Research Plan- QIntern 2025

Viraj Dsouza
Quantum Computing Researcher, BQP
`virajdanieldsouza@gmail.com`

1. Project Title

Efficient Block-Encoding of Dense Matrices via QRAM using the Qrisp Framework

2. Project Overview

This project provides a scalable and modular implementation of the block-encoding techniques for dense matrices proposed in recent work [1]. Block-encoding is a key technique in quantum algorithms that enables quantum access to classical matrices. We closely follow the methodology outlined in [1], which introduces two primary constructions: a *minimal-depth method*, achieving T-depth $\mathcal{O}(\log(N/\epsilon))$, and a *minimal-count method*, achieving T-count $\mathcal{O}(N\log(1/\epsilon))$. These constructions rely on efficient quantum circuit design for QRAM-based input models and state preparation subroutines. Our implementation, built using **Qrisp** [2] —a modern, Python-based high-level quantum programming framework—bridges theory and application, making these techniques accessible for quantum algorithms in linear system solving, Hamiltonian simulation, machine learning, etc. The qrisp implementation provides:

- Modular circuit constructions using intuitive syntax,
- Core data structure support such as `QuantumArray`, `QuantumDictionaries`, and others, which can facilitate QRAM implementations,
- Integration with the Jasp submodule for automatic scaling of quantum circuits, and
- Backend execution through hardware and simulators.

This implementation provides quantum developers with efficient state preparation and QRAM-based loading routines with optimal T-depth and T-count scaling, enabling for applications in algorithms beyond block-encoding. This helps bridge the gap between circuit-level research and practical quantum algorithm deployment.

3. Expected Background and Preparation

Interns are expected to have a solid understanding of basic quantum computing principles, linear algebra and familiarity with simple quantum algorithms. A working knowledge of Python and some experience with any quantum programming language are recommended. As preparation, interns are encouraged to skim through the reference paper [1] and practice writing simple quantum programs using Qrisp tutorials.

4. Six week Execution Plan

Week	Milestones
Week 1	Understand the theoretical foundations of block-encoding and familiarize with the Qrisp framework by writing basic quantum programs.
Week 2-3	Study and implement the two QRAM-based techniques—*Select-Swap* and *Bucket Brigade*—as described in the reference paper, using Qrisp.
Week 4	Read and implement both *Fixed Precision* and *Pre-rotated* state preparation techniques with modular Qrisp implementations.
Week 5	Integrate QRAM and state preparation components to complete the full block-encoding pipeline. Compute and validate resource estimates (T-depth and T-count), and compare against theoretical predictions.
Week 6	Test the scalability of the implementation using real-world matrix data, and perform robustness and performance checks to ensure readiness for downstream applications.

5. Deliverables

- A well-documented Jupyter notebook with scalable Qrisp implementations, submitted as an open-source contribution.
- (Optional) An arXiv paper with detailed comparative analysis of QRAM-based and other block-encoding techniques, co-authored with interested interns.

6. Guidance and Extension

I will be actively involved in both the theoretical and implementation aspects of the project, providing regular support via Slack or Discord to address any queries. For Qrisp-specific issues, the core Qrisp developers have agreed to assist via Slack. A short introductory presentation can be arranged if needed. Weekly progress meetings will be held to review milestones, with additional meetings upon request. If interns are interested in extending the work beyond the project duration, we can collaboratively pursue an arXiv publication comparing the implemented QRAM-based block-encoding techniques with alternative approaches, including performance benchmarks, resource estimates, and application case studies. Intern-driven ideas for further development are highly encouraged.

References

[1] B. David Clader, Alexander M. Dalzell, Nikitas Stamatopoulos, Grant Salton, Mario Berta, and William J. Zeng. Quantum Resources Required to Block-Encode a Matrix of Classical Data . *IEEE Transactions on Quantum Engineering*, 3(01):1–23, January 2022.

[2] Raphael Seidel, Sebastian Bock, René Zander, Matic Petrič, Niklas Steinmann, Nikolay Tcholtchev, and Manfred Hauswirth. Qrisp: A framework for compilable high-level programming of gate-based quantum computers, 2024.