Forum TERATEC 23

Unlock the future

31 MAI & 1er JUIN 2023 • Au Parc Floral, Paris

Un événement organisé par infoprodigital Teratec

Forum TERATEC 23

Quantum experiences from an hpc perspective

GABRIEL STAFFELBACH
ADRIEN SUAU
gabriel.staffelbach@cerfacs.fr

CERFACS

Research center focused on training and technology transfer using High performance computing

Concentrate competences in HPC, numerical methods, modelling to tackle scientific problems

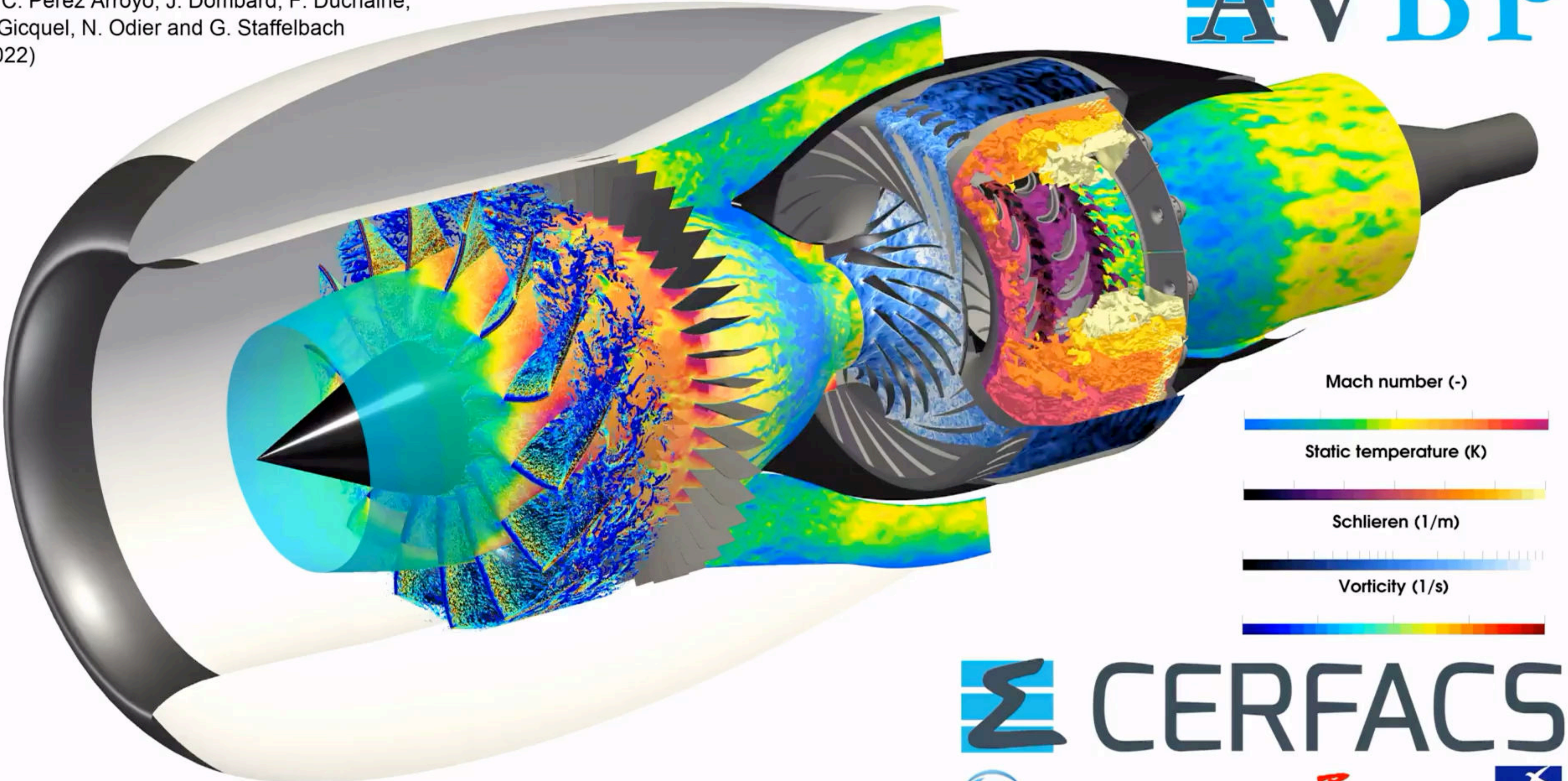
TOTAL cnes ONERA THE FRENCH AEROSPACE LAB

eDF SAFRAN AEROSPACE·DEFENCE·SECURITY AIRBUS METEO FRANCE

DGEN-380 engine Large Eddy Simulation at take-off conditions

by C. Pérez Arroyo, J. Dombard, F. Duchaine,
L. Gicquel, N. Odier and G. Staffelbach
(2022)

AVBP



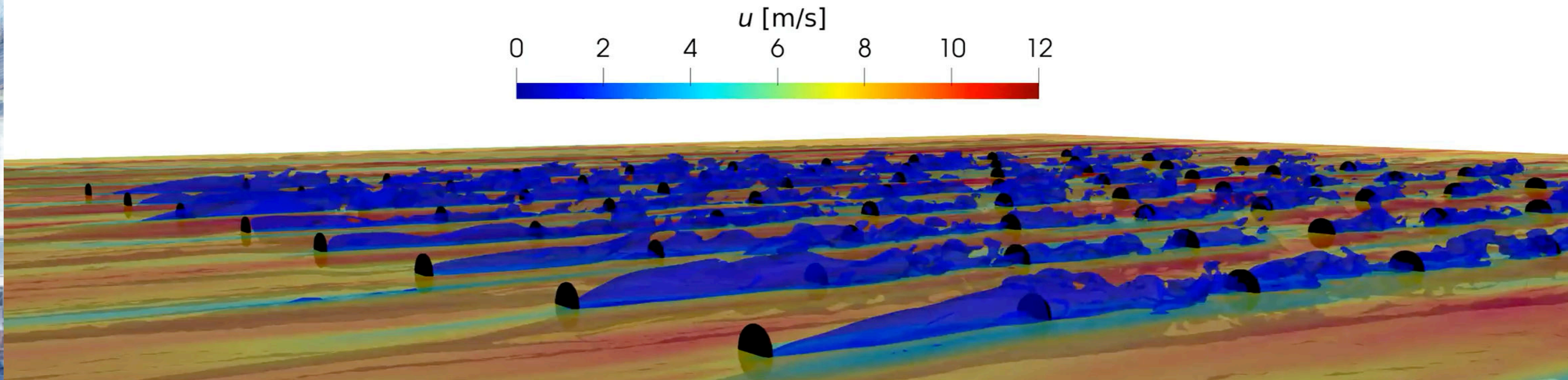
CERFACS

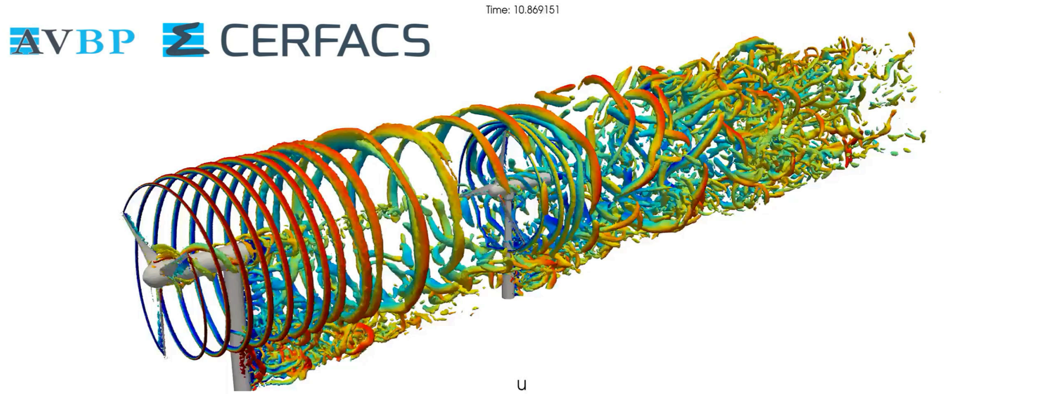
SAFRAN AKIRA dgac

These results benefitted of funding or developments from:
project ATOM (DGAC/SafranTech No 2018-39), PRACE (20th Call Project Access FULLEST), EXCELLERAT (H2020 823691), EPEEC (H2020 801051) and GENCI (A0122A06074).

