Implementing Advanced Pulse Control for Atomic Qubits

A combination of amplitude, frequency and phase modulation is often required to control the state of Ion-Trap and Neutral Atoms based qubits. Timing accuracy, maintaining phase coherence, and compensating temperature related variations in analog and mixed signal electronics add complexity to the control system, causing an increment in the required hardware resources and costs. The quality of the pulses generated directly affects aspects such a gate fidelity for quantum gates, which is a primary metric to measure the performance of Quantum Computers Project

This project involves investigating efficient ways of describing arbitrary, bandlimited waveforms with cubic splines, while optimising the curve fitting error and the number of employed cubic polynomials. After modelling the devised curve fitting methods through python, the selected method will be integrated within Riverlane Control System, and used to described real pulses aiming at performing tasks such as Ion or Atom trapping, shuttling, or generate advanced quantum gates, using schemes such as Mølmer–Sørensen

Profiling

Performance profiling is the measurement and analysis of an aspect of software performance such as memory usage, execution time, or error propagation, to aid in optimising said software. The Deltaflow.Control software stack is constantly growing, with several moving parts providing many opportunities for performance improvements. Unlocking these performance gains then improves the user experience by making using Deltaflow.Control more responsive and improving the efficiency of users' experiment workflows.

You will be tasked with profiling the execution time (other factors may also be analysed, given time) of the Deltaflow.Control software stack. This will involve measuring the execution time of different parts of the stack with different stimuli and analysing the results to find where the bottlenecks are. You will then work towards alleviating these bottlenecks, improving system performance. You will also work on the implementation of basic telemetry to streamline future performance analysis.

Constraint solving the control scheduling problem

Deltaflow.Control is the classical hardware+software system that drives pulses into, and reads signals out from, the quantum hardware. A problem that Deltaflow.Control has to solve is scheduling the pulses over a set of hardware ports at high speed, maximizing parallelism, to ns accuracy, and under several application and system constraints.

We currently have tools to do this. But it would be interesting to build a model of the solution space using a modern constraint solving system like https://www.minizinc.org/. This would represent the Deltaflow.Control domain as a scheduling, bin-packing, etc type of optimization problem. The model can be used to both explore the domain, and to verify the correctness and efficiency of our current tools.

Quantum Error Correction in The Real World

For quantum computers to be reliable enough to run the most useful algorithms, those that provide significant advantage over classical alternatives, they need to be able to detect and correct errors as they occur, a process called quantum error correction (QEC). At Riverlane, we know that QEC is *the* grand challenge in quantum computing. Thus, we are tackling QEC across the quantum computing stack as part of our quantum operating system Deltaflow.OS®.

Help Riverlane make quantum computing useful sooner by contributing to our effort to bridge the gap between the theoretical and practical side of QEC. This will be achieved by researching, designing and implementing new components for an existing codebase that has been developed by Riverlane. A successful outcome for this project would look like several methods from literature being implemented, with the results from running the implementations being presented at the end of the internship.

General Lattice Surgery

At Riverlane we are building a software which enables us to test various quantum error correction ideas. Planar codes, for example the surface code, seem particularly promising towards building fault tolerant quantum computation. Going beyond quantum memory, the next natural step with planar codes is to do lattice surgery. With lattice surgery one can do universal quantum computing in a fault tolerant way. Special cases of lattice surgery are already implemented in our software, and the goal of this project will be to investigate and implement more general cases. This will involve studying papers and coding in Python.

Floquet Codes

Floquet codes are a family of quantum error correcting codes that have recently been of significant interest. They work by performing pairwise measurements of data qubits according to a measurement schedule, with detectors inferred from products of measurement results from the previous two rounds. One example is the Honeycomb Code, which is implemented on a periodic hexagonal lattice. However, Floquet codes are more general, and can be implemented on any 3-colourable graph. The aim this project is to investigate and implement these more general Floquet codes. This could include both planar and periodic versions of Floquet codes.

Quantum Embedding for Studying Heterogenous Catalysis

In heterogeneous processes, the catalyst and the reactants form different phases that require different theoretical approaches. In practice, this often involves calculating the ground state energy of molecules (adsorbate) at different surface sites. To make the problem tractable, one considers the region around the adsorbate with higher precision methods while the rest of a system is treated with low-cost computational methods such as density functional theory. The embedded region can be then described with quantum chemistry methods such as coupled cluster theory or can be simulated on a quantum computer.

In this work, the intern will learn how to use an electronic structure software in order to construct the Hamiltonian associated with embedded region and write an interface for our internal tools. The intern will also learn how to use advanced qubitisation techniques and implement them in software. Finally, the intern will use this software to work on industry relevant applications.

Intermolecular interactions on a quantum computer

Simulation of quantum chemical systems is one of the most promising applications of quantum computers. However, modelling of large molecules such as proteins will not be possible with quantum computers with *ab initio* Hamiltonians even if the current state of the art quantum algorithms will be improved by several order of magnitudes.

In this work the intern will learn how to describe large scale systems and intramolecular interactions on a quantum computer using model Hamiltonians. This work will be carried out with top academic partners.

Automating and Optimising Qubit Calibration

Running experiments to calibrate qubit parameters—determining the exact settings with which to control the qubits—is an essential prerequisite before a quantum computer can be used to run algorithms. Unfortunately, calibration represents one of the most time-consuming activities that quantum computing labs are forced to do: calibration consists of many experiments and must be done continuously because laboratory conditions are constantly changing. Adding to the challenge is that many of these experiments are currently being run manually.

In this project you will help our team deliver a calibration solution that our hardware partners will use in their labs. In particular, candidates will work with our team and partners to implement new ways to automate experiments. Candidates who have a software engineering background will design APIs and software libraries that will simplify and enhance calibration activities. Candidates with a machine learning, data science or information visualisation background will create tools to process, analyse, and learn from experiment data.