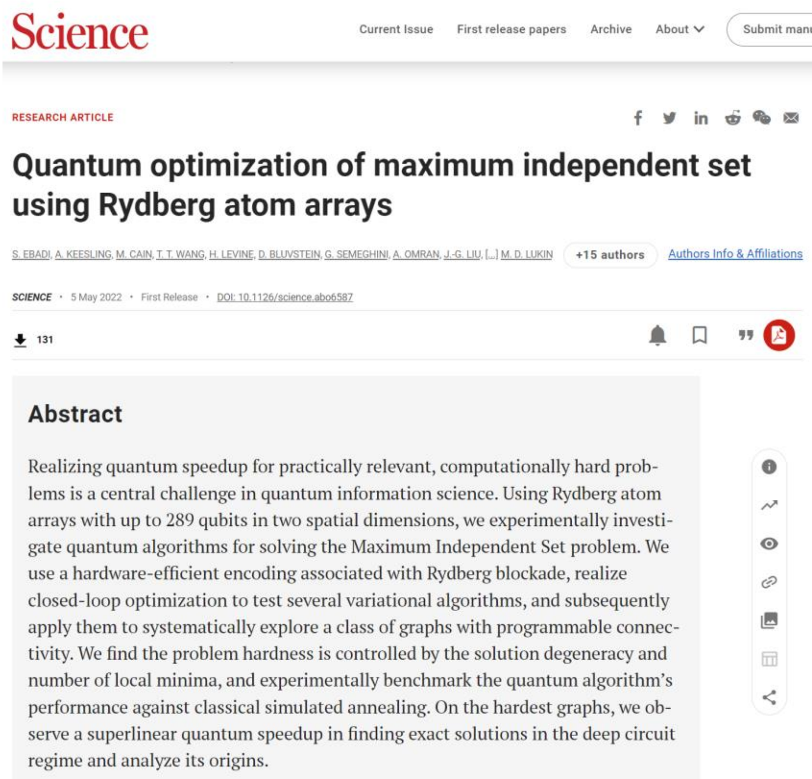
Introduction to quantum computing in the circuit model

Lecture plan

- Introduction – state of technology, complexity of quantum algorithms, computational models
- Quantum circuit model
  - Qubits
  - Qubit measurements
  - Unitary transformations and quantum gates
- Examples of quantum algorithms: Grover and Shor algorithms
- How to use quantum computing in your research
- Quantum Algorithm Zoo
- Other issues in quantum computing

Why quantum computing?

- Because quantum computers are already being built
- Because quantum computers can perform certain tasks faster (in terms of computational complexity) than classic computers


Science

Current Issue First release papers Archive About Submit man

RESEARCH ARTICLE

Quantum optimization of maximum independent set using Rydberg atom arrays

S. EBADI, A. KEESLING, M. CAIN, T. T. WANG, H. LEVINE, D. BLUVSTEIN, G. SEMEGHINI, A. OMRAN, J.-G. LIU, [...] M. D. LUKIN +15 authors Authors Info & Affiliations