Time: 1795 s

80 floating offshore wind turbine simulation - CERFACS - ADASTRA (CINES/GENCI)





Wall-modelled Large Eddy Simulation of two inline wind turbines, Dabas et al 2022

[1] Pierella, F., Krogstad, P.-Å. et Sætran, L. (2014). Blind test 2 calculations for two in-line model wind turbines where the downstream turbine operates at various rotational speeds. Renewable Energy, 70:62–77.

Questions ?

Can we use quantum computing for our applications or new applications ?

Look at quantum from an HPC expert with a skeptic point of

Main concerns :

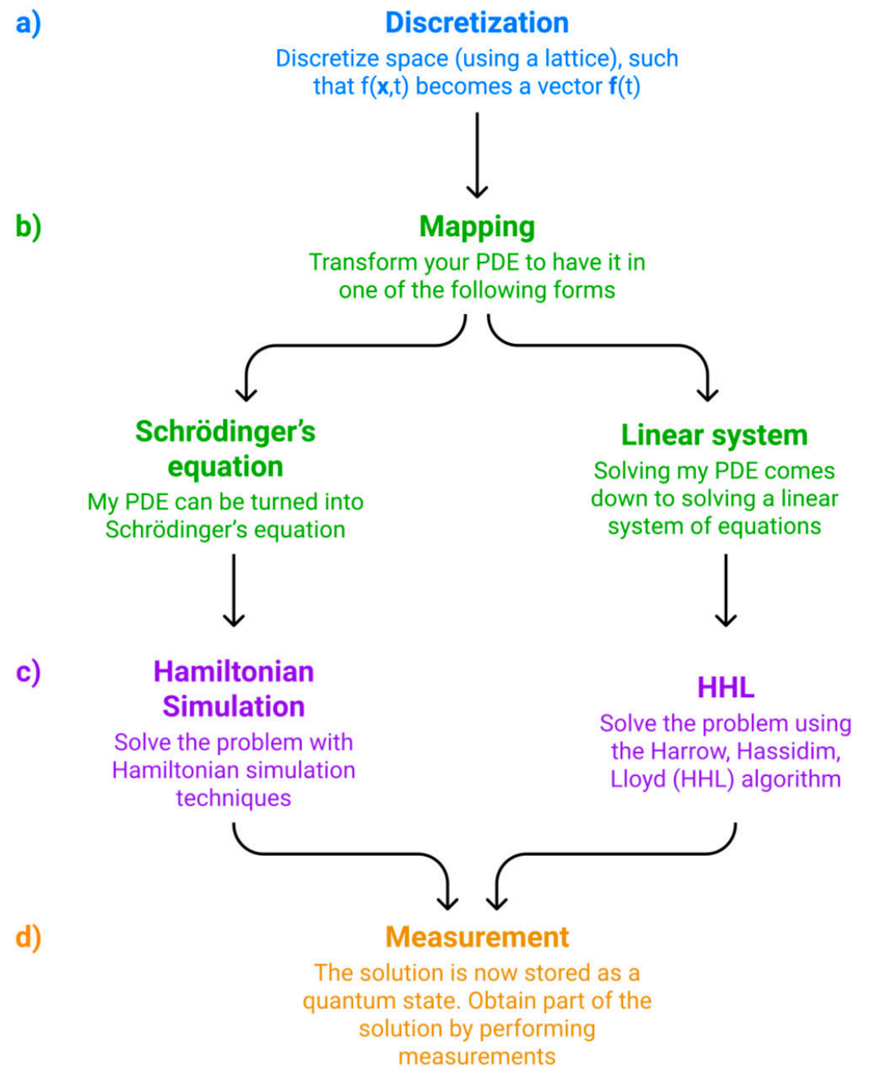
How to program ?

Can we estimate ressources ?

Can we optimise ?

Can we trust the results ?

Solving partial differential equations using Quantum



Quantum algorithms solving the Schrödinger equation are called Hamiltonian simulation algorithms.

Schrödinger's equation is a natural approach for quantum computing and easily identifiable in classical computing

- 1st foray into Quantum algorithms: hamiltonian simulation of the wave equation with dirichlet boundary conditions

Solving the wave equation using Quantum (simulators)

