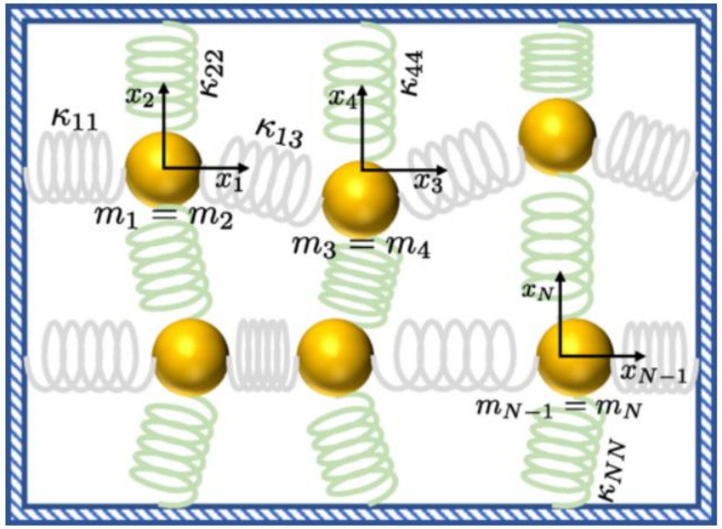
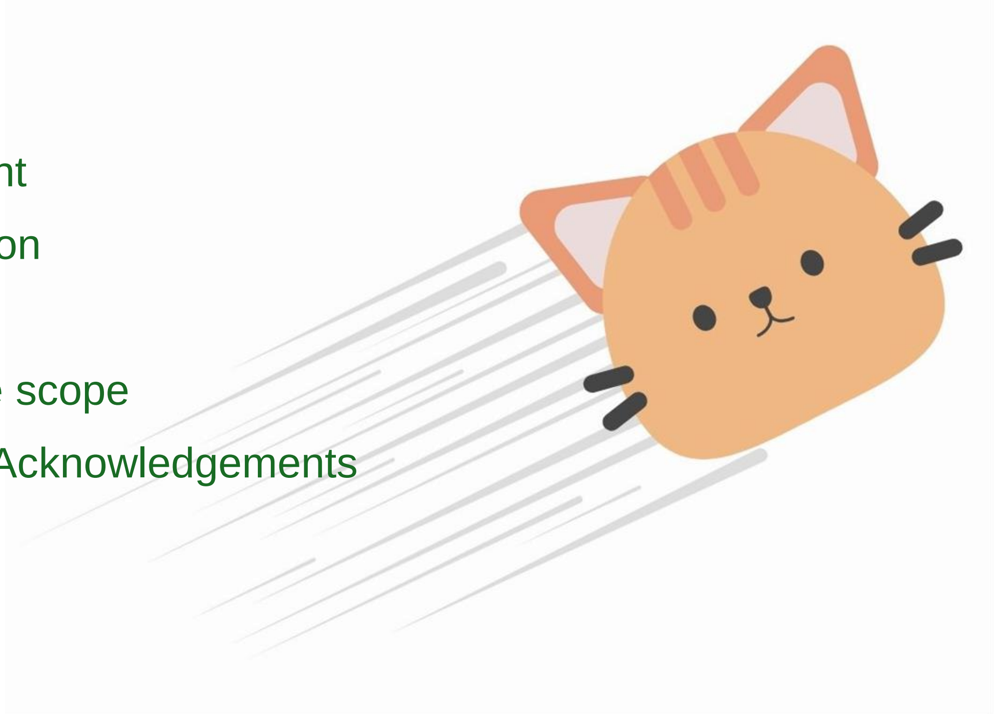


Network of classical coupled oscillators (Baboush et al, 2023, https://doi.org/10.1103/PhysRevX.13.041041 , [1])

Development of Novel Quantum Algorithms

Exponential quantum speedup in simulating coupled classical oscillators.