SCIENCE • 5 May 2022 • First Release • DOI: 10.1126/science.abo6587

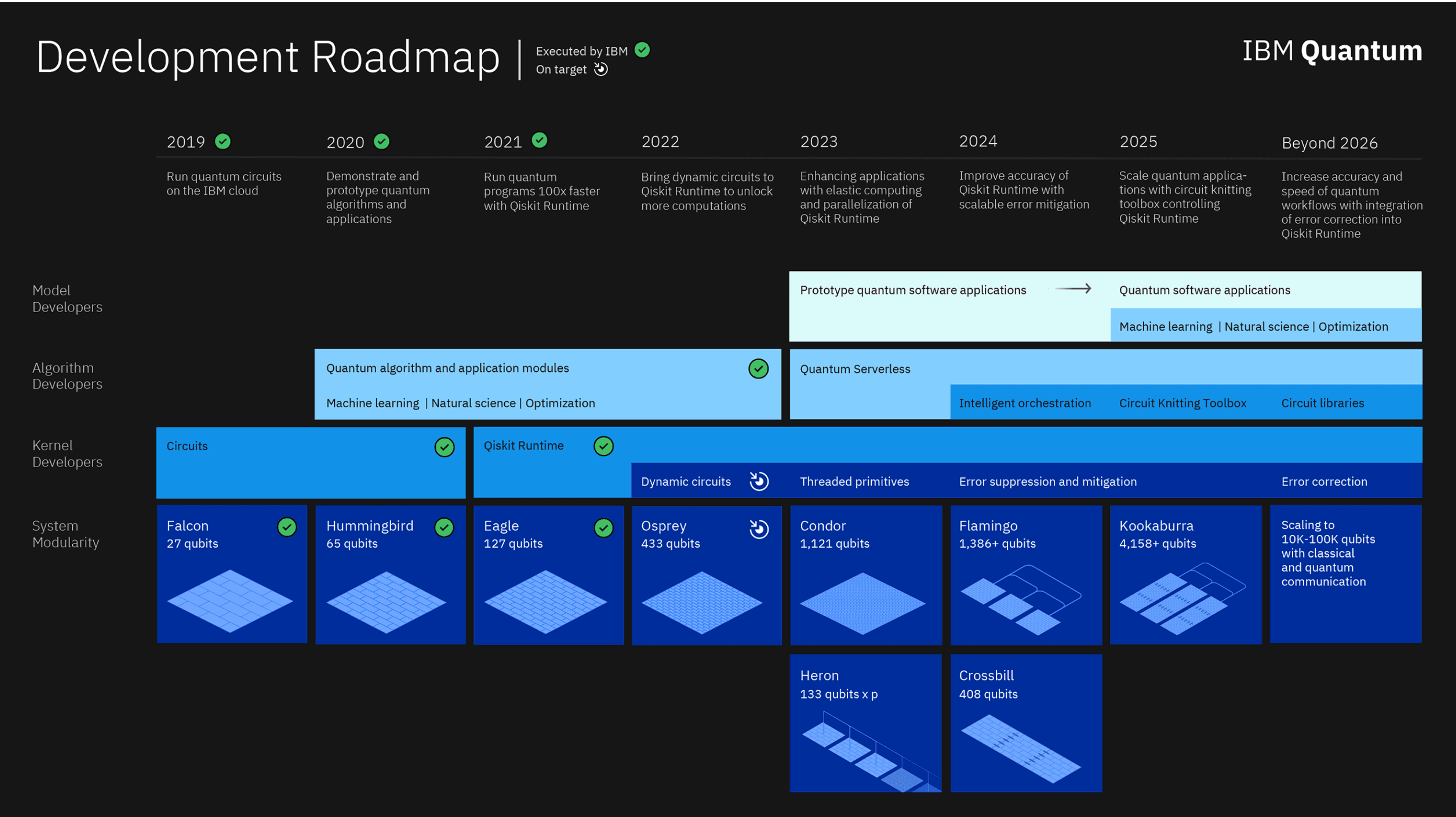
131

Abstract

Realizing quantum speedup for practically relevant, computationally hard problems is a central challenge in quantum information science. Using Rydberg atom arrays with up to 289 qubits in two spatial dimensions, we experimentally investigate quantum algorithms for solving the Maximum Independent Set problem. We use a hardware-efficient encoding associated with Rydberg blockade, realize closed-loop optimization to test several variational algorithms, and subsequently apply them to systematically explore a class of graphs with programmable connectivity. We find the problem hardness is controlled by the solution degeneracy and number of local minima, and experimentally benchmark the quantum algorithm's performance against classical simulated annealing. On the hardest graphs, we observe a superlinear quantum speedup in finding exact solutions in the deep circuit regime and analyze its origins.

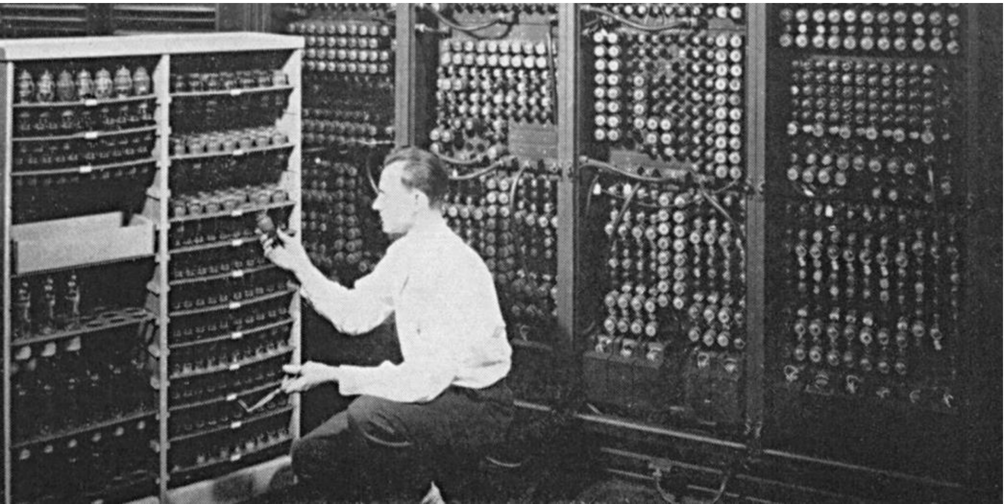
Quantum computers

- Top Quantum Computing Companies
  - IBM – 433 qubits announced in 2022, 1121 in 2023, 4158 (3 x 1386) in 2025
  - Google - 54-qubit processor 'Sycamore', October 23, 2019, Quantum Supremacy – a calculation that would require 10,000 years on the fastest supercomputer in 200s
  - QCI
  - Xanadu
  - Microsoft Azure Quantum
  - D-Wave Systems
- Practical applications will require at least thousands of qubits


Development Roadmap
Executed by IBM
On target
IBM Quantum
2019
2020
2021
2022
2023
2024
2025
Beyond 2026
Run quantum circuits on the IBM cloud
Demonstrate and prototype quantum algorithms and applications
Run quantum programs 100x faster with Qiskit Runtime
Bring dynamic circuits to Qiskit Runtime to unlock more computations
Enhancing applications with elastic computing and parallelization of Qiskit Runtime
Improve accuracy of Qiskit Runtime with scalable error mitigation
Scale quantum applications with circuit knitting toolbox controlling Qiskit Runtime
Increase accuracy and speed of quantum workflows with integration of error correction into Qiskit Runtime
Model Developers
Prototype quantum software applications
Quantum software applications
Machine learning | Natural science | Optimization
Algorithm Developers
Quantum algorithm and application modules
Machine learning | Natural science | Optimization
Quantum Serverless
Intelligent orchestration
Circuit Knitting Toolbox
Circuit libraries
Kernel Developers
Circuits
Qiskit Runtime
Dynamic circuits
Threaded primitives
Error suppression and mitigation
Error correction
System Modularity
Falcon
27 qubits
Hummingbird
65 qubits
Eagle
127 qubits
Osprey
433 qubits
Condor
1,121 qubits
Flamingo
1,386+ qubits
Kookaburra
4,158+ qubits
Scaling to 10K-100K qubits with classical and quantum communication
Heron
133 qubits x p
Crossbill
408 qubits