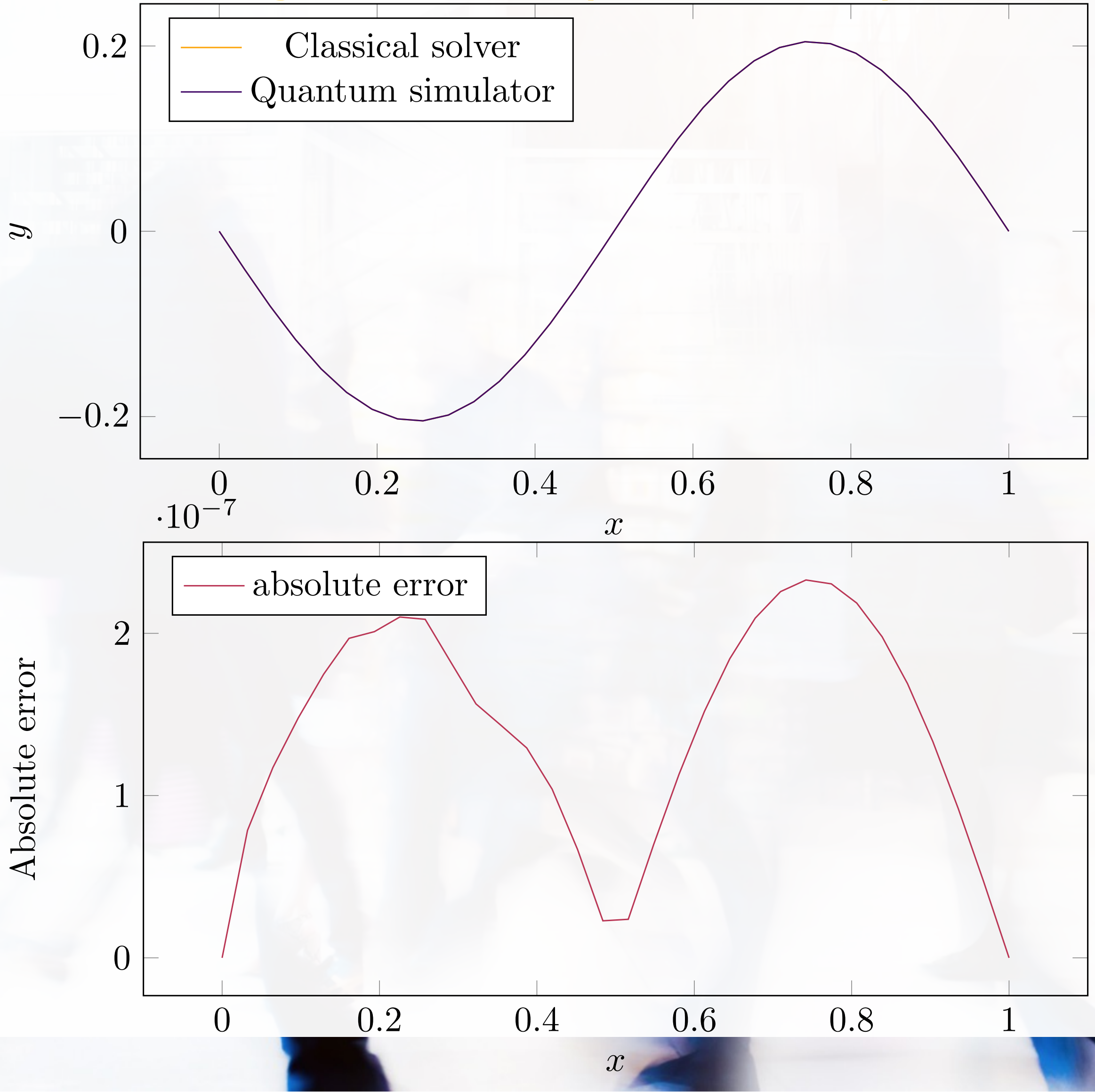
Wave equation

Find $\phi : ([0,1], R^+) \to \mathbb{R}$ such that

$$\frac{\partial^2}{\partial t^2}\phi(x,t) = \frac{\partial^2}{\partial x^2}\phi(x,t)$$

with Dirichlet boundary conditions

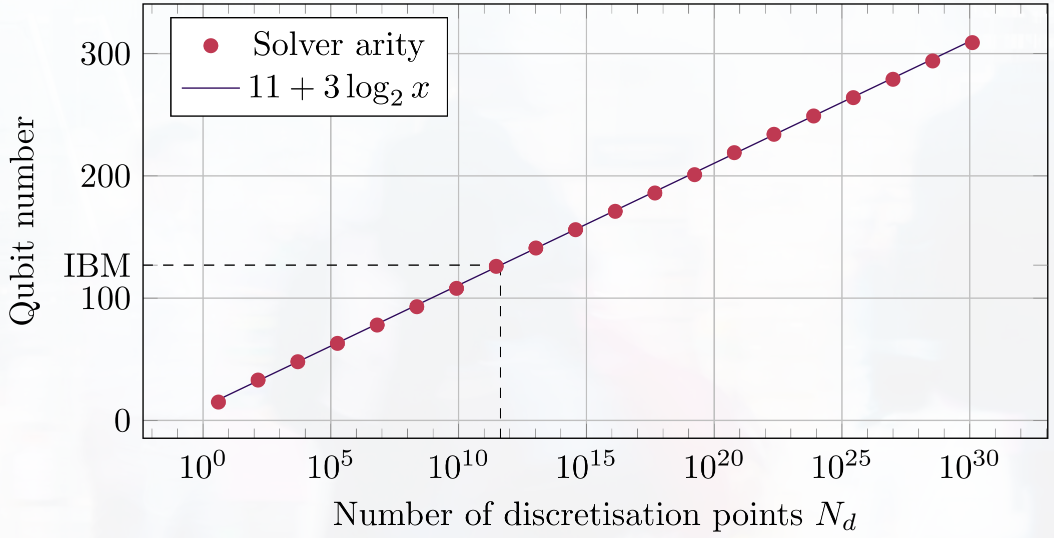
$$\frac{\partial}{\partial x}\phi(0,t) = \frac{\partial}{\partial x}\phi(1,t) = 0.$$

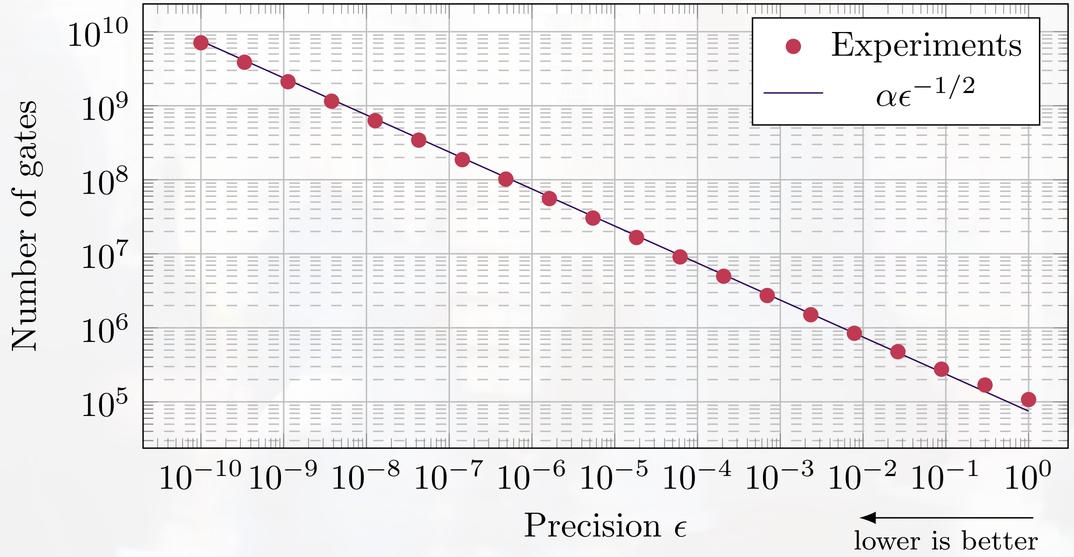


A. Suau, G. Staffelbach, and H. Calandra. 2021. Practical Quantum Computing: Solving the Wave Equation Using a Quantum Approach. ACM Transactions on Quantum Computing. https://doi.org/10.1145/3430030

Analysing the quantum circuit

Estimating the viability of the quantum solver and ressources requirements





- For the 1D solver , 100+ qubit would already be an important step ..
- But here we are talking about perfect qubits …
  - Single precision would require 10^8 gates … with as many possibilities of error.

Yes we can …

Take aways :

We can solve the problem

But it can be very costly and here we are estimating perfect hardware

Compilation of the quantum circuit is very slow (> 1h)

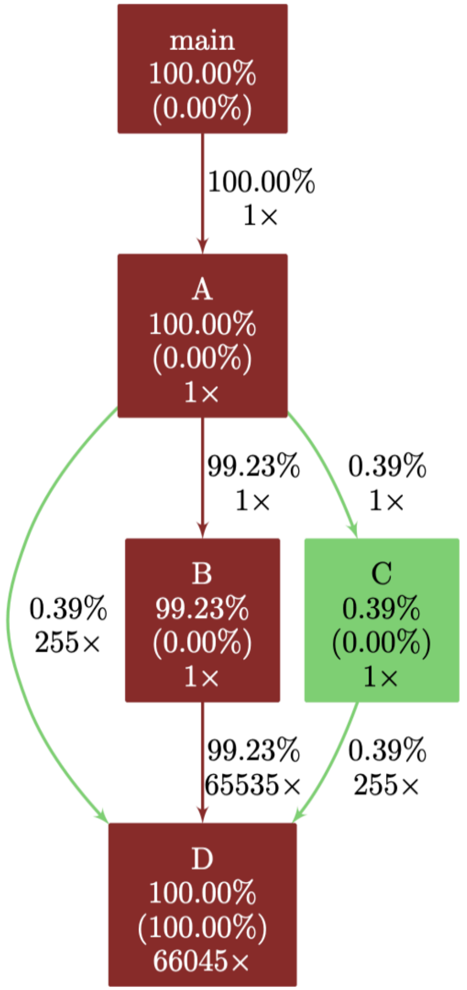
How do we optimise ?