Content:

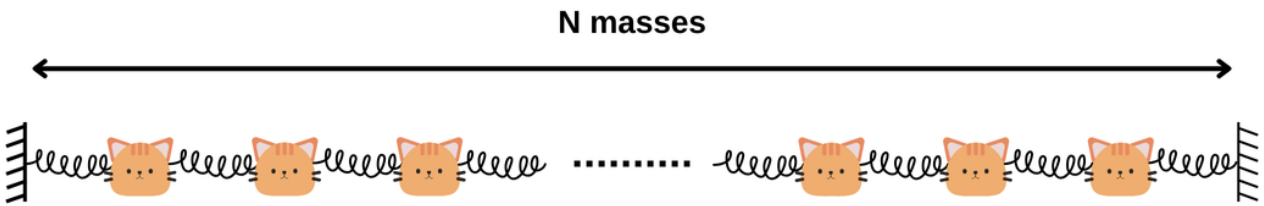


1. Problem Statement
2. Solution Description
3. Achievements
4. Issues and Future scope
5. Bibliography and Acknowledgements

The Abstract Oscillators Team:
Kerem Yurtseven
Cristina Radian
Viraj Daniel DSouza

Problem Statement

Building on the theoretical framework from the 2023 study by Babbush et al., our project seeks to translate the exponential quantum speedup in simulating coupled classical oscillators into a **practical implementation using Classiq**, compatible with current quantum hardware and software.



Project Importance: Achieving this practical quantum advantage would revolutionize fields like material science, climate modeling, and medical technology by enabling unprecedented accuracy and efficiency in complex simulations, driving innovation across multiple disciplines.

Toy Problem – Task 1

Configuration

- **N = 2**
- **m1 = m2 = 1**
- **K12 = 1**

$$|\psi(t)\rangle := \frac{1}{\sqrt{2E}} \begin{pmatrix} \sqrt{\mathbf{M}}\dot{\vec{x}}(t) \\ i\vec{\mu}(t) \end{pmatrix}$$