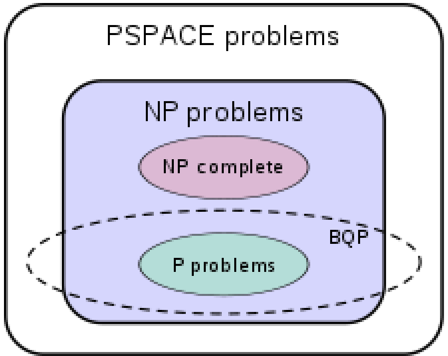
Quantum computers

- Quantum effects occur only in strictly isolated systems
- Necessary insulation from the environment by orders of magnitude greater than thermal insulation



The complexity of quantum algorithms

- Quantum computers can solve certain tasks more efficiently (in terms of computational complexity) than classic ones, e.g. acceleration:
  - Exponential: Electrical Resistance, simulation of quantum computers?
  - Superpolynomial: integer factorization, Pattern matching, String Rewriting, Matrix Powers, Solving differential equations
  - Polynomial: Black-box search, Matrix order, Constraints Satisfaction
- So far, it has not been possible to find an effective quantum algorithm for any NP-complete problem

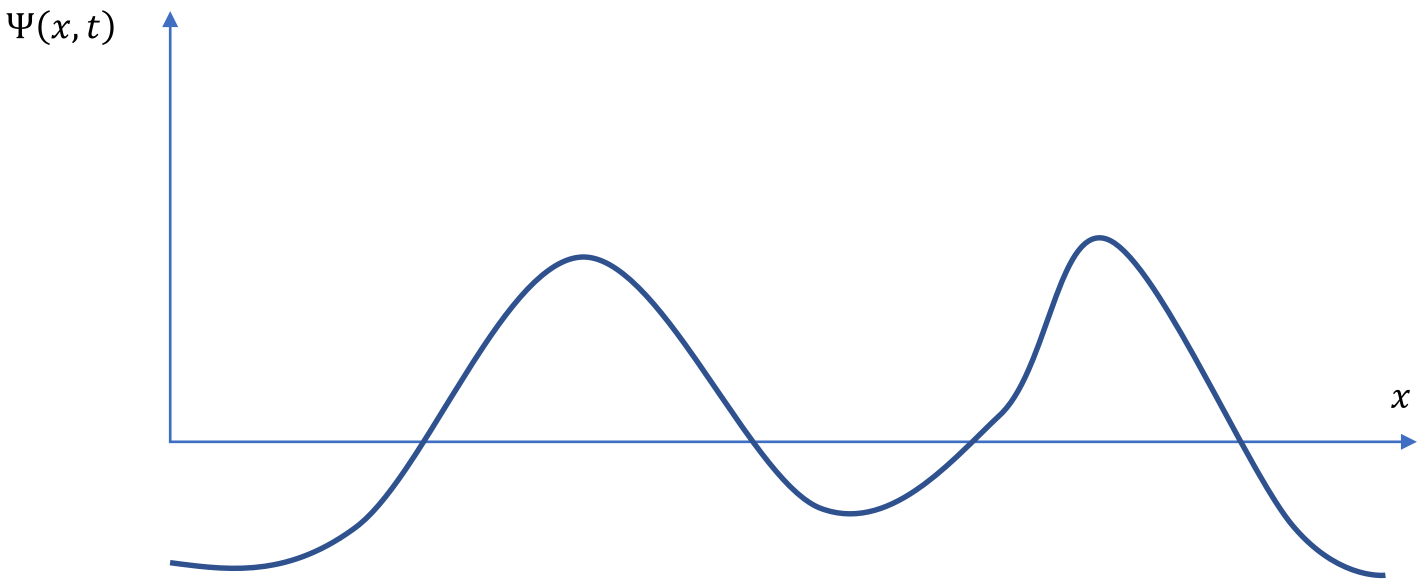


At least two quantum computing models

- **Quantum gate array, circuit model**
- Adiabatic Quantum Computation (quantum annealing), e.g. D-Wave