How do we debug ?

From classical to Quantum profiling

Profiling in classical HPC :

1. Benchmark the program.
2. Isolate portions of the program that takes a significant amount of resources
3. Improve the isolated portions.
4. Come back to step 1. until desired performance is obtained.



Towards quantum profiling

Existing approaches in classical computing provide a quick and simple view of the call graph and the usage of ressources

Qprof : highly modular cross framework and fast profiling.

Adrien Suau, Gabriel Staffelbach, and Aida Todri-Sanial. 2022. Qprof: A gprof-Inspired Quantum Profiler. ACM Transactions on Quantum Computing. https://doi.org/10.1145/3529398

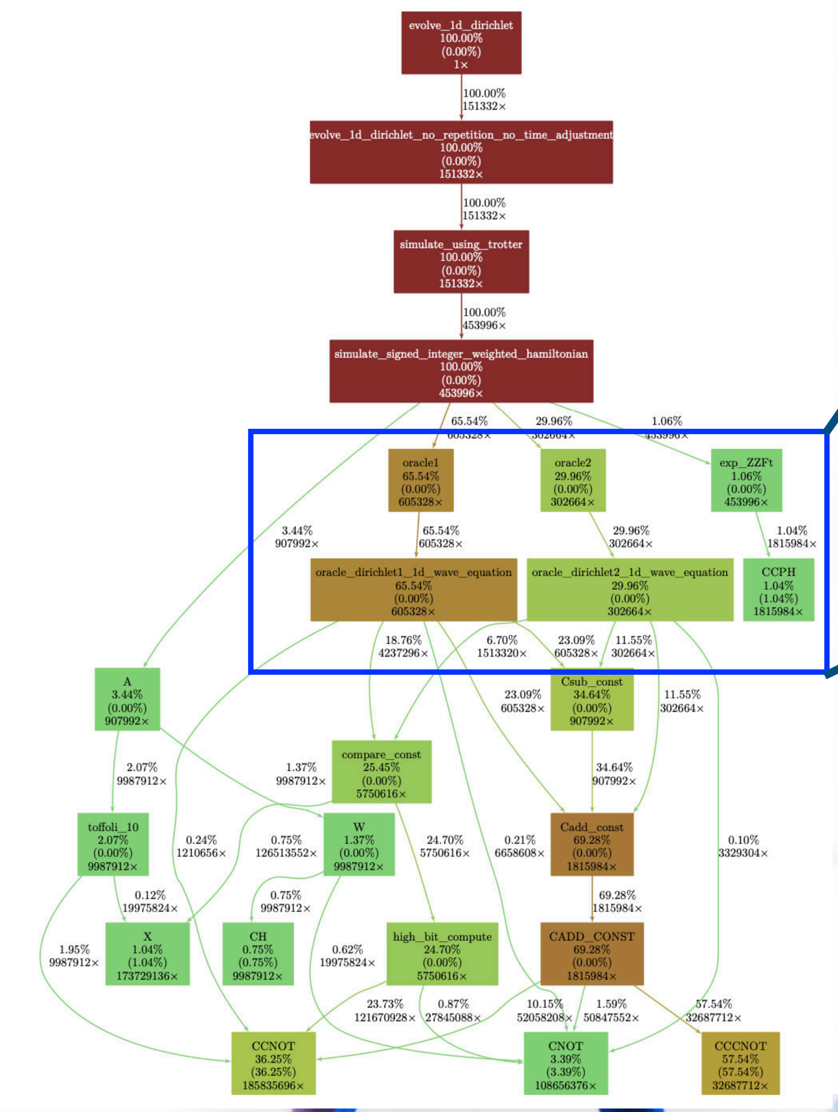
Application to QatHS

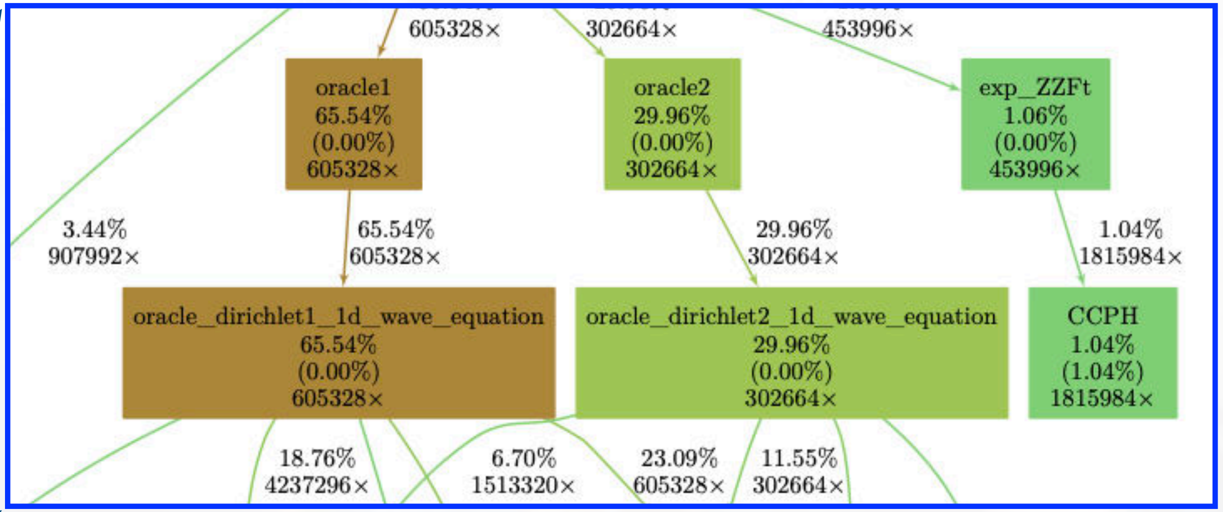
qaths: A Python 3 library of Hamiltonian simulation implementations

https://cerfacs.gitlab.io/qaths/

The qaths library aims at implementing various Hamiltonian Simulation algorithms with the help of qat, the Atos Python library for quantum computing.

Qprof on QatHS





Detailed view of the execution of the circuit.

Qprof takes as input the characteristics of the hardware provided by IBM/etc ..

Does not provide sampling execution but a simulation of the execution of the circuit => almost instantaneous

Oracles are costly (95.5% of the overall gate budget).

Addition is the most costly operation overall (nearly 70% of the overall gate budget).

Insights with Qprof :

Replacing Draper’s adder by an arithmetic-based adder (improved T-count and overall gate count).

Hand-optimised the oracle implementations.

