Development of Novel Algorithms

By Team: Quantum Cats

Mahroo

Cynthia Cruz Sanchez, (do we put what we studied? No)

Johanna

Womanium Summer 2024, Projet Component

Selected paper:

Exponential quantum speedup in simulating coupled classical oscillators

Ryan Babbush,[1] Dominic W. Berry,[2] Robin Kothari,[1] Rolando D. Somma,[1] and Nathan Wiebe[3,4,5]

[1] *Google Quantum AI, Venice, CA, United States*
[2] *School of Mathematical and Physical Sciences, Macquarie University, Sydney, NSW, Australia*
[3] *Department of Computer Science, University of Toronto, Toronto, ON, Canada*
[4] *Pacific Northwest National Laboratory, Richland, WA, United States*
[5] *Canadian Institute for Advanced Research, Toronto, ON, Canada*

(Dated: September 21, 2023)

We present a quantum algorithm for simulating the classical dynamics of 2^n coupled oscillators (e.g., 2^n masses coupled by springs). Our approach leverages a mapping between the Schrödinger equation and Newton's equation for harmonic potentials such that the amplitudes of the evolved quantum state encode the momenta and displacements of the classical oscillators. When individual masses and spring constants can be efficiently queried, and when the initial state can be efficiently prepared, the complexity of our quantum algorithm is polynomial in n, almost linear in the evolution time, and sublinear in the sparsity. As an example application, we apply our quantum algorithm to efficiently estimate the kinetic energy of an oscillator at any time. We show that any classical algorithm solving this same problem is inefficient and must make $2^{\Omega(n)}$ queries to the oracle and, when the oracles are instantiated by efficient quantum circuits, the problem is BQP-complete. Thus, our approach solves a potentially practical application with an exponential speedup over classical computers. Finally, we show that under similar conditions our approach can efficiently simulate more general classical harmonic systems with 2^n modes.

Key Details for Final Project Presentation:

All teams must submit a final project presentation deck by linking it on their Github project repos by **Aug.9**.
You may optionally also attach a recording of your presentation on your GitHub repo.
The maximum duration of the presentation is **3 minutes**.
Project finalists will be invited to present their presentations on the Demo Day [Aug.15].

Recommended format for Final Project Presentation:

The following should be covered in your presentation:

- Problem Statement - what is your project about? What problem are you trying to solve? [~30 seconds]
- Your project solution - what were the different objectives of your project and how did you achieve them? [~60-90 seconds]
- Success - Did you achieve your desired results? What were your success metrics? How did your project make an impact on the current field of Quantum Science? [~30-60 seconds]
- Future scope - What are some future steps you would recommend to extend your project? What limitations are you facing today stopping you from achieving your future recommendations? [~30 seconds]

The Womanium Logo [link] should be clearly visible on your presentation.
Each presentation slide must display/ mention “Womanium Quantum+AI Project”.
You are NOT allowed to use any third-party images/ logos without proper references in your presentation.

1) Participants need to implement, optimize, one of the following quantum algorithms using Classiq:

a) Classically verifiable quantum advantage from a computational Bell Test.

b) Exponential quantum speedup in simulating coupled classical oscillators.

c) Evidence for the utility of quantum computing before fault tolerance.

4. Project Tasks/ Deliverables:

2) The first step will be to implement a toy problem of the chosen algorithm.

This should be a simple implementation (the simplest possible) that covers

the:

a) Encoding of the problem.

b) The key algorithmic building blocks of the chosen paper (e.g.

Hamiltonian simulation).

c) The readout and post-processing.

The implementation should be scalable, such that it is clear how to extend it

for a more complicated scenario, and it should be checked and tested using

a simulator.

3) The second step is to enlarge the problem for a more complicated scenario.

In this step, the actual problems from the papers should be implemented

(e.g. the actual Hamiltonian that is shown in the paper should be

implemented). Resources estimation in terms of circuit depth, circuit width

and number of 2-qubit gates should be made and compared across several

hardwares.

4) The last step would be to optimize the solution for the most adequate

Deliverables:

- Slides that summarize the work.
- The .qmod and .qprog files for each step.
- The Python Jupyter notebooks of each step (if applicable).