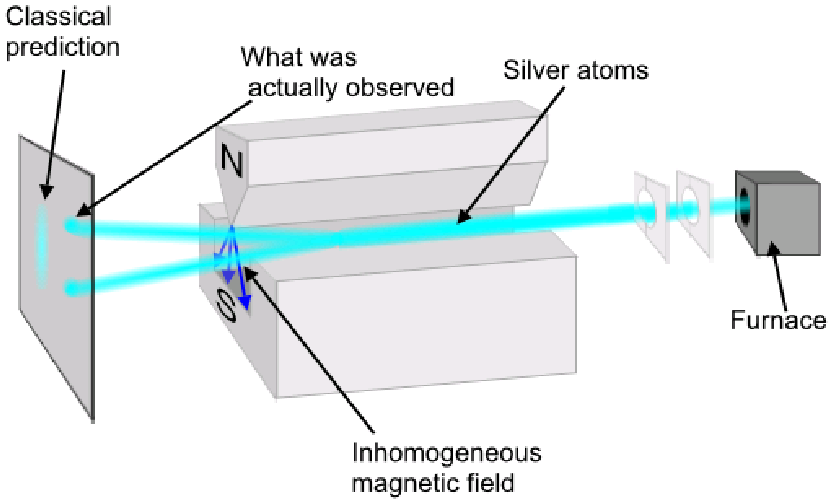
Quantum mechanics

- Wave function



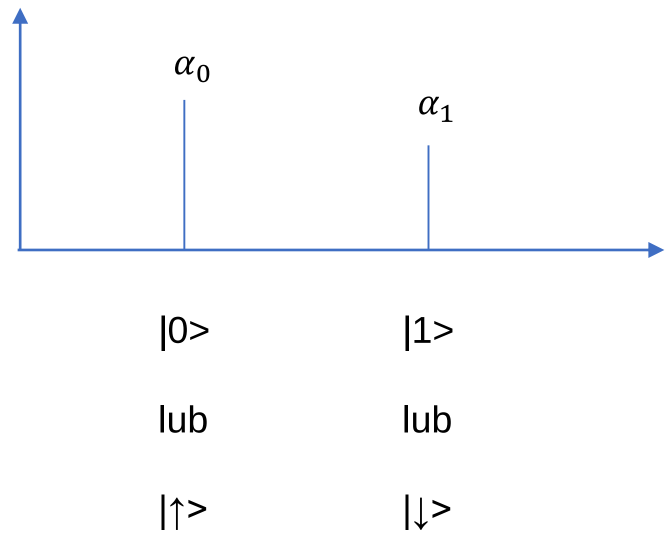
Qubit – a quantum system with two base states

- Examples
  - Atom/electron spin
  - Position of the electron in two orbits
- Stern-Gerlach experiment



Qubit – a quantum system with two base states

- α_0, α_1 – amplitudes of the base states, complex numbers
- $|\alpha_0|^2 + |\alpha_1|^2 = 1$
- Qubit is in a single state – single superposition of base states
- Every linear combination of quantum states is a quantum state



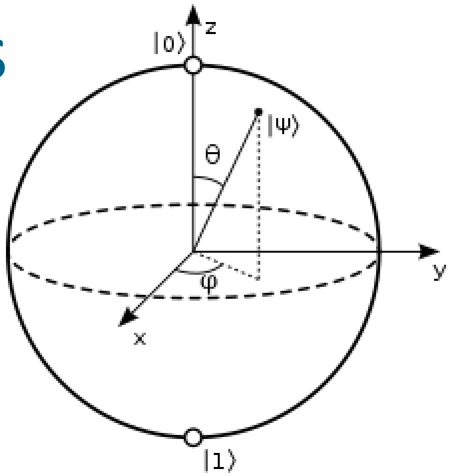
Qubit – a quantum system with two base states

- Dirac notation, „bra-ket”:
  - $\alpha_0|0> +\alpha_1|1>$
  - $\alpha_0|\uparrow> + \alpha_1|\downarrow>$
- Matrix notation:
  - $\begin{bmatrix}\alpha_0\\ \alpha_1\end{bmatrix}$

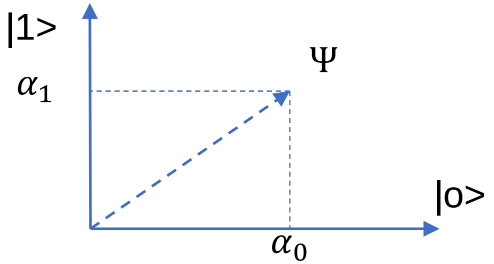
- $\begin{bmatrix}1\\ 0\end{bmatrix} = |0>$, $\begin{bmatrix}0\\ 1\end{bmatrix} = |1>$ –base states
- $\begin{bmatrix}\alpha_0\\ \alpha_1\end{bmatrix}$ - superposition of base states