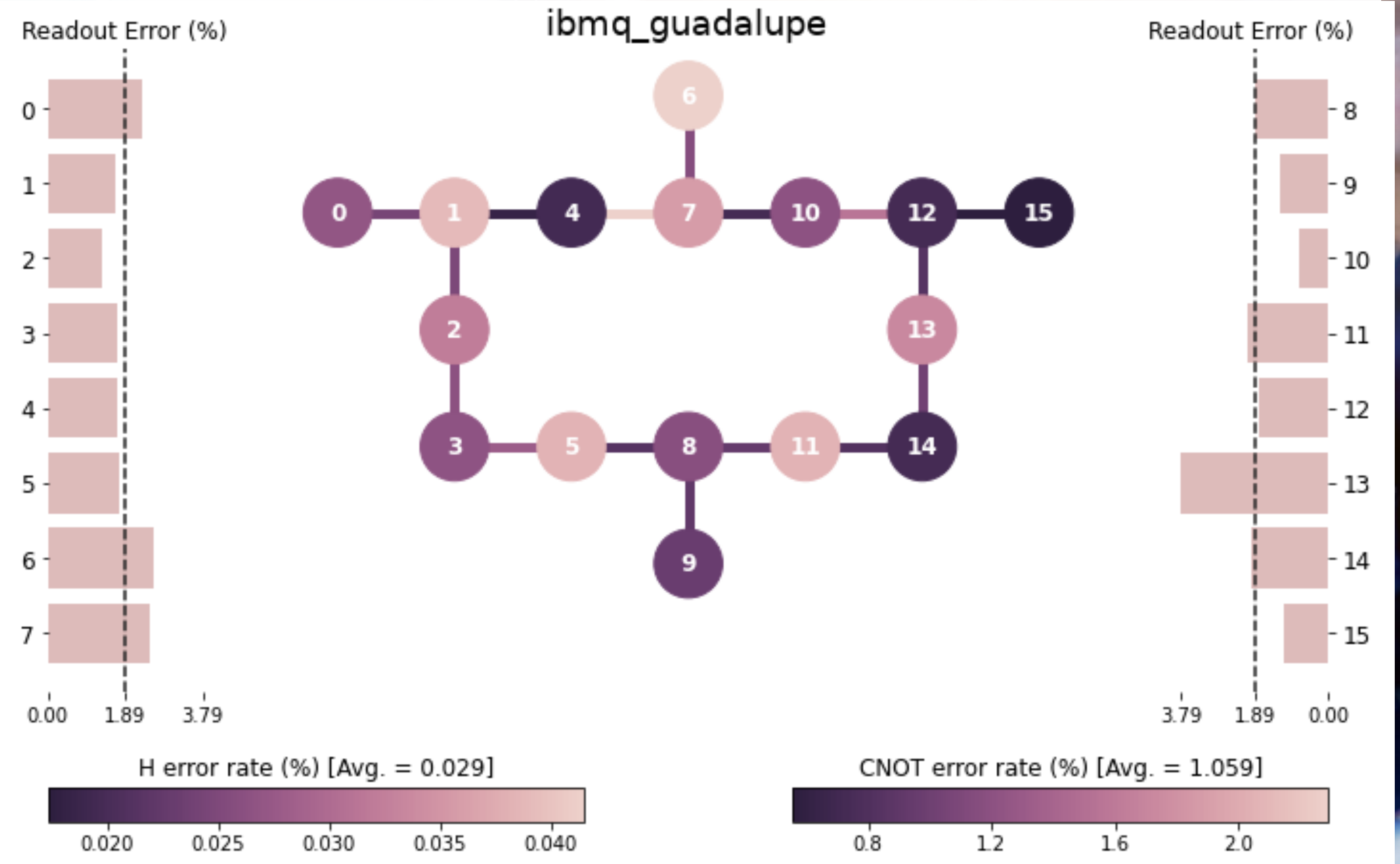
What about real hardware ?

Compiling the circuit checks :

- Hardware native quantum gates
- Respect topology

Optimisations :

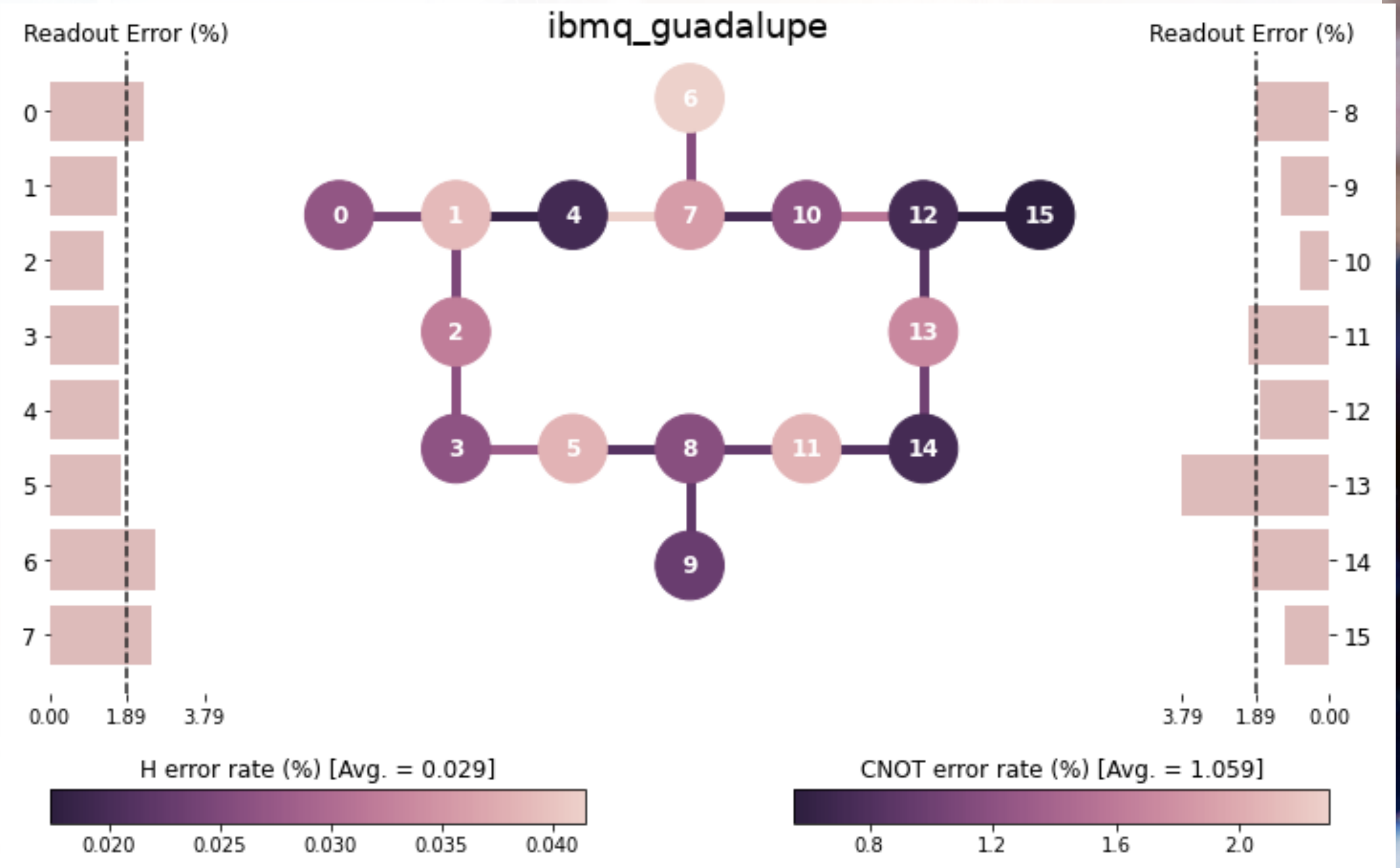
- Total number of qubits and gates
- Total number of specific gates (T/ CX)



