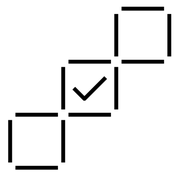
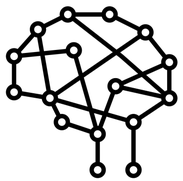
Project Idea - Quantum Othello with Q-Reinforcement Learning

Games are an interesting test bed for artificial intelligence research, as they provide a self-contained environment with fixed rules. DeepBlue, Watson and AlphaGo are only a few examples of algorithms that were put through their paces in games before applying them to different problems.

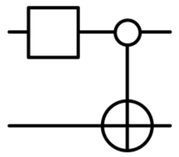


Othello is a perfect information, zero-sum, two-player strategy game played on an 8x8 board, and has already been used in classical artificial intelligence research. The board stages are highly volatile, each new move can change a large area of the board. Despite its simple rules, the game of Othello is not trivial, containing of approximately 10^{28} legal positions. The game tree itself has approximately 10^{58} nodes.



We propose the implementation of a Quantum Othello game using quantum computing together with classical machine learning techniques to create a (self-improving) computer opponent players can compete against.

Othello is also seen as a Markov Decision Problem in reinforcement learning. In addition, the mixed application of Convolutional Neural Networks result in a better accuracy predicting moves.



The Quantum opponent creates winning strategies using a Variational Quantum Circuit for Deep Reinforcement Learning. The implementation will utilise PyTorch to train a Deep Q-Learning neural network with a Quantum Computing based hidden layer.

Impact

Convolutional Neural Networks in combination with quantum mechanics are novel to this approach where the game state and action space is large.

Provide an analysis of encoding the state of a complex game in a format suitable for usage in current Quantum Machine Learning environments.

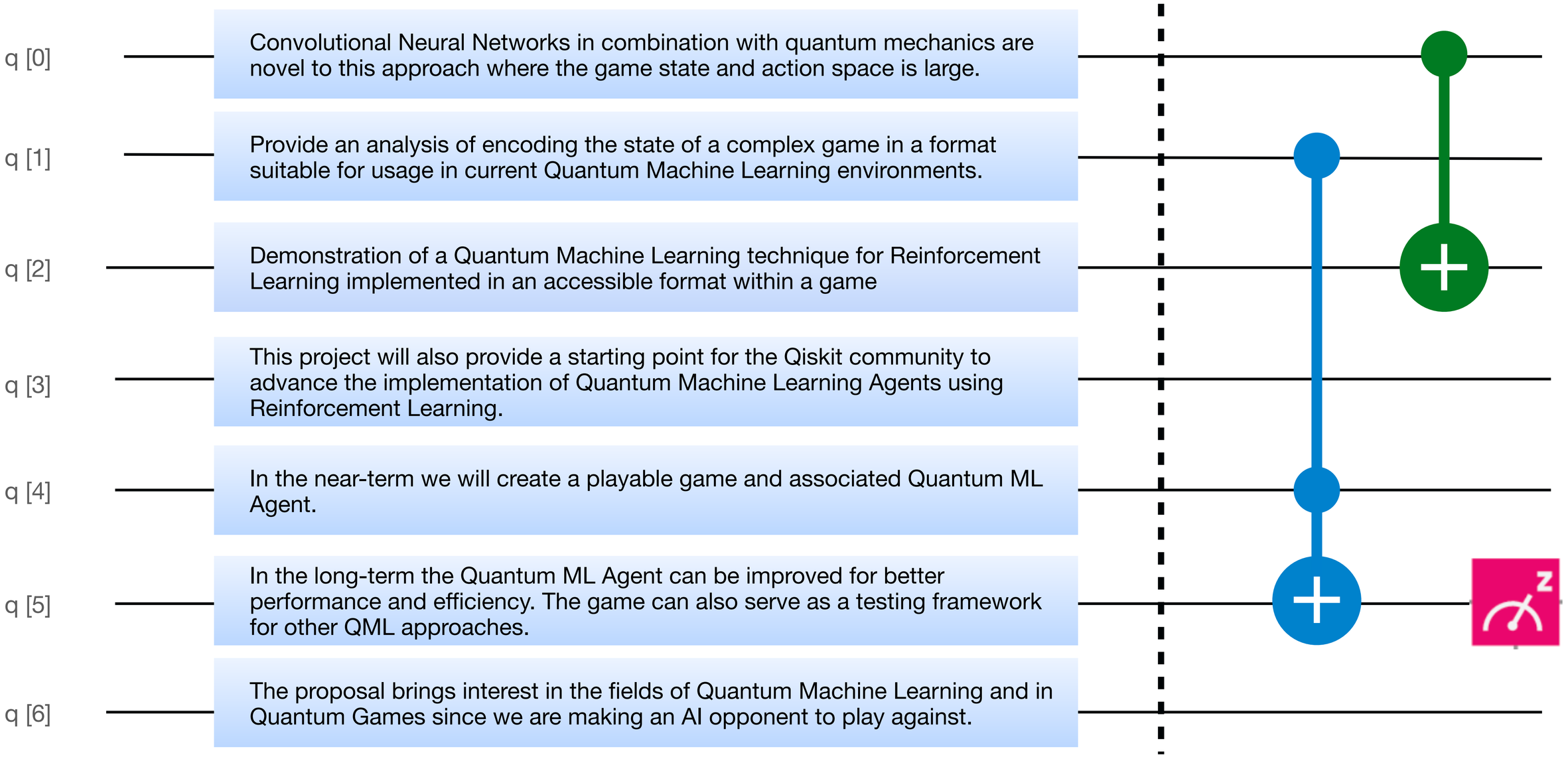
Demonstration of a Quantum Machine Learning technique for Reinforcement Learning implemented in an accessible format within a game