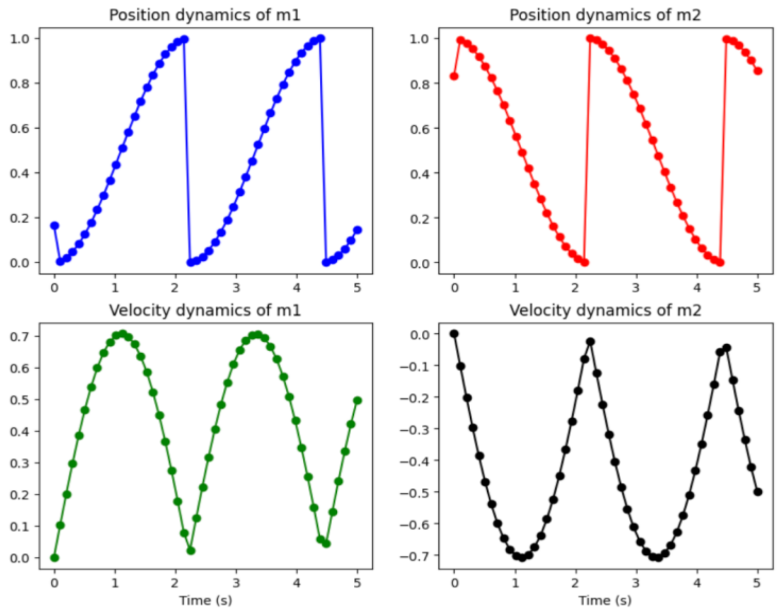
The final quantum state

Methodology

Pre-Processing → Pauli Decomposition → Hamiltonian Evolution → Post-Processing

Product-based	Block Encoding
Exponentiation Suzuki Trotter	Qubitization QSVT

Results

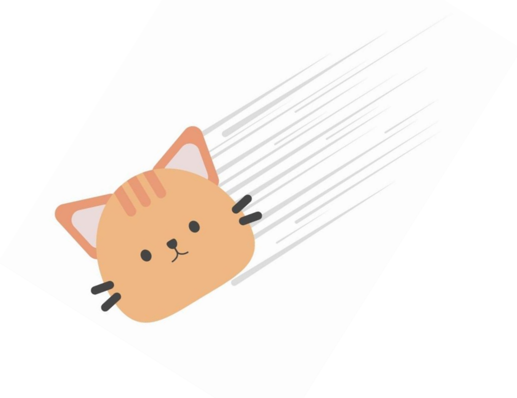


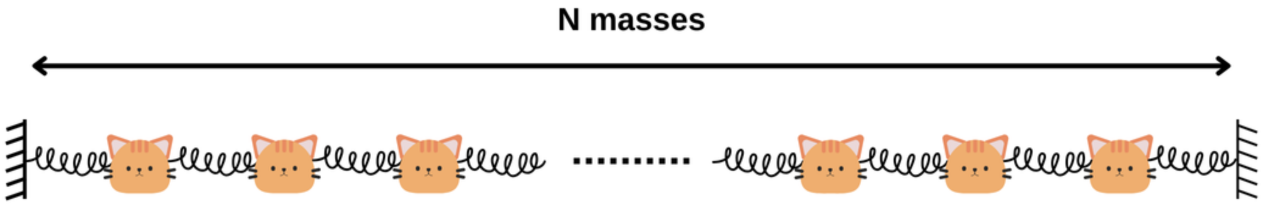
Errors at t = 0.5

	Velocities	Positions
Mass 1	%0.82	%1.78
Mass 2	%0.63	%0.30

!PHASE ERROR!

