Benchmarking neutral atom-based quantum processors at scale

Andrea B. Rava[1,2], Kristel Michielsen[1,3], J. A. Montañez-Barrera[1]

[1]Jülich Supercomputing Centre, Forschungszentrum Jülich, D-52425 Jülich, Germany
[2]RWTH Aachen University, 52056 Aachen, Germany
[3]Department of Computer Science, University of Cologne, 50931 Cologne, Germany

Motivations

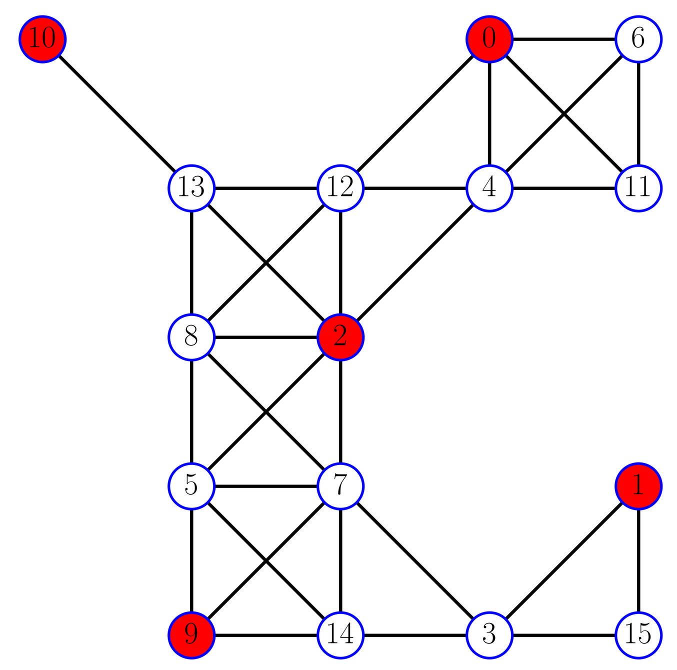
1. Neutral-atom quantum processors have rapidly advanced as scalable platforms for analog and digital quantum computation
2. Benchmarking at scale is essential to quantify their algorithmic performance beyond gate fidelities, especially for analog, Rydberg-based computation.
3. We introduce a scalable, problem-based benchmark using the Quantum Adiabatic Algorithm on Maximal Independent Set instances to assess and compare neutral-atom QPUs.
4. We have created a repository with up to 1000-qubits problems and we solve instances up to 102 qubits on Fresnel and Aquila.

Methods

1. Find optimal solutions for Maximal Independent Set (MIS) problem of Diagonal-connected Unit-disk Grid Graphs (DUGG)

$$C(\vec{x}) = -\sum_{i=1}^{N} x_i + \infty \sum_{(i,j)\in E} x_i x_j, \quad x_i \in \{0,1\}$$

