CENTRE EUROPÉEN DE RECHERCHE ET DE FORMATION AVANCÉE EN CALCUL SCIENTIFIQUE

Advances in implementation of Hamiltonian Simulation algorithms

Application to the 1-D wave equation

Presented by
Adrien Suau
adrien.suau@cerfacs.fr
CERFACS

In collaboration with
Gabriel Staffelbach
gabriel.staffelbach@cerfacs.fr
CERFACS

Thanks to

- **Reims university** for giving me an access to their QLM.
- **Total** for giving me an access to their QLM & work.
- **Atos** for the technical support.
- **Ter@tec** for all the Quantum Computing events.

Purpose of this presentation

Goal of the presentation

Present & analyse the results of the quantum wave equation solver implementation

Not included in this presentation

- Full explanation of the algorithm used and the implementation

Objective:

Answer to:

- With today's resources & algorithms, are we able to implement a solver?
- Is the implementation efficient when compared to classical?

Is quantum computing now?

Actual quantum technology

Quantum wave equation solver

Results on a practical case

Quantum hardware is real...

- **IBM**: 4 chips, 20 qubits max.
- **Intel**: 3 chips, 49 qubits max.
- **Google**: 1 chip, 72 qubits max.

...but a lot of technical difficulties

- Low coherence times
- High error-rates
- Scaling qubit count is very challenging

What is a *quantum simulator*?

A **classical** software running on **classical** CPU that emulates **quantum** hardware

Characteristic	Real hardware	Simulator (QLM)
Can provide a quantum speedup	✓	×
Maximum Qubits	$\sim 70^*$	$\sim 40^*$
Error-free	×	✓
Debug information	×	✓
Hardware independant	×	✓

*Data gathered in May 2019

Today, simulators are used instead of hardware chips because:

- Software is easier to debug.
- Can adapt to specific hardware afterwards.
  Example: the wave equation solver has been implemented on the QLM and then adapted to IBM chip “Melbourne”.

Actual quantum technology

Quantum wave equation solver

- What is Hamiltonian Simulation?
- Why Hamiltonian Simulation is important?
- How to solve the wave equation on a quantum computer?

Results on a practical case

Actual quantum technology

Quantum wave equation solver

- What is Hamiltonian Simulation?
- Why Hamiltonian Simulation is important?
- How to solve the wave equation on a quantum computer?

Results on a practical case

Time-dependent Schrödinger equation

The solution of the time-dependent Schrödinger equation governing the evolution of a physical system

$$\frac{d}{dt} \left| \Psi\left(t\right) \right\rangle = -iH \left| \Psi\left(t\right) \right\rangle$$

is given by

$$\left| \Psi\left(t\right) \right\rangle = e^{-iHt} \left| \Psi\left(0\right) \right\rangle$$

Time-dependent Schrödinger equation

The solution of the time-dependent Schrödinger equation governing the evolution of a physical system

$$\frac{d}{dt} \left|\Psi\left(t\right)\right\rangle = -iH \left|\Psi\left(t\right)\right\rangle$$

is given by

$$\left|\Psi\left(t\right)\right\rangle = e^{-iHt} \left|\Psi\left(0\right)\right\rangle$$

Remark:

The matrix H has a size that grows exponentially with the physical system size.

Problem formalisation:

Given an Hamiltonian matrix H, a time t, a precision ϵ and a basis of several quantum gates, find a sequence of quantum gates $U = U_1 \dots U_n$ picked from the given basis that approximates the unitary matrix e^{-iHt} such that

$$\left|\left|e^{-iHt} - U\right|\right|_{\mathsf{sp}} \leqslant \epsilon$$

with $|| \cdot ||_{\mathsf{sp}}$ the spectral norm.