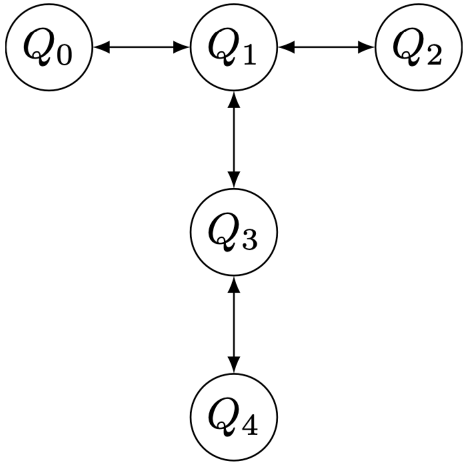
What about real hardware ?

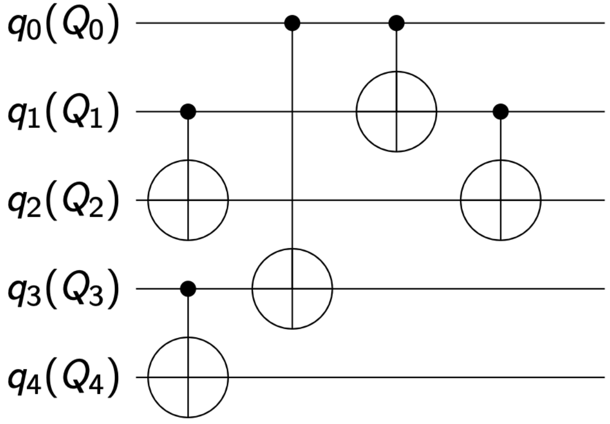
Optimisation is topology aware but not hardware aware:

- T1, T2 decoherence times
- Error rates
-Measurement error rates

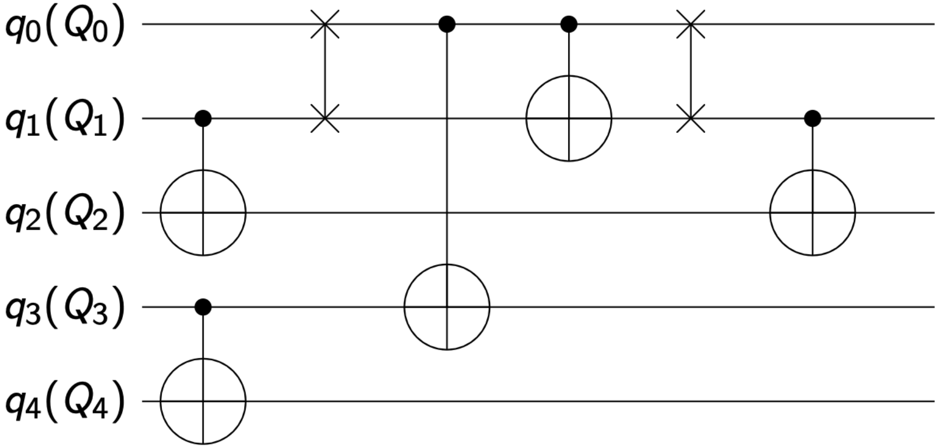


Adding hardware awareness to the compiler





SABRE algorithm checks the connectivity between qubits and adds swap gates to ensure the algorithm can be executed



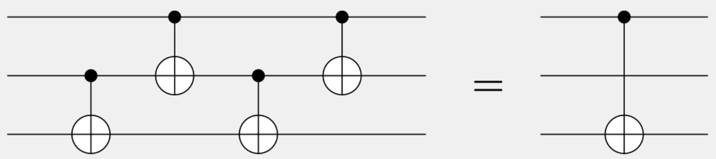
Gushu Li, et al. 2019. Tackling the Qubit Mapping Problem for NISQ-Era Quantum Devices. (ASPLOS '19). https://doi.org/10.1145/3297858.3304023

Hardware aware optimisation

Heuristic cost function:

$$D = \alpha_1 S + \alpha_2 \mathcal{E} + \alpha_3 T$$

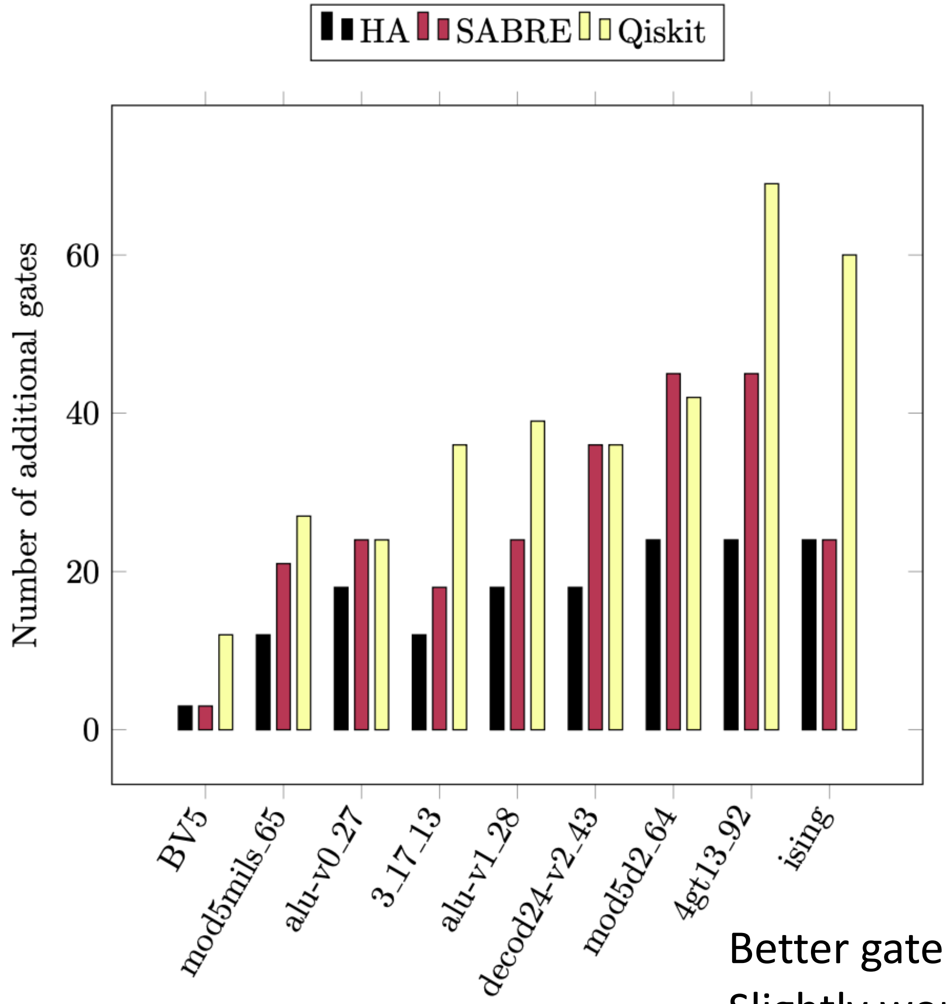
The Bridge gate

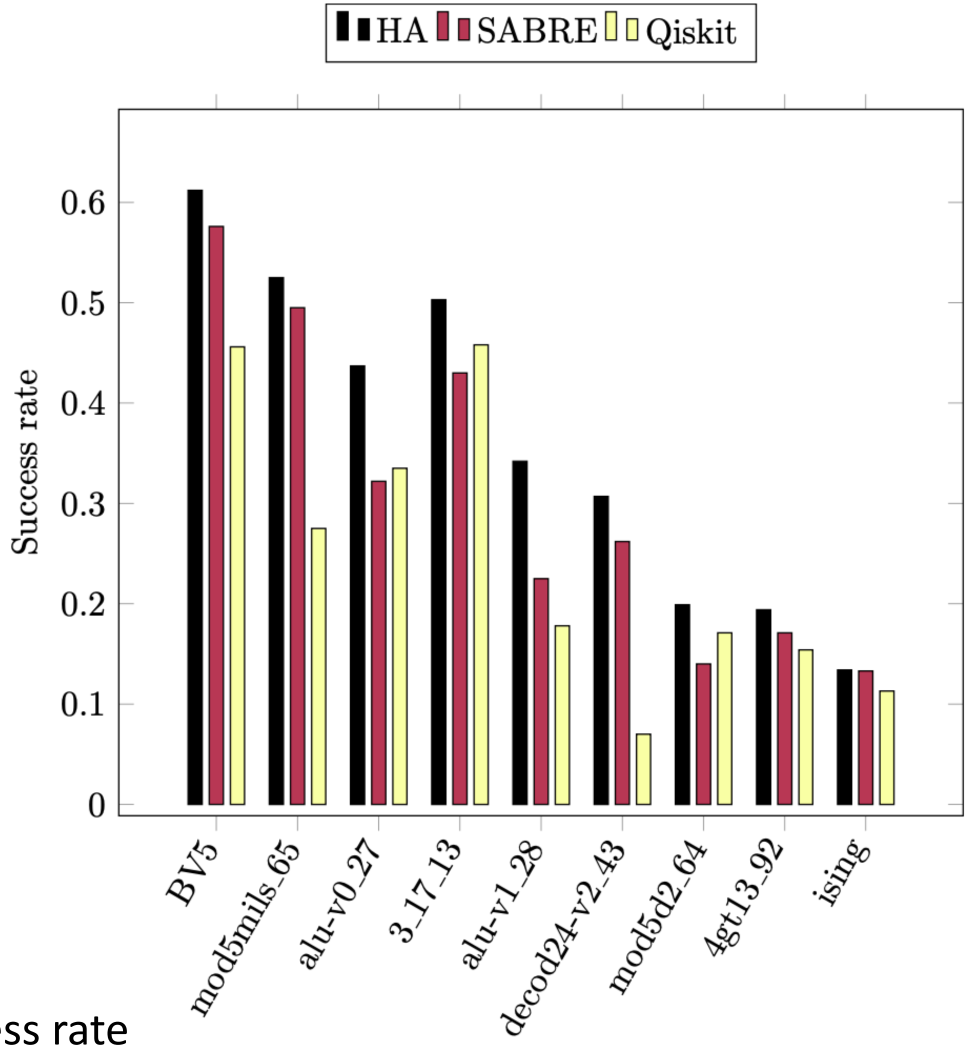


Introduce hardware related information into the cost heuristic and introduce the bridge gate to give more freedom to the optimizer

S. Niu, A. Suau, G. Staffelbach and A. Todri-Sanial, "A Hardware-Aware Heuristic for the Qubit Mapping Problem in the NISQ Era," in IEEE TQC, vol. 1, pp. 1-14, 2020, doi: 10.1109/TQE.2020.3026544.