This project will also provide a starting point for the Qiskit community to advance the implementation of Quantum Machine Learning Agents using Reinforcement Learning.

In the near-term we will create a playable game and associated Quantum ML Agent.

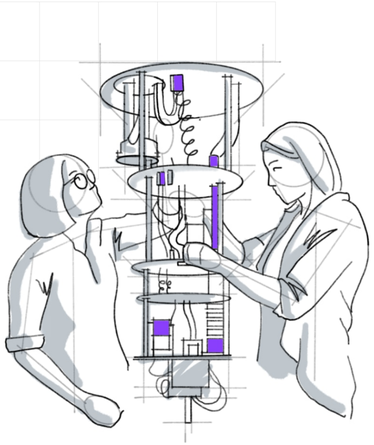
In the long-term the Quantum ML Agent can be improved for better performance and efficiency. The game can also serve as a testing framework for other QML approaches.

The proposal brings interest in the fields of Quantum Machine Learning and in Quantum Games since we are making an AI opponent to play against.



Implementation

Timeline, Skills and Qiskit Libraries



Team Beat-the-Quantum-Machine: Enda Cahill, Barbora Hrdá, Nouhaila Innan, David Peral, Divyanshu Singh

7th of May - 14th of May

- Implementation of the Othello mechanics and rules.
- Designing and implementing the user interface.
- Building initial ML Agent
- State and Action Encoding research

Skills:

- Game Development: Barbora, Divyanshu,
- UI: Team
- ML: David, Enda, Nouhaila

15th of May - 28 of May

- Implementation of the Quantum Layer.
- Refine State Encoding approach
- Compare results of different approaches

Skills:

- QML: David, Enda, Nouhaila
- Qiskit: Barbora, Divyanshu

28th of May - 4th of June

- Iterative testing, and final adjustments
- Finalizing project report
- Buffer time

Skills:

- Testing: Team
- Project Report Writing: Team

4th of June

Handing in:

- Working Prototype
- Project Report
- Project Video

Libraries / Modules / Game Engines:

- Qiskit-Aqua for the implementation of Machine Learning algorithm
- Qiskit-Aer for access to the backends
- PyTorch
- PyGame

For more information, see also: https://github.com/HayleySummer/Qiskit_Hackathon_Europe or Discord Channel: #team-beat-the-quantum-machine