Qubit visualizations

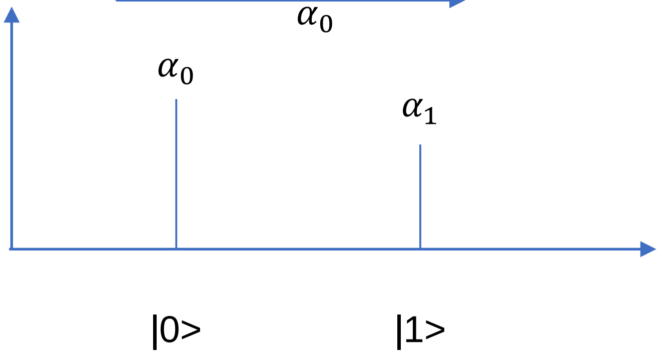
- Bloch sphere



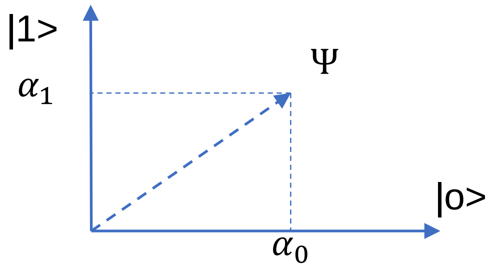
- Hilbert space



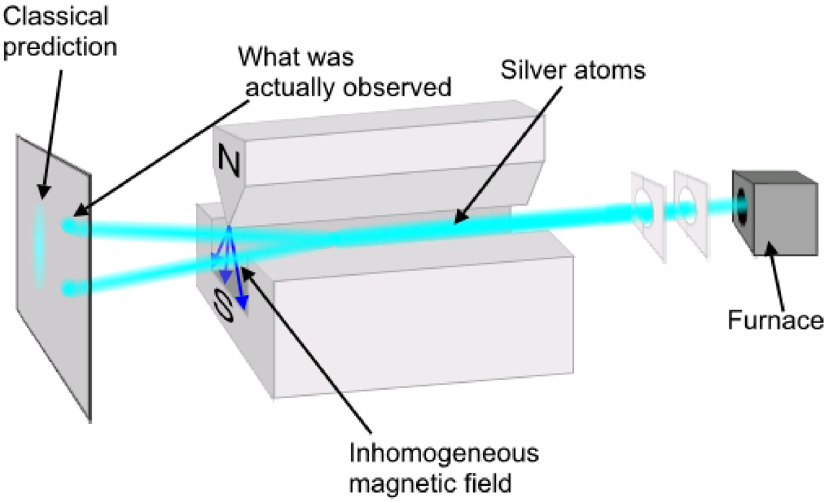
- Amplitudes of base state



Qubit measurement



- $\alpha_0|0> + \alpha_1|1> \Leftrightarrow \begin{bmatrix} \alpha_0 \\ \alpha_1 \end{bmatrix}$
- With probability $|\alpha_0|^2$ state 0> will be observed
- With probability $|\alpha_1|^2$ state 1> will be observed
- $|\alpha_0|^2 + |\alpha_1|^2 = 1$
- After the measurement, the qubit is in one of the base states (!)



Quantum state of many qubits

- Interacting qubits form a single quantum system
- 2 qubits 4 base states
- 3 qubits 8 base states
- *n* qubits 2^n base states
- 4158 qubits $2^{4158} \approx 10^{1251}$ base states (Number of particles in the observable Universe $\approx 10^{80}$)
- Base states of three qubits:
  - |000>, |001>, |010>, |011>, |100>, |101>, |110>, |111>
  - The system can be in any linear superposition of these states

Quantum state of three qubits

- $\alpha_{000}|000> + \alpha_{001}|001> + \dots\ \alpha_{111}|111> \Leftrightarrow \begin{bmatrix} \alpha_{000} \\ \alpha_{001} \\ \vdots \\ \alpha_{111} \end{bmatrix}$
- $|\alpha_{000}|^2 + |\alpha_{001}|^2 + \cdots + |\alpha_{111}|^2 = 1$

Complete measurement of multiple qubits

- $\alpha_{000}|000\rangle + \alpha_{001}|001\rangle + \ldots\ \alpha_{111}|111\rangle \Leftrightarrow \begin{bmatrix} \alpha_{000} \\ \alpha_{001} \\ \vdots \\ \alpha_{111} \end{bmatrix}$
- Probability of measuring the state $|000\rangle = |\alpha_{000}|^2$
- …

Partial measurement of multiple qubits

- E.g. one qubit
- Probability of measuring the state|0> of the first qubit is equal to the sum of the probabilities of measuring all base states where this qubit has the state |0>, i.e. |000>, |001>, |010>, |011>
  - $|\alpha_{000}|^2 + |\alpha_{001}|^2 + |\alpha_{010}|^2 + |\alpha_{011}|^2$

Quantum state of many qubits

- Tensor product
- $\begin{bmatrix}\alpha_0\\ \alpha_1\end{bmatrix} \otimes \begin{bmatrix}\beta_0\\ \beta_1\end{bmatrix} = \begin{bmatrix}\alpha_0 \begin{bmatrix}\beta_0\\ \beta_1\end{bmatrix}\\ \alpha_1 \begin{bmatrix}\beta_0\\ \beta_1\end{bmatrix}\end{bmatrix} = \begin{bmatrix}\alpha_0\beta_0\\ \alpha_0\beta_1\\ \alpha_1\beta_0\\ \alpha_1\beta_1\end{bmatrix}$

Entangled states

- States that cannot be decomposed (factored) into separate qubit states
- E.g. (one of Bell states):
  - $\begin{bmatrix} 1/\sqrt{2} \\ 0 \\ 0 \\ 1/\sqrt{2} \end{bmatrix} = 1/\sqrt{2}\,|00> + 1/\sqrt{2}\,|11>$

Evolution of quantum states

- Described by unitary matrices U
- $\Psi' = U\Psi$
- Unitary matrices
  - $UU^\dagger = U^\dagger U = I$
  - $U^\dagger$ - Hermitian conjugate – assembly of transposition and conjugate operations
  - Orthonormal
  - Reversible operations
- $U = e^{-i\hat{H}t/\hbar}$

NOT Gate

- $M\neg = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
- $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}^{\dagger} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
- Examples:
- $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$
- $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \sqrt{1/3} \\ \sqrt{2/3} \end{bmatrix} = \begin{bmatrix} \sqrt{2/3} \\ \sqrt{1/3} \end{bmatrix}$
- $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix} = \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix}$

Hadamard's gate

- $H = {}^{1}/_{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}$
- ${}^{1}/_{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}{}^{1}/_{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}^{\dagger} = {}^{1}/_{2}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix} = \begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}$
- Examples:
- ${}^{1}/_{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix}{}^{1}/_{\sqrt{2}}\\{}^{1}/_{\sqrt{2}}\end{bmatrix}$
- ${}^{1}/_{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}\begin{bmatrix}{}^{1}/_{\sqrt{2}}\\{}^{1}/_{\sqrt{2}}\end{bmatrix} = \begin{bmatrix}1\\0\end{bmatrix}$