Kinetic Energy Estimation – Task 2

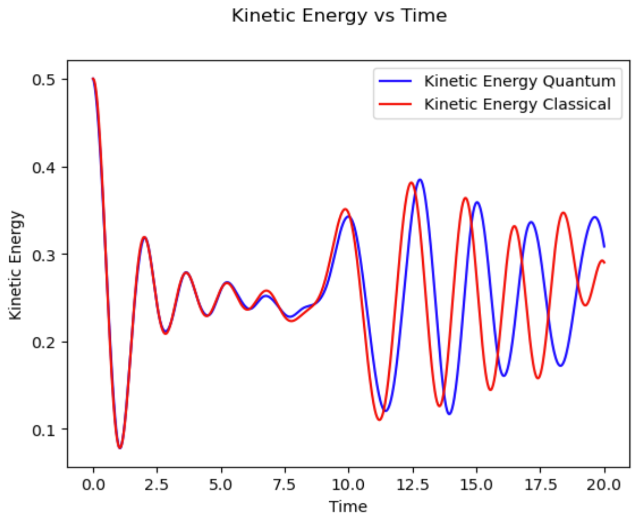


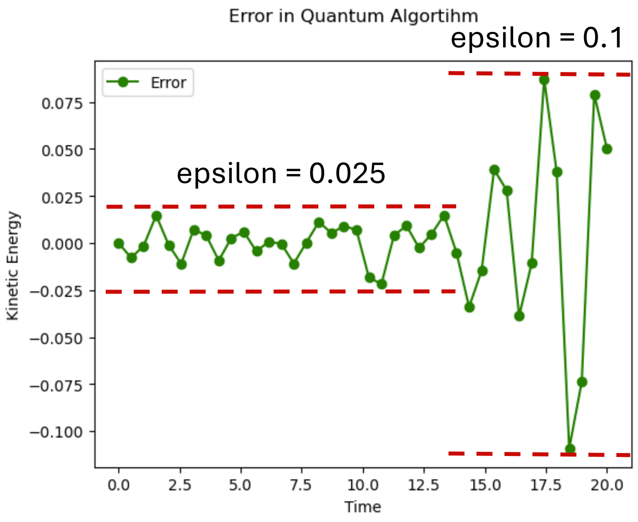


Real Example can be simulated with N = 8 masses to prove the final equation

$$\left| \hat{k}_V(t) - \frac{K_V(t)}{E} \right| \leq \epsilon$$

Params: m = 1, k = 1, Suzuki Trotter



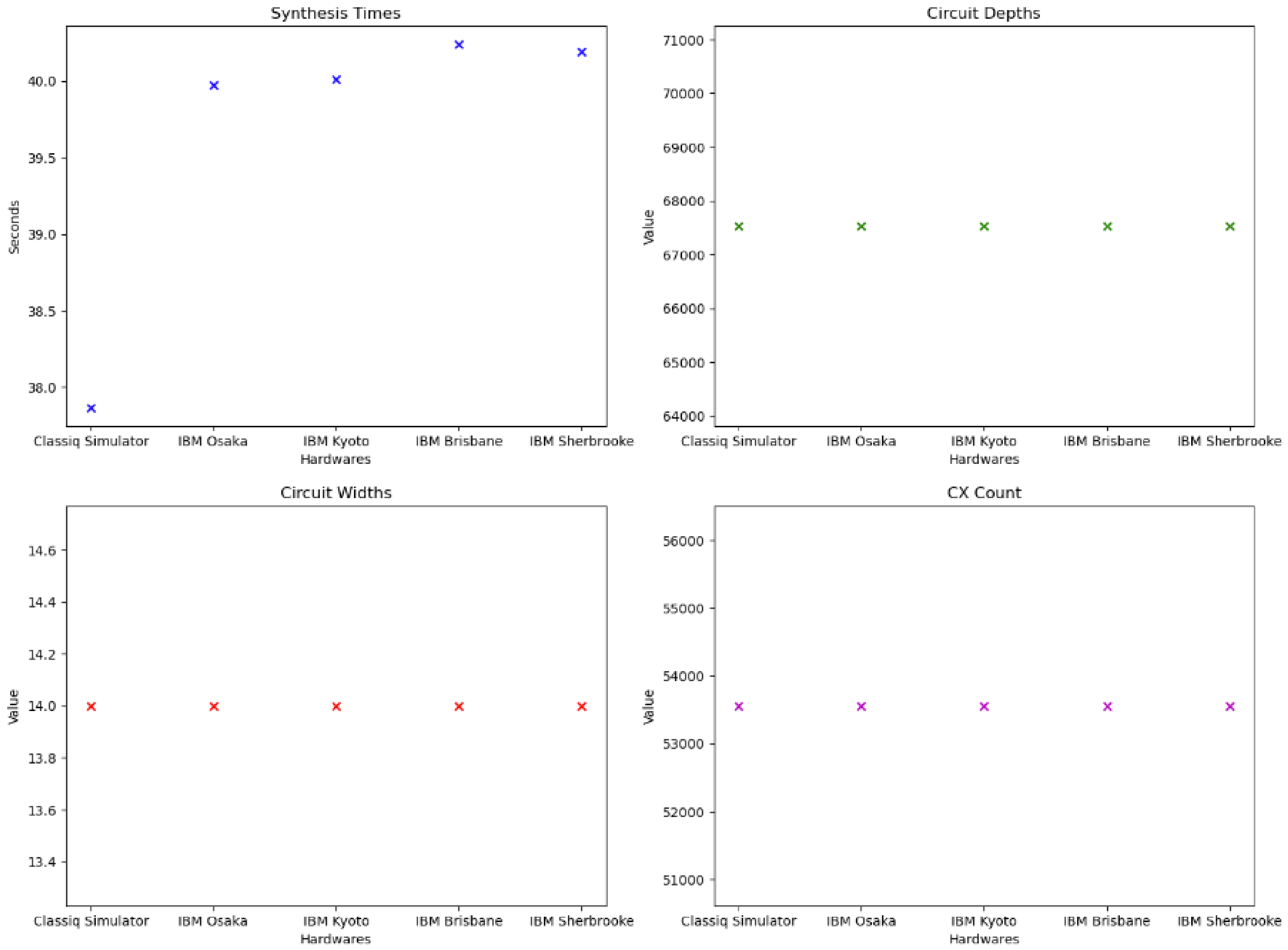


Resource Estimation

Real Quantum Hardwares are critical to understand the importance and reality of quantum algorithms

Compared the following simulator and IBM Quantum Hardware for Kinetic Energy Estimation