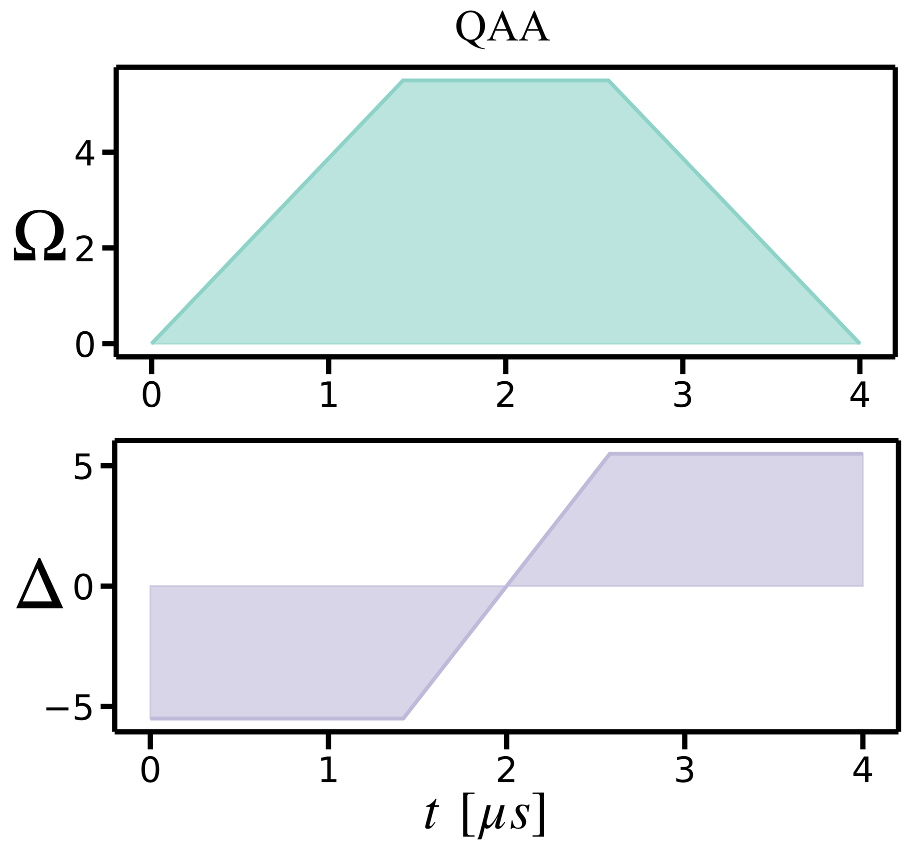
DUGG graph with MIS solution



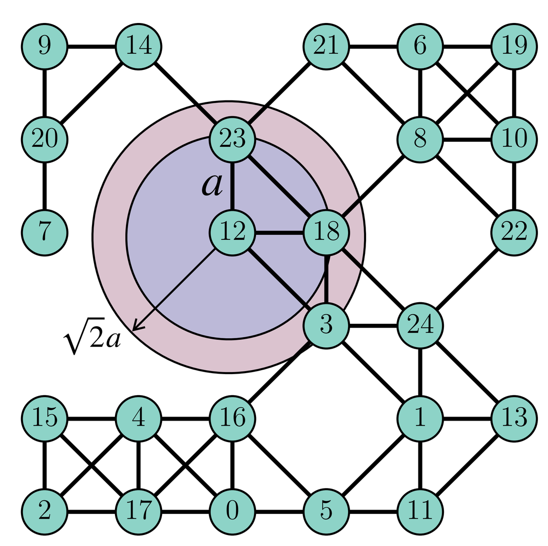
2. Define Quantum Adiabatic and Qauntum Approximate Optimization algorithms (QAA) and (QAOA) on neutral atoms devices

$$\hat{H}_{\text{Ryd}} = \sum_{i}^{N}\left[\frac{\Omega(t)\hbar}{2}\hat{\sigma}_i^x - \Delta(t)\hbar\hat{n}_i + \sum_{j<i}\frac{C_6}{R_{ij}^6}\hat{n}_i\hat{n}_j\right]$$

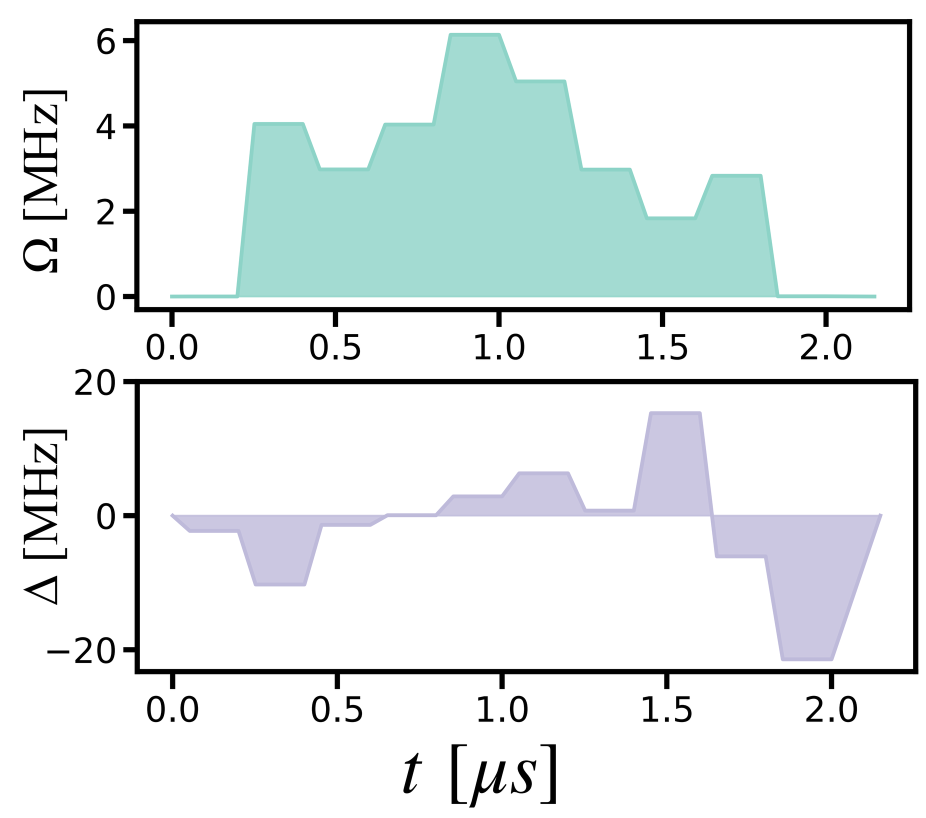
QAA



DUGG graph



QAOA



$$C_6^{\text{Fresnel}}/a_{\text{Fresnel}}^6 = C_6^{\text{Aquila}}/a_{\text{Aquila}}^6$$