CNOT Gate (Controlled NOT)

- $$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$
- Example:
- $$\begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1/\sqrt{2} \\ 0 \\ 1/\sqrt{2} \\ 0 \end{bmatrix}$$
- $$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} \\ 0 \\ 1/\sqrt{2} \\ 0 \end{bmatrix} = \begin{bmatrix} 1/\sqrt{2} \\ 0 \\ 0 \\ 1/\sqrt{2} \end{bmatrix}$$ - entangled Bell's state

Composed transformations (gates) running on different qubits

- $U = U_1 \otimes U_2$

Universality of quantum gates

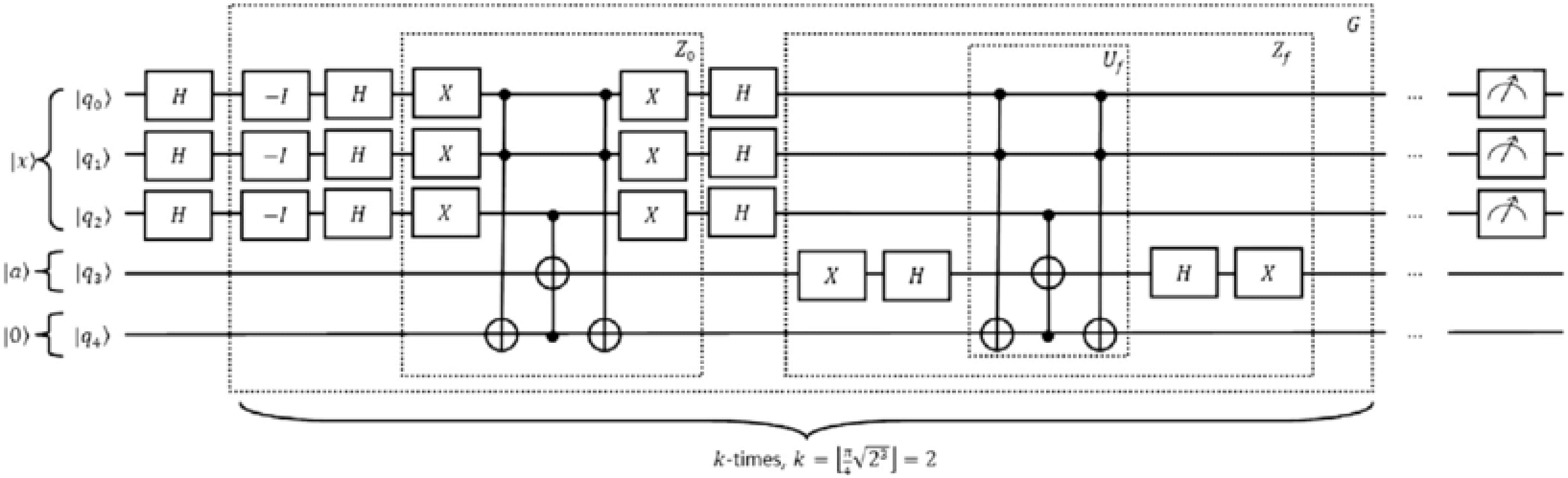
- There exist sets of (e.g. four) quantum gates with which any unitary transformation can be approximated
- Gates acting on one and two qubits are sufficient
- Of course, there is the question of efficiency (e.g. the number of gates used)

Timothée Goubault de Brugière, Marc Baboulin, Benoît Valiron, Cyril Allouche, Quantum circuits synthesis using Householder transformations, Computer Physics Communications, Volume 248, 2020.

Quantum algorithm tools

- Setting initial qubit values
- Unitary transformations
- Complete or partial measurement
- Classic algorithms

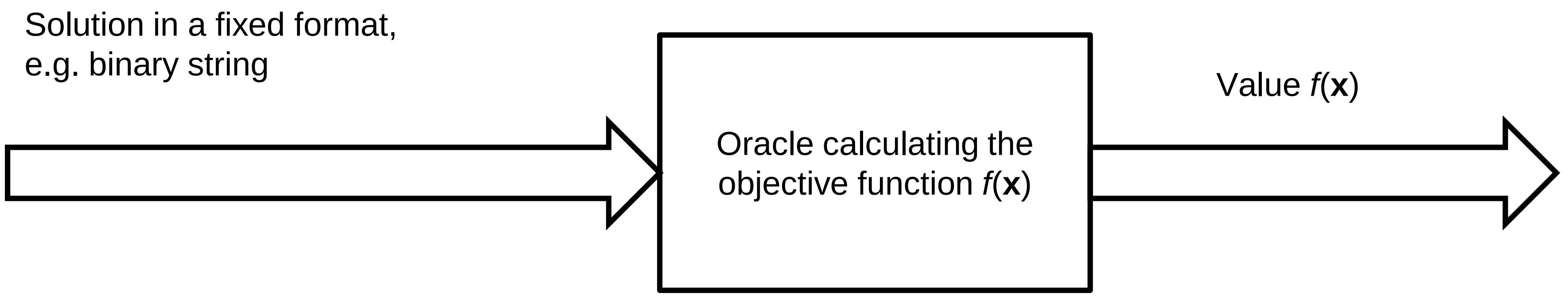
Circuit model - Quantum gate array



Black-box search

- $f: \{0,1\}^n \longrightarrow \{0,1\}$
- $f(\boldsymbol{x}) = 1$ only for one $\boldsymbol{x}^*$
- Complexity on classic computers $O(N), N = 2^n$ (No free lunch theorem)