- Classiq Simulator
- IBM Osaka
- IBM Kyoto
- IBM Sherbrooke
- IBM Brisbane



Resource Estimation with Suzuki Trotter

Optimization – Task 3

All of the hardwares give same circuit parameters except synthesis time. So, choose IBM Osaka with lowest time and try to optimize it

IBM Osaka	No Optimization	Depth Optimization	Width Optimization
Synthesis Time	39.97 s	66.69 s	67.00 s
Depth	67537	67537	67537
Width	14	14	14
CX Gate Number	53561	53561	53561

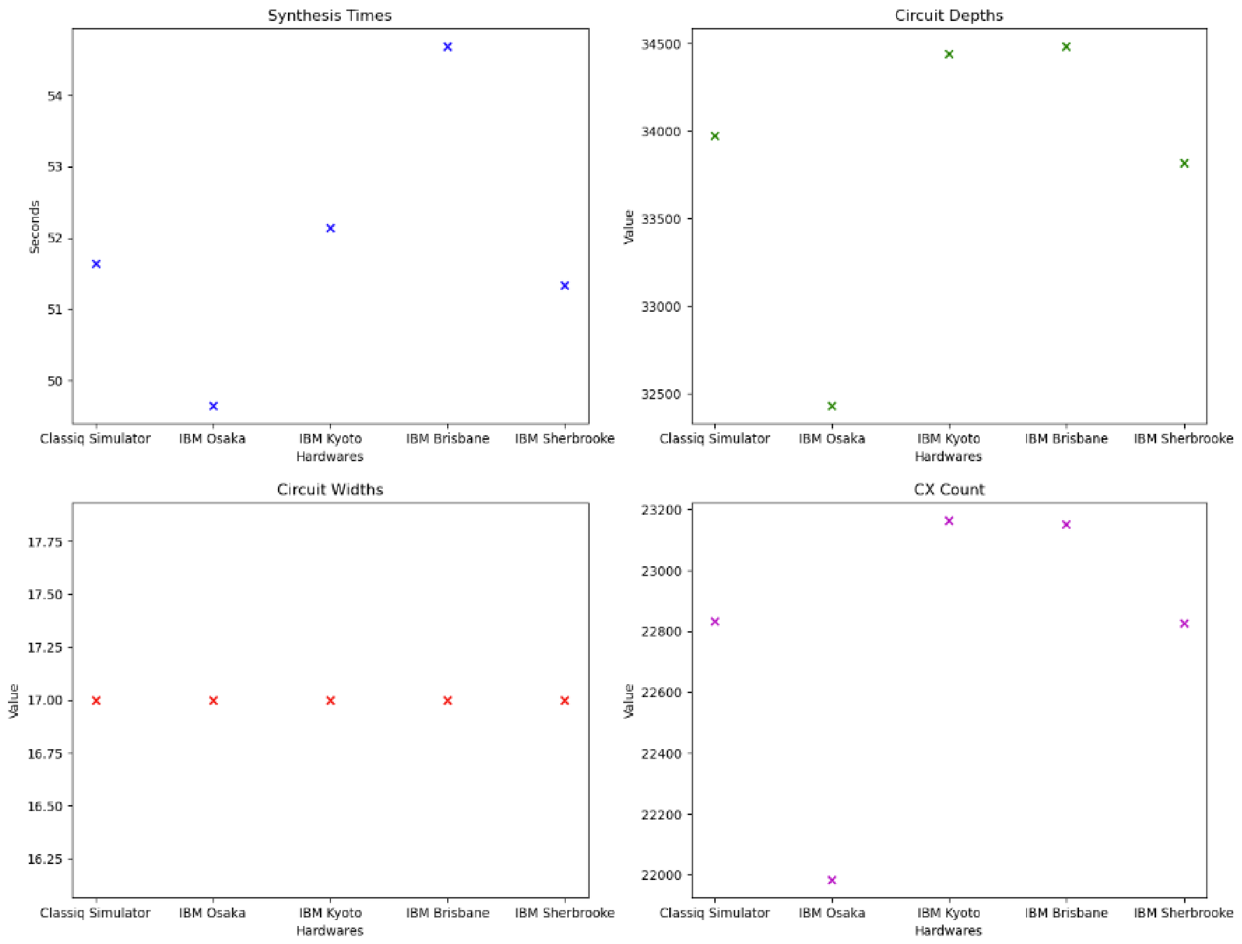
!NO OPTIMIZATION!
CHANGE EVOLUTION METHOD

Qubitization

Qubitization for <u>Kinetic Energy Estimation</u>, we set **N = 2** due to very long synthesis and execution times