Comparison neutral atoms devices' specs

	Fresnel	Aquila
C_6	865723 MHz	5420441 MHz
max num. atoms	60	256
$a_{\min}$	5 μm	4 μm
$t_{\max}$	6 μs	4 μs
$\Omega_{\max}$	$2 \times 2\pi$ MHz	$2.5 \times 2\pi$ MHz
$\lvert\Delta\rvert_{\max}$	$7.75 \times 2\pi$ MHz	$20 \times 2\pi$ MHz

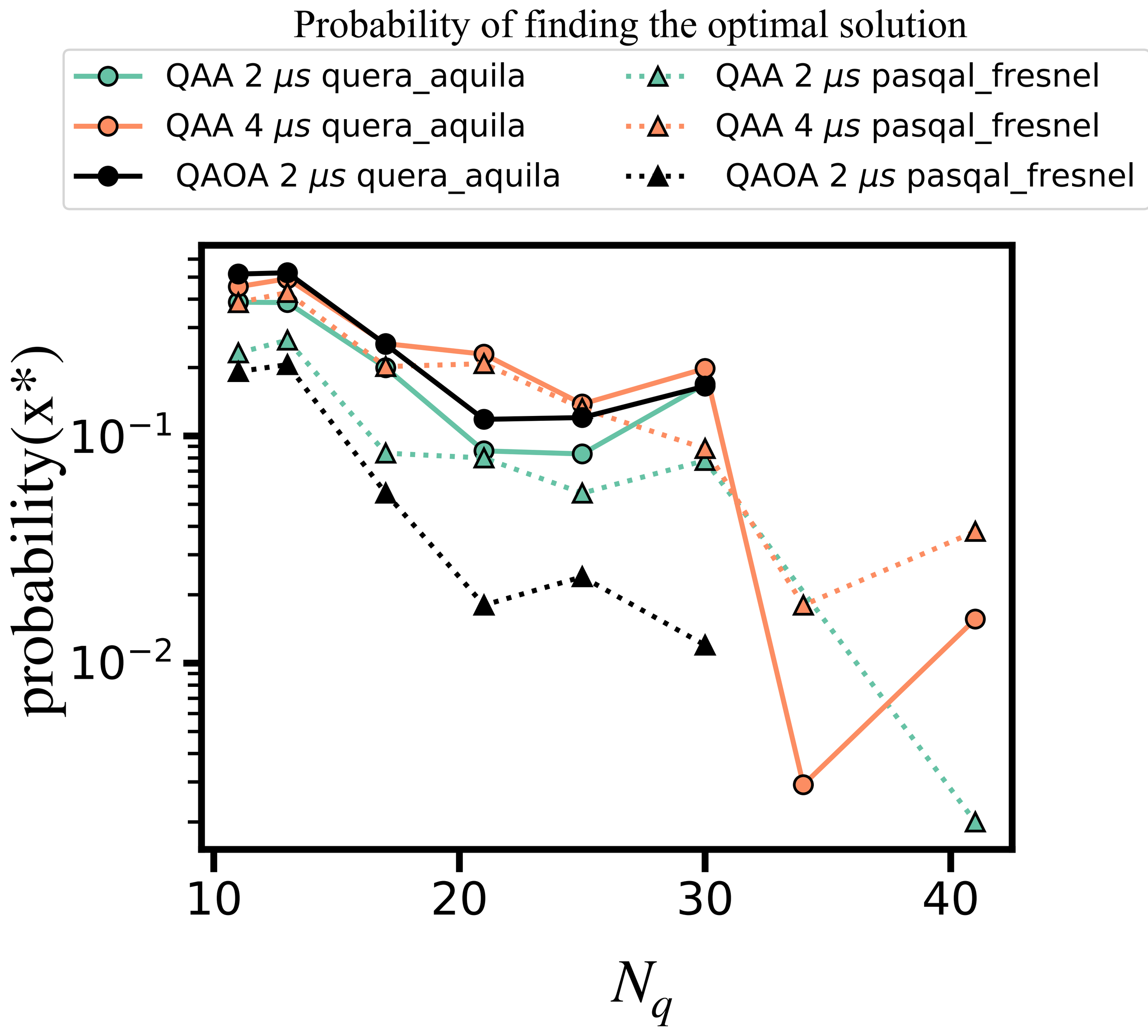
3. Implement the two algorithms on QuEra's Aquila and Pasqal's Fresnel neutral atom devices. Where we had run each experiments three times, taking 200 samples each time.

Results

Raw data

- The 4 μs QAA yields the highest success probability on both platforms.
- For equal durations (2 μs), QAOA outperforms QAA on Aquila, while on Fresnel, QAOA performs worse than both QAA and Aquila.
- Overall, Fresnel underperforms relative to Aquila, except for the 4 μs QAA at larger system sizes (N > 30).