Hamiltonian Simulation can be used as a subroutine for:

1. the computation of molecular energies[1]
2. linear systems resolution[2]
3. graph algorithms[3]
4. partial differential equations resolution[4]

[1] https://docs.microsoft.com/en-us/quantum/libraries/chemistry/
[2] Harrow, Hassidim, and Lloyd, "Quantum Algorithm for Linear Systems of Equations".
[3] Childs, Cleve, et al., "Exponential algorithmic speedup by quantum walk".
[4] Childs and Liu, "Quantum spectral methods for differential equations".

According to Costa, Jordan, and Ostrander[5], the wave equation

$$\frac{d^2}{dt^2}\phi = \frac{d^2}{dx^2}\phi$$

- + boundary conditions
- + initial conditions
- + fixed propagation speed $c = 1$.

can be solved by simulating the action of a specific Hamiltonian to a quantum state encoding the initial state.

[5] Pedro C. S. Costa, Stephen Jordan, and Aaron Ostrander. "Quantum algorithm for simulating the wave equation". In: *Physical Review A* 99 (1 Jan. 2019). Phys. Rev. A 99, 012323 (2019). DOI: 10.1103/PhysRevA.99.012323. eprint: 1711.05394v1. URL: http://arxiv.org/abs/1711.05394v1.

Actual quantum technology

Quantum wave equation solver

Results on a practical case

- Methodology
- Quantum solver VS. finite differences
- Required hardware characteristics

Actual quantum technology

Quantum wave equation solver

Results on a practical case

- Methodology
- Quantum solver VS. finite differences
- Required hardware characteristics

Default values

If not stated otherwise, the default values used for each graph are:

- Precision $\left|\left|e^{-iHt} - U\right|\right|_{\text{sp}} \leqslant \epsilon = 10^{-5}$
- Number of discretisation points $N_{\text{discr}} = 32$
- Order of the product-formula used $PF_{\text{order}} = 1$
- Simulation physical time $t = 1$

Actual quantum technology

Quantum wave equation solver

Results on a practical case

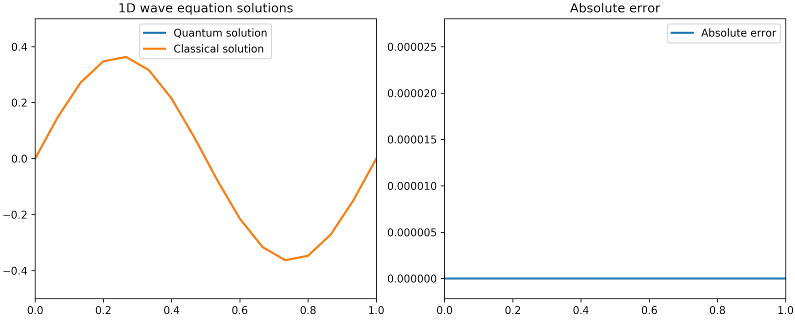
- Methodology
- Quantum solver VS. finite differences
- Required hardware characteristics

Quantum solver:

- $\epsilon = 10^{-2}$
- $N_{\text{discr}} = 16$
- $PF_{\text{order}} = 1$
- $t \in [0, 1]$

Classical solver

- Finite differences
- $\delta x = \frac{1}{(N_{\text{discr}}+1)}$
- $\delta t = 10^{-5}$

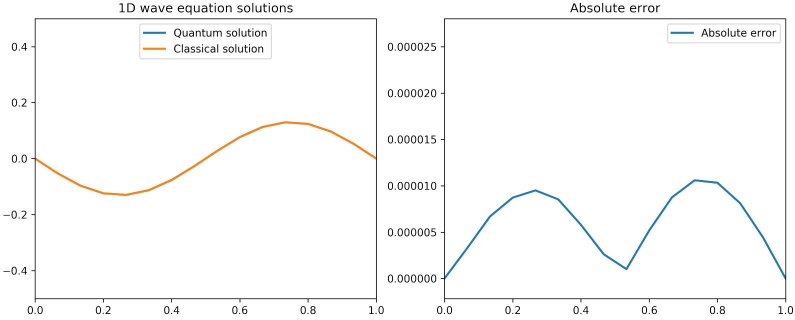


Quantum solver:

- $\epsilon = 10^{-2}$
- $N_{\text{discr}} = 16$
- $PF_{\text{order}} = 1$
- $t \in [0, 1]$

Classical solver

- Finite differences
- $\delta x = \frac{1}{(N_{\text{discr}}+1)}$
- $\delta t = 10^{-5}$



About gate counts and timing estimations

- Gate counts do not take into account hardware topology
- Gate counts are computed from the **generated** circuits (no post-generation optimisation)
- Gate counts performed after a translation from the simulator gate-set to IBM's chips gate-set
- Estimated execution times are computed using [1] and [2]

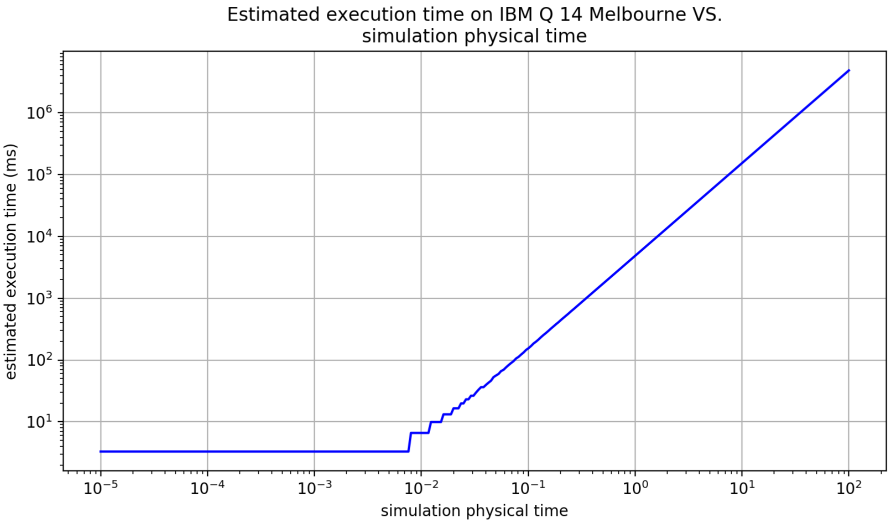
[1] https://github.com/Qiskit/ibmq-device-information/tree/master/backends/melbourne/V1#gate-specification

[2] https://github.com/Qiskit/ibmq-device-information/blob/master/backends/melbourne/V1/version_log.md#gate-specification

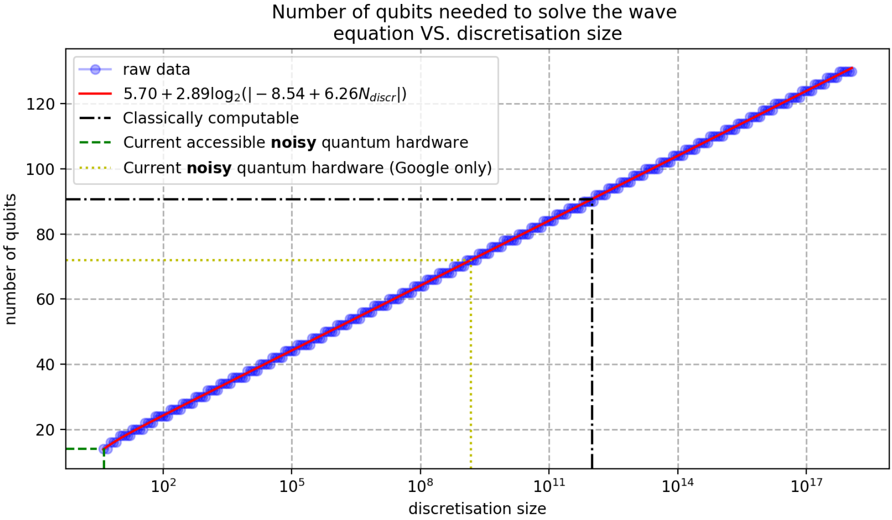
No fast-forwarding theorem:

The optimal gate complexity for a **generic** Hamiltonian simulation algorithm is $\mathcal{O}(t)$, t being the simulation time.

