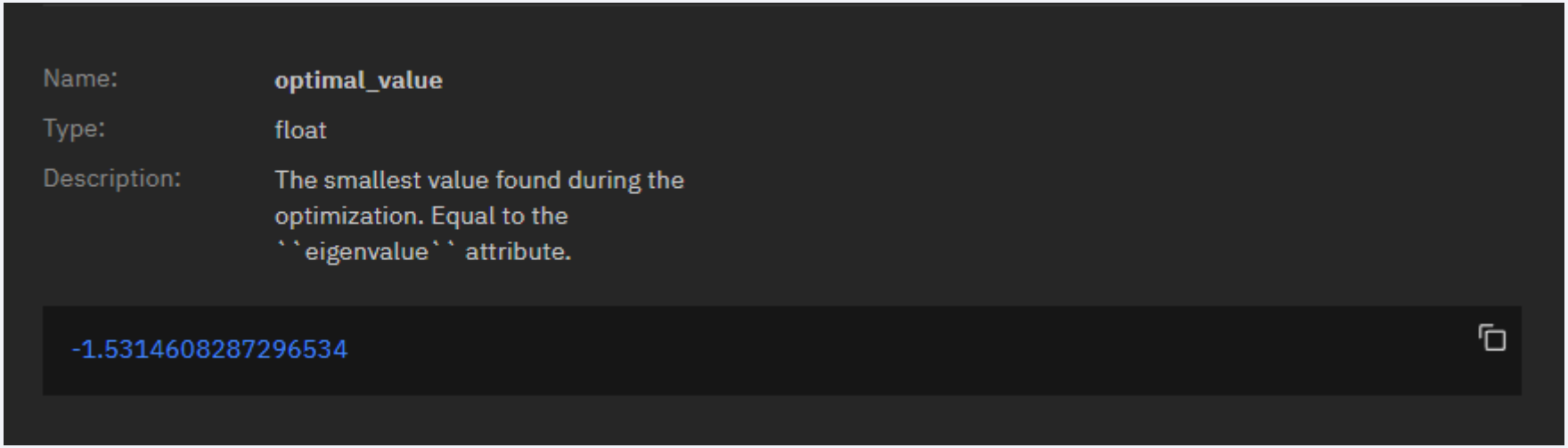
```
vqe2 = VQE(wavefunction2, optimizer=cobyla, quantum_instance=quantum_instance)
result2 = vqe2.compute_minimum_eigenvalue(h2)
print('Result:', np.round(result2.optimal_value,3))
```

```
Result: -1.749
```

Qiskit Runtime

VQEProgram

With Qiskit Runtime we can use the function VQEProgram present in Qiskit Nature to execute a VQE in real quantum devices. Below there is a print screen with the result for the Hamiltonian h2 defined in the piece of code presented in the previous slide. In this example, we used SPSA as the optimizer and we got a result with relative percentual error of approximate 13% in comparison with the reference value -1.79 MeV obtained through statevector

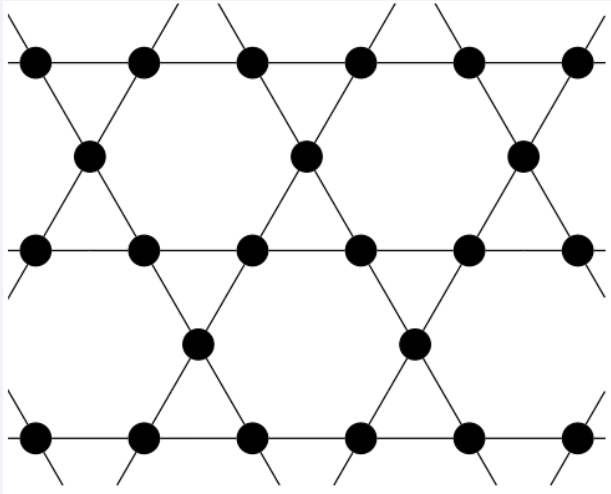


Further work

- Compute the energy and ground state for higher order Hamiltonians H_N
- Compute relevant observables (Z0, Z1, X0X1 and Y0Y1) and physical correlators
- Test and compare different optimizers and ansatzes.

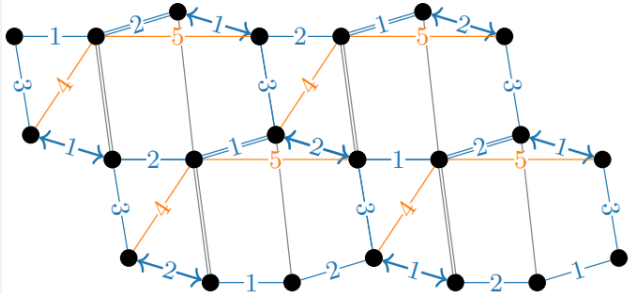
Kagome antiferromagnetic Heisenberg model

$$H = \sum_{\langle i,j \rangle} \vec{S}_i \cdot \vec{S}_j = \sum_{\langle i,j \rangle} X_i X_j + Y_i Y_j + Z_i Z_j$$



We want to find the ground state, since it's a really intricate problem due to the lattice configuration.

$$U(\theta) = \prod_{i=1}^{p} \left[e^{-i\theta_{i,0} H_0} \prod_{\langle k,l \rangle} e^{-i\theta_{i,kl} \vec{S}_k \cdot \vec{S}_l} \right]$$



1. Let all —1—, ↖1↘ and ═1═ connected qubits interact
2. Let all —2—, ↖2↘ and ═2═ connected qubits interact
3. Let all 3 connected qubits interact
4. Swap all ↖1↘ and ↖2↘ connected pairs of qubits
5. Let 4 connected qubits interact via by ∥
6. Let —5— connected qubits interact via by ═1/2═
7. Swap ↖1↘ and ↖2↘ back for initial configuration

arxiv:2108.08086v1

Project Timeline

Tasks planning for the next 3 months

October - 2021

- Focus on the deuteron problem
- Computer higher order terms
- Explore Qiskit Nature implementations
- Move into the next problem

November - 2021

- Tackle the Heisenberg model
- Explore different lattice configurations
- Find the ground state and other observables
- Generalize the work and try to put it into Qiskit Nature.

December - 2021

- Colect the relevant result and presente them as open notebooks for everyone

Thank you!

Questions?