Classic oracle model, black box



Classic oracle – a function with an unknown implementation

```
double function (Vector<boolean>) {
    // Unknown implementation
    ...
    return...
}
```

Grover's quantum algorithm for black-box search

- Complexity:
- $O\left(\sqrt{N}\right)$ Iterations $\left(\Omega\left(\sqrt{N}\right)\right)$?
- $O\left(\sqrt{N}\log N\right)$ gates

Grover's algorithm

- Each base state n qubits encode one of the $N = 2^n$ solutions

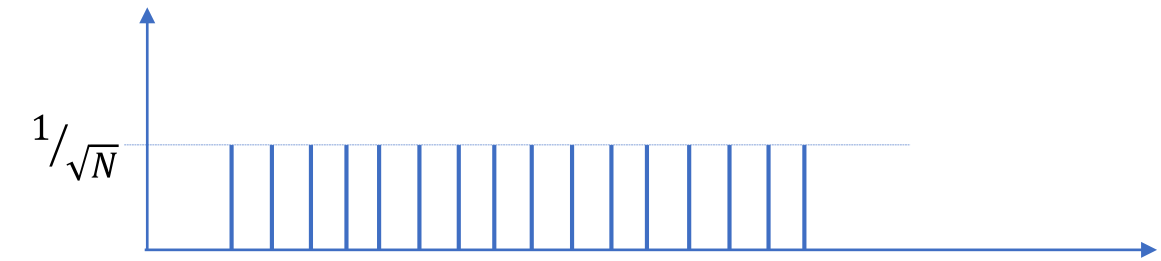
1. Set the initial state in uniform superposition $|s> = \begin{bmatrix} 1/\sqrt{N} \\ 1/\sqrt{N} \\ \vdots \\ 1/\sqrt{N} \end{bmatrix}$
2. **Repeat** $O(\sqrt{N})$ times:
   1. $|s> = U_{\omega}|s>$
   2. $|s> = (2P - I)|s>$
3. Measure all qubits (with high probability the state corresponding to the correct solution is obtained)
4. Check the solution, if it is not correct, repeat the algorithm

U_ω - oracle

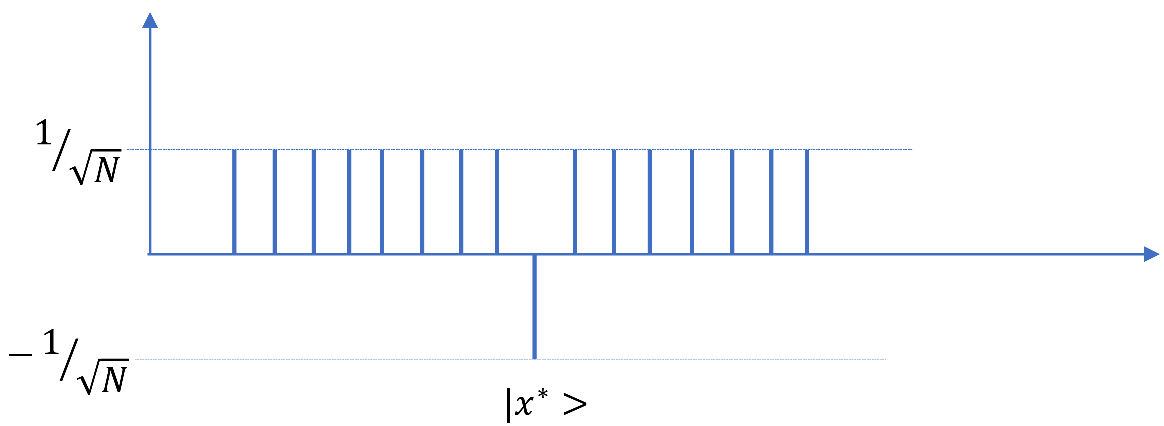
- $U_\omega : \begin{cases} |x> \mapsto -|x>, \ f(x) = 1 \\ |x> \mapsto |x>, f(x) = 0 \end{cases}$
- $U_\omega = \begin{bmatrix} 1 & & & & \\ & 1 & & & \\ & & -1 & & \\ & & & \ddots & \\ & & & & 1 \end{bmatrix}$