Compared the following simulator and IBM Quantum Hardwares

- Classiq Simulator
- IBM Osaka
- IBM Kyoto
- IBM Sherbrooke
- IBM Brisbane



Resource Estimation with Qubitization

Hardwares differ a lot in depth and CX gate number and **IBM Osaka** is the most appropriate amongst them. We chose this for the next step- **Optimization**

Optimization with Qubitization

IBM Osaka has a maximum of **127 qubits**, so we tried to trade-off very large depths for some qubits

IBM Osaka	No Optimization	Depth Optimization	Width Optimization	Depth Opt. 127 Max
Synthesis Time	49.65 s	189.37 s	818.17 s	2350.92 s
Depth	32342	19767	56581	**19744**
Width	17	1273	15	**127**
CX Gate Number	21984	14836	39098	**14816**

SUCCESSFUL OPTIMIZATION

Optimization with Qubitization

127 Qubits

Achievements

- Task 1
  - **Toy examples successfully simulated and compared with classical results**
  - Multiple evolution methods implemented with considerable accuracy
- Task 2
  - **A real problem from the paper illustrated and theoretical assertions supported with simulations**
  - Resource estimation and hardware comparison for the real problem
- Task 3
  - Most appropriate hardware chosen and tried to get optimized results with different evolution methods
  - **Optimal transpiled circuit created for the implementation of the algorithm on real quantum hardwares**

Future Scope and Issues

Issues

- Post-processing the global phase
- Uncertain positions with singular B matrix
- Higher number of masses

Future Scope

- Further optimization on real quantum hardware
- Real quantum hardware simulations with higher number of masses
- Post-processing for the results of real quantum hardware
- Generalization for different subjects such as electric circuit simulations, molecular simulations, climate modelling, material science etc.

Bibliography and Acknowledgements

Bibliography:

1. R. Babbush, D. W. Berry, R. Kothari, R. D. Somma, and N. Wiebe, *Exponential Quantum Speedup in Simulating Coupled Classical Oscillators*, Phys. Rev. X 13, 041041 (2023), https://doi.org/10.1103/PhysRevX.13.041041
2. 2. G.H. Low, I.L. Chuang, *Hamiltonian simulation by Qubitization*, published in Quantum, 3 (2019), p. 163, https://doi.org/10.22331/q-2019-07-12-163
3. András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe. 2019. Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics. In Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing (STOC 2019). Association for Computing Machinery, New York, NY, USA, 193–204. https://doi.org/10.1145/3313276.3316366, link to paper
4. M. Szegedy, "Quantum speed-up of Markov chain based algorithms," In 45th Annual IEEE symposium on foundations of computer science, pages 32–41, 2004, link to paper
5. Classiq Github Glued Trees https://github.com/Classiq/classiq-library/blob/9c43f05f3d498c8c72be7dcb3ecdaba85d9abd6e/algorithms/glued_trees/glued_trees.ipynb#L4
6. Classiq Github Hamiltonian Qubitization https://github.com/Classiq/classiq-library/tree/9c43f05f3d498c8c72be7dcb3ecdaba85d9abd6e/tutorials/hamiltonian_simulation/hamiltonian_simulation_with_block_encoding
7. Classiq documentation https://docs.classiq.io/latest/

Acknowledgments

A heartfelt gratitude to Womanium Team for designing & organizing this program and offering scholarships.
Special thanks to Eden Shirman, Tomer Goldfriend, and everyone at Classiq.
This project uses Classiq Github by Classiq.

THANKS!