

Computing Rigid Body Rotations with Qiskit

#7

Problem Definition



This project aims to study the feasibility of using current Quantum Computing devices to implement a method to compute the Euler-Rodrigues Parameters (ERP). These parameters allow for defining a transformation matrix to calculate rigid body rotations

```
from qiskit import QuantumCircuit, execute
from qiskit import Aer, IBMQ
from qiskit.providers.aer.noise import NoiseModel

# Choose a real device to simulate from IBMQ provider
provider = IBMQ.load_account()
backend = provider.get_backend('ibmq_vigo')
coupling_map = backend.configuration().coupling_map

# Generate an Aer noise model for device
noise_model = NoiseModel.from_backend(backend)
basis_gates = noise_model.basis_gates

# Generate 3-qubit GHZ state
num_qubits = 3
circ = QuantumCircuit(3, 3)
circ.h(0)
circ.cx(0, 1)
circ.cx(1, 2)
circ.measure([0, 1, 2], [0, 1, 2])

# Perform noisy simulation
backend = Aer.get_backend('qasm_simulator')
job = execute(circ, backend,
              coupling_map=coupling_map,
              noise_model=noise_model,
              basis_gates=basis_gates)
result = job.result()

print(result.get_counts(0))
```

Team



- Abhay Kamble
- Anuranan Das
- Emilio Pelaez
- Hojun Lee
- Parmeet Singh Chani
- Ran-Yu Chang
- Daniel Sierra-Sosa

Euler-Rodrigues Formula



This formula was first discovered by Euler and later rediscovered independently by Rodrigues, hence the terms Euler parameters for the parameters a, b, c, d and Euler–Rodrigues formula for the rotation matrix in terms of a, b, c, d [1].

$$\begin{bmatrix} a^2 + b^2 - c^2 - d^2 & -2ad + 2bc & 2ac + 2bd \\ 2ad + 2bc & a^2 - b^2 + c^2 - d^2 & -2ab + 2cd \\ -2ac + 2bd & 2ab + 2cd & a^2 - b^2 - c^2 + d^2 \end{bmatrix}$$

where $a^2 + b^2 + c^2 + d^2 = 1$

Alternatively,

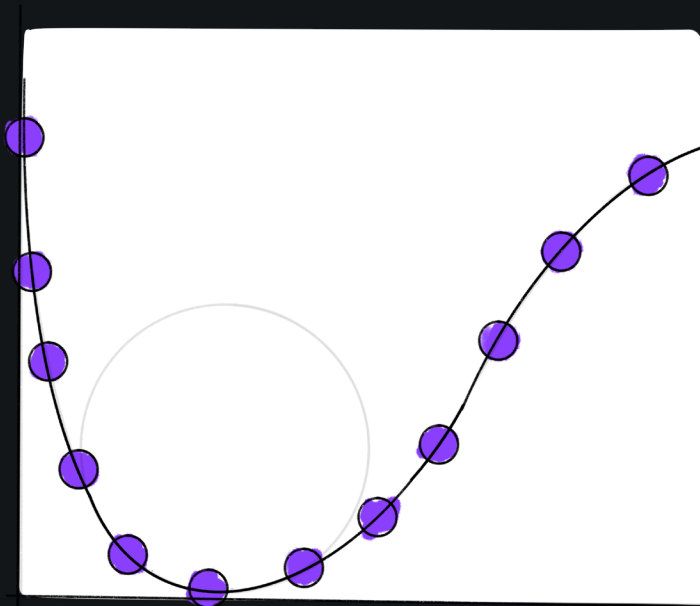
$$\vec{x}_{rot} = \vec{x} \cdot \cos(\theta) + (\vec{k} \times \vec{x}) \sin(\theta) + \vec{k}(\vec{k} \cdot \vec{x})(1 - \cos(\theta))$$

where \vec{k} rotation axis and θ the rotation angle.

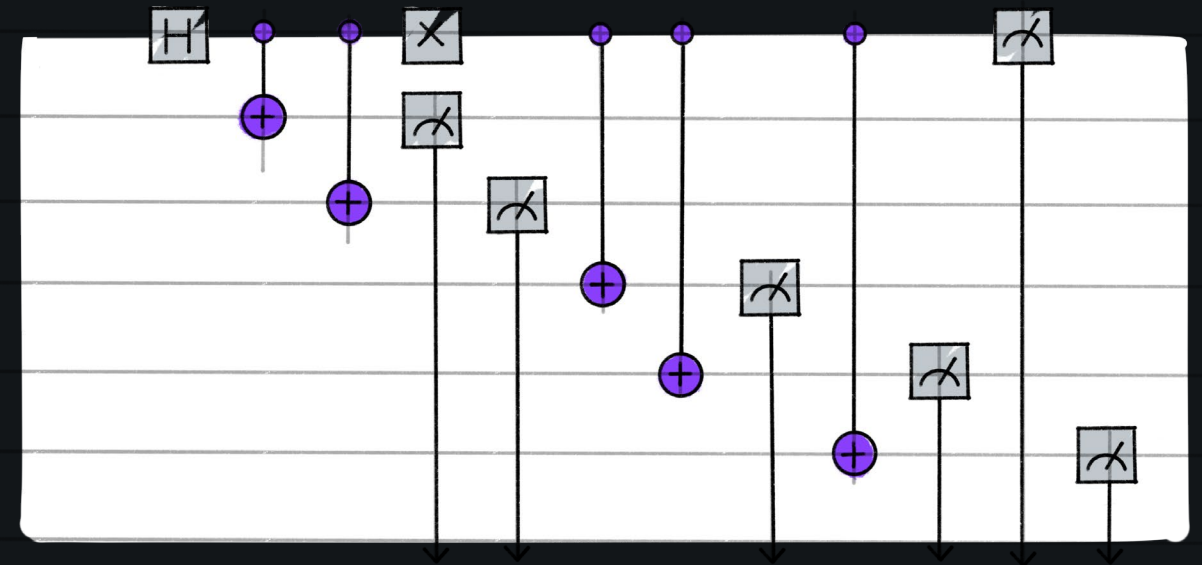
[1] Mebius, J. E. (2007). Derivation of the Euler-Rodrigues formula for three-dimensional rotations from the general formula for four-dimensional rotations. *arXiv preprint math/0701759*.