Probability of finding the optimal solution

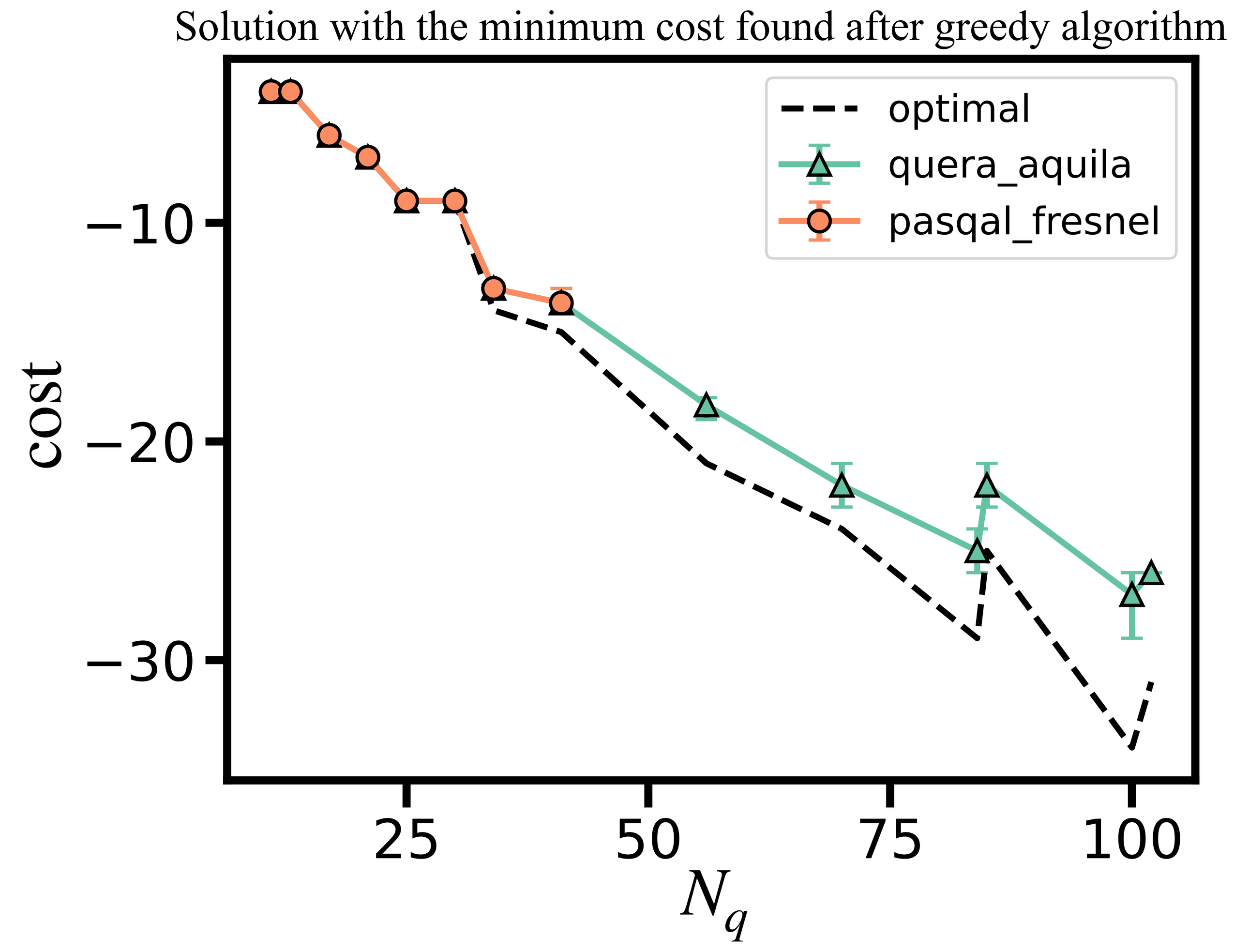


Post-processed (greedy algorithm)

perform bitflips on obtained bitstrings and keep if result improves

- After greedy optimization, both devices achieve near-optimal results.
- Aquila's performance degrades with increasing system size.

Solution with the minimum cost found after greedy algorithm



Conclusions

- Equivalent interaction scaling between devices allows meaningful cross-platform benchmarking.
- Aquila generally outperforms Fresnel, though Fresnel is more robust for large-N, long-evolution QAA runs.
- Greedy post-processing greatly enhances performance across both platforms.
- QAOA does not systematically outperform QAA, but remains valuable for testing non-uniform control schedules.

This project has received funding from the European High-Performance Computing Joint Undertaking (JU) under grant agreement No 101018180. The JU receives support from the European Union's Horizon 2020 research and innovation programme and Germany, France, Italy, Ireland, Austria and Spain in equal parts.