HA vs SABRE vs Qiskit

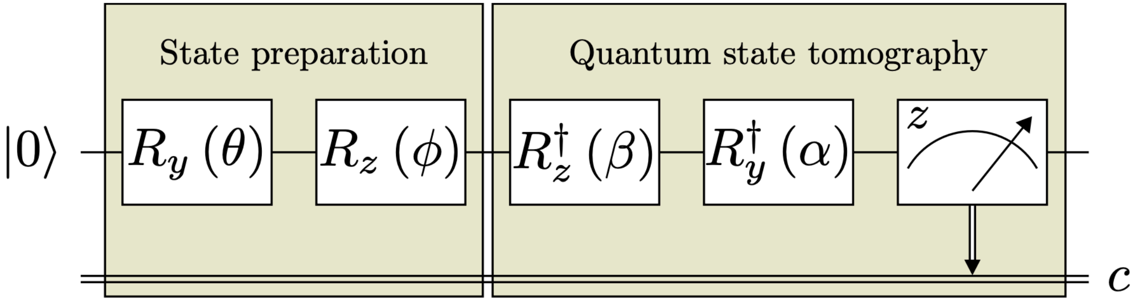




Better gate count

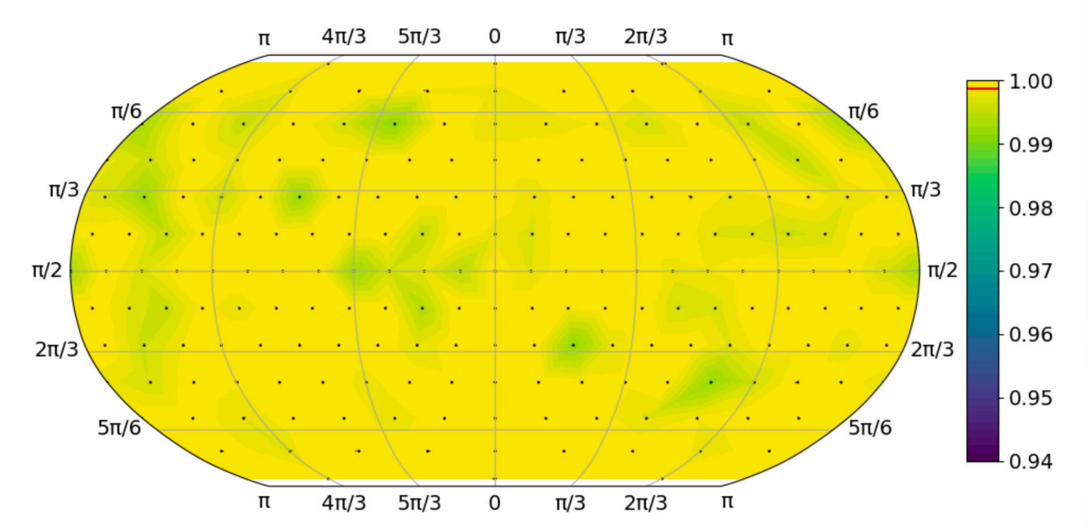
Slightly worse Success rate

Understanding the source of error

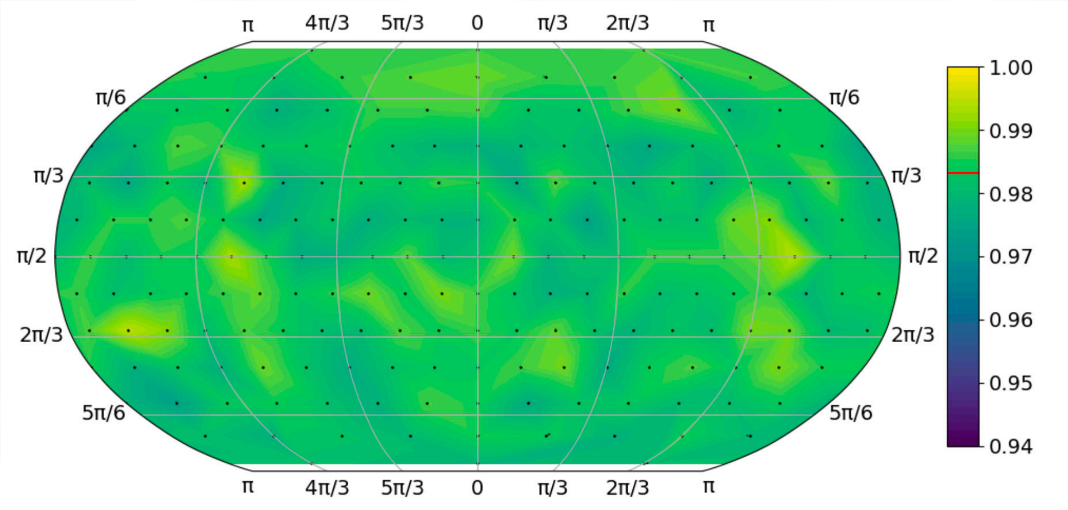


Use simple circuit to evaluate computational error

Perfect simulator - 20000 shots



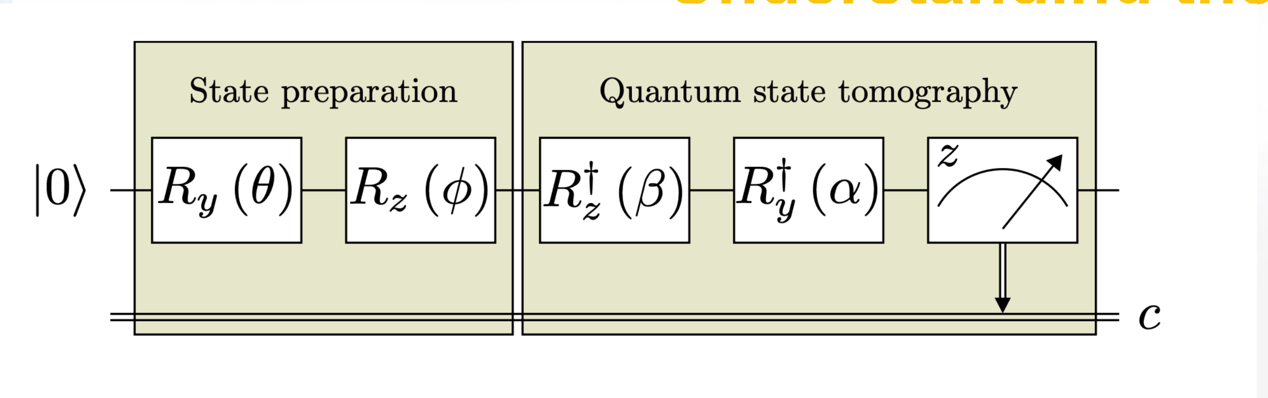
Noisy simulator - ibm_lagos calibrations - 20000 shots



Vector Field Visualization of Single-Qubit State Tomography, Suau et al.

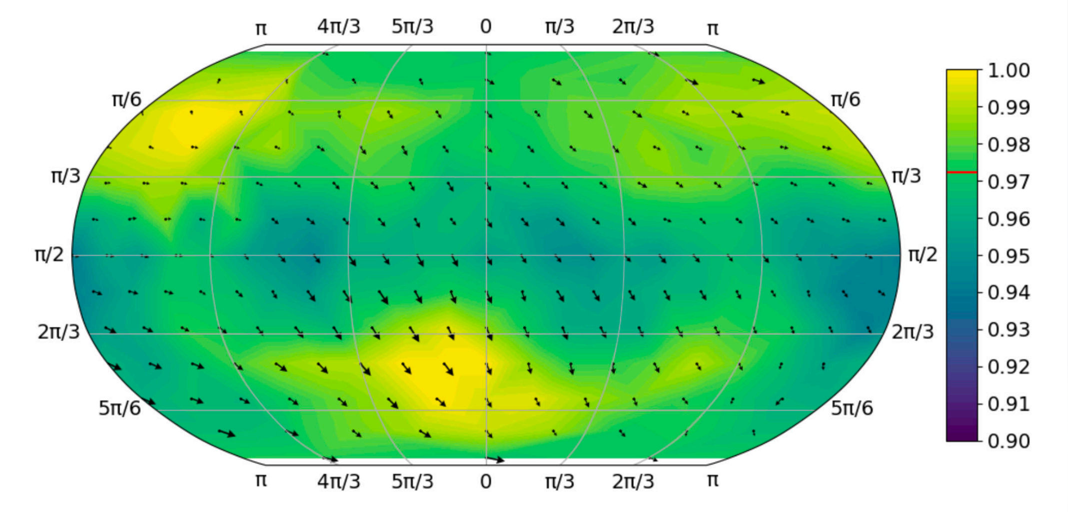
https://arxiv.org/pdf/2205.02483

Understanding the source of error

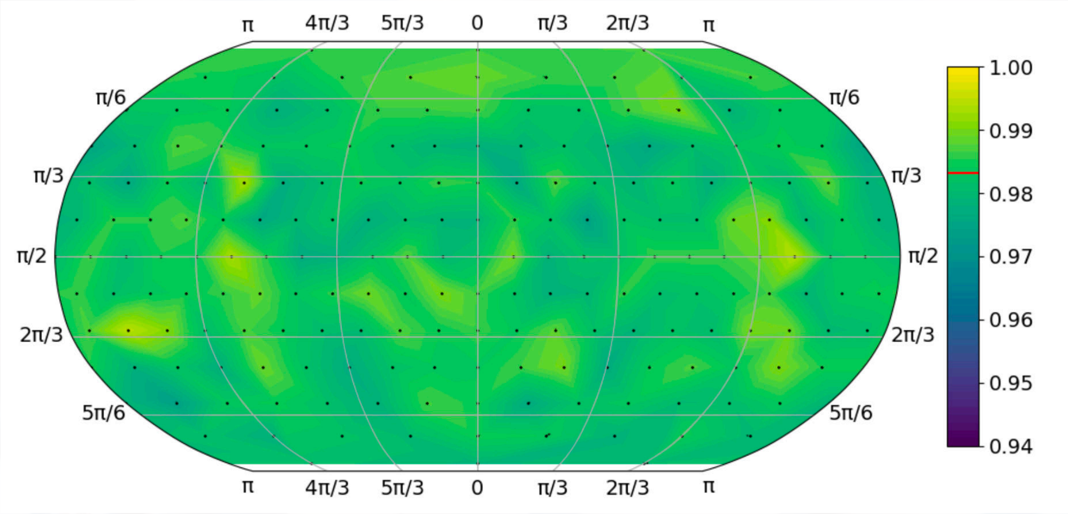


Use simple circuit to evaluate computational error

IBM Lagos hardware - 20000 shots - 0 delay



Noisy simulator - ibm_lagos calibrations - 20000 shots



Takeaways

From an HPC perspective the technology is very interesting but needs to be manipulated carefully

Some inpactful resultats already here but an ever evolving landscape

Hamiltonian simulations are not the only way to tackle the problem