Proposed Solutions

A Variational approach



A Circuit approach



A Variational approach

Hybrid Quantum-Classical Neural Networks [3] will be used for rotating the initial state(initial position) to the final state, after rotation through the euler angles (α, β, γ) .

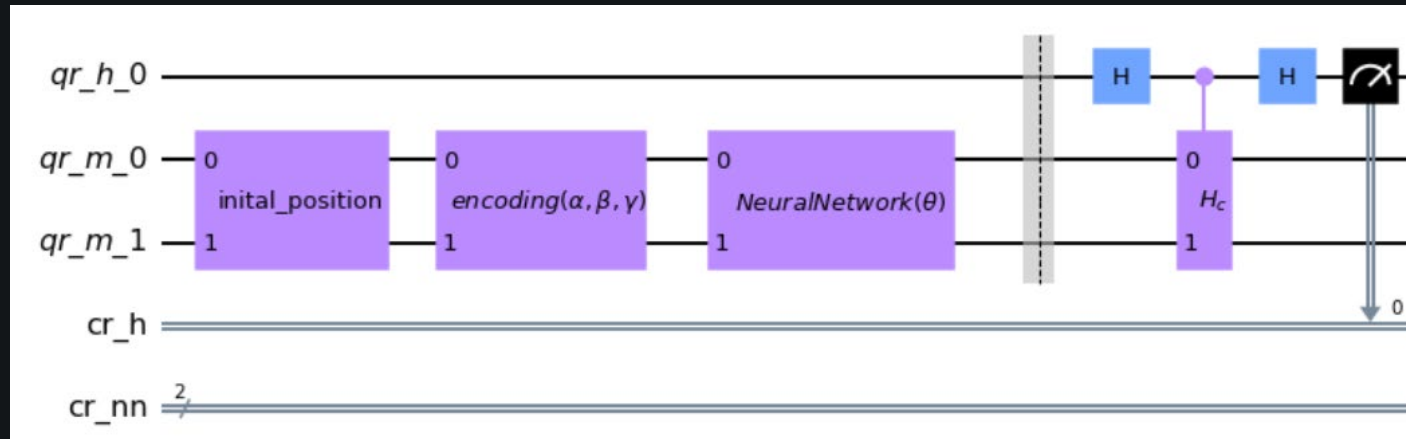
Variational encoding will be used to encode the euler angles.

The cost hamiltonian, will be used to give a measure of how far the state generated from our circuit is to the original true state. It is defined as:

$$H_c = I - |gt\rangle\langle gt|$$

where $|gt\rangle$ is the final true state.

Hadamard test will be used to measure the expectation value of our cost Hamiltonian



[3] Xia, Rongxin, and Kais, Sabre. Hybrid Quantum-Classical Neural Network for Calculating Ground State Energies of Molecules. United States: N. p., 2020. Web. <https://doi.org/10.3390/e22080828>.

[4] Bravo-Prieto, Carlos et al. "Variational Quantum Linear Solver: A Hybrid Algorithm for Linear Systems." Bulletin of the American Physical Society (2020)

A Circuit Approach

We are looking for an efficient way of solving the cross product that defines the rotation matrix, which could be done by defining the matrix in a block encoding way such as described in [2]. A reformulation of the problem could be finding the parameters that satisfy the quaternion definition for the same rotation, which could be more efficiently implementable by using the group $SU(2)$ to represent three-dimensional rotations in a 2 by 2 matrix.

$$U = \begin{pmatrix} a + di & b + ci \\ -b + ci & a - di \end{pmatrix} = a \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + b \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} + c \begin{pmatrix} 0 & i \\ i & 0 \end{pmatrix} + d \begin{pmatrix} i & 0 \\ 0 & -i \end{pmatrix}$$
$$= aI + ic\sigma_x + ib\sigma_y + id\sigma_z$$

[2] Chakraborty, S., Gilyén, A., & Jeffery, S. (2018). *The power of block-encoded matrix powers: improved regression techniques via faster Hamiltonian simulation.* *arXiv preprint arXiv:1804.01973.*