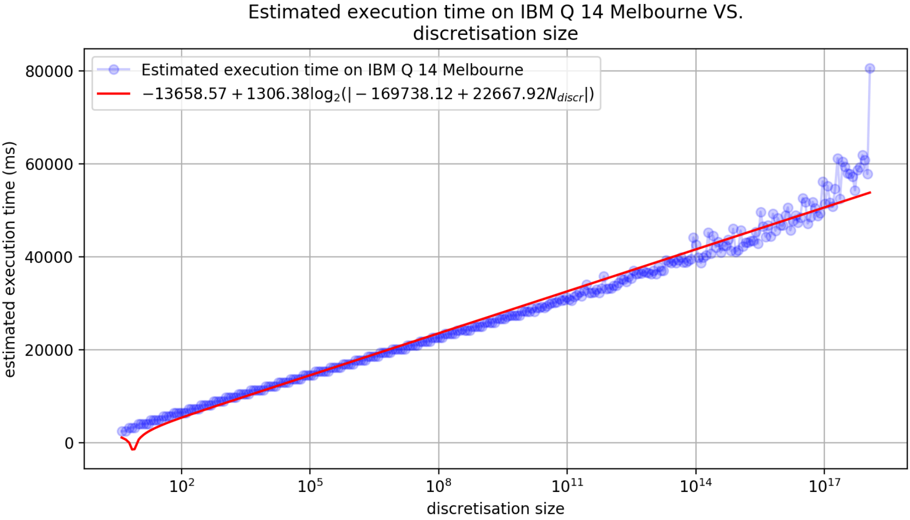
Andrew M. Childs and Robin Kothari. "Limitations on the simulation of non-sparse Hamiltonians". In: (Aug. 2009). Quantum Information and Computation 10, 669-684 (2010). eprint: 0908.4398v2. URL: http://arxiv.org/abs/0908.4398v2.



Estimated execution time on IBM Q 14 Melbourne VS. simulation physical time
estimated execution time (ms)
simulation physical time
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}
10^{-5}
10^{-4}
10^{-3}
10^{-2}
10^{-1}
10^{0}
10^{1}
10^{2}



Number of qubits needed to solve the wave equation VS. discretisation size
raw data
$5.70 + 2.89\log_2(|-8.54 + 6.26N_{discr}|)$
Classically computable
Current accessible **noisy** quantum hardware
Current **noisy** quantum hardware (Google only)
number of qubits
120
100
80
60
40
20
10^2
10^5
10^8
10^{11}
10^{14}
10^{17}
discretisation size


Estimated execution time on IBM Q 14 Melbourne VS. discretisation size
Estimated execution time on IBM Q 14 Melbourne
$-13658.57 + 1306.38\log_2(|-169738.12 + 22667.92 N_{discr}|)$
estimated execution time (ms)
80000
60000
40000
20000
0
10^2
10^5
10^8
10^{11}
10^{14}
10^{17}
discretisation size

Conclusion

1. Validated implementation of a quantum wave equation solver
2. Confirms theoretical complexity
3. Hardware requirements are clear but too high for today's chips
4. Atos QLM allows to investigate algorithms and start evaluating the benefit of practical applications

- Extend the wave equation solver to inhomogeneous medium (non constant propagation speed c),
- Implement Neumann boundary conditions,
- Implement a 2-D wave equation solver,

Current work group:

1. **Charles Moussa**: PhD student in Quantum Machine Learning.
2. **Yuan Yao**: Intern studying the Variational Quantum Eigensolver.
3. **Tam'si Ley**: Intern studying Quantum Gradient Descent.
4. **Adrien Suau**: Starting PhD in quantum computing in the next few months.

Supervised by:

- **Henri Calandra**
- **Gabriel Staffelbach**

Thank you for your attention!

Any question?

Contact information:

- Mail: adrien.suau@cerfacs.fr
- Phone: +33(0)5 61 19 31 19