Effect of U_ω

- $|s>$



- $U_\omega|s>$



Transformation $2P - I$

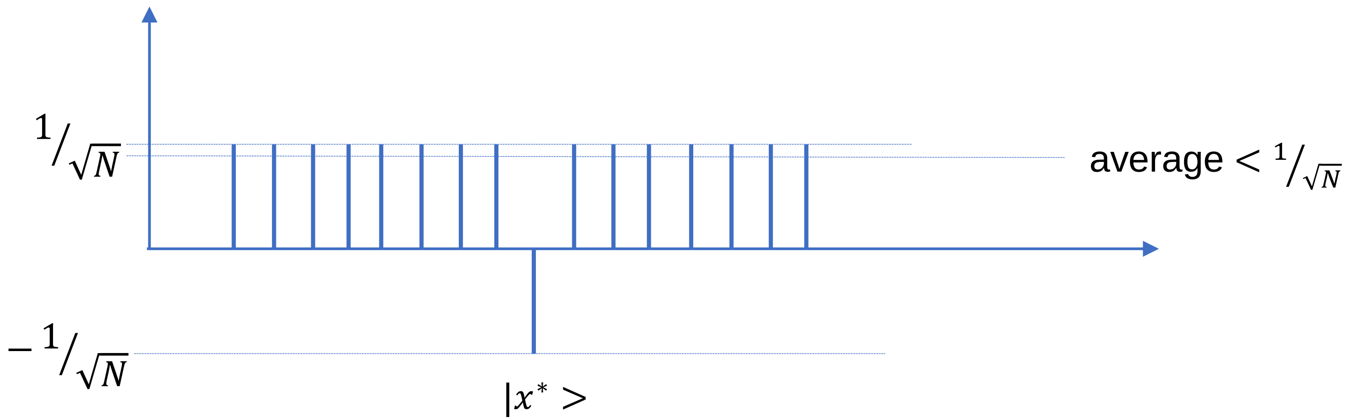
- $P = \begin{bmatrix} 1/N & 1/N & 1/N & \ddots & 1/N \\ 1/N & 1/N & 1/N & \ddots & 1/N \\ 1/N & 1/N & 1/N & \ddots & 1/N \\ \ddots & \ddots & \ddots & \ddots & \ddots \\ 1/N & 1/N & 1/N & \ddots & 1/N \end{bmatrix}$
- $2P - I = \begin{bmatrix} 2/N - 1 & 2/N & 2/N & \ddots & 2/N \\ 2/N & 2/N - 1 & 2/N & \ddots & 2/N \\ 2/N & 2/N & 2/N - 1 & \ddots & 2/N \\ \ddots & \ddots & \ddots & \ddots & \ddots \\ 2/N & 2/N & 2/N & \ddots & 2/N - 1 \end{bmatrix}$

Effect of $2P - I$ – reflection around the average

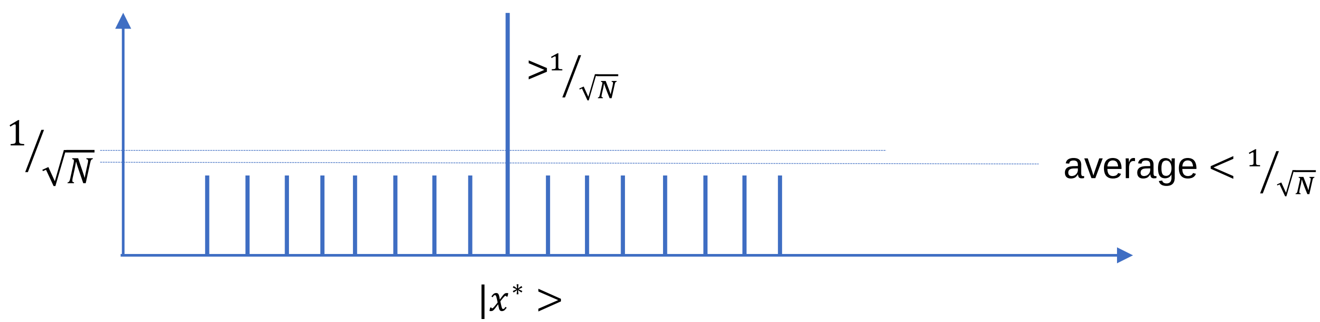
- $$2P\begin{bmatrix}\alpha_1\\ \alpha_2\\ \vdots\\ \alpha_N\end{bmatrix} = \begin{bmatrix}{}^2/_N\sum_i \alpha_i\\ {}^2/_N\sum_i \alpha_i\\ \vdots\\ {}^2/_N\sum_i \alpha_i\end{bmatrix} = \begin{bmatrix}2\overline{\alpha}_i\\ 2\overline{\alpha}_i\\ \vdots\\ 2\overline{\alpha}_i\end{bmatrix}$$
- $$(2P - I)\begin{bmatrix}\alpha_1\\ \alpha_2\\ \vdots\\ \alpha_N\end{bmatrix} = \begin{bmatrix}2\overline{\alpha}_i - \alpha_1\\ 2\overline{\alpha}_i - \alpha_2\\ \vdots\\ 2\overline{\alpha}_i - \alpha_N\end{bmatrix} = \begin{bmatrix}\overline{\alpha}_i + (\overline{\alpha}_i - \alpha_1)\\ \overline{\alpha}_i + (\overline{\alpha}_i - \alpha_2)\\ \vdots\\ \overline{\alpha}_i + (\overline{\alpha}_i - \alpha_N)\end{bmatrix}$$

Effect of $2P - I$ – reflection around the average

- $U_\omega|s>$



- $(2P - I)U_\omega|s>$



$$2P - I = H^{\otimes n} R H^{\otimes n}$$

- $H^{\otimes n} = H \otimes H \otimes \cdots \otimes H$
- $H^{\otimes 3} = {}^{1}\!/_{\sqrt{8}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} =$
- $= {}^{1}\!/_{\sqrt{8}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} =$
- $= {}^{1}\!/_{\sqrt{8}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix}$

Transformation R

- $R: \begin{cases} |x>\mapsto |x>, |x> = |00\ldots0> \\ |x>\mapsto -|x>,\ x \neq 0 \end{cases}$
- $R = \begin{bmatrix} 1 & & & & \\ & -1 & & & \\ & & -1 & & \\ & & & \ddots & \\ & & & & -1 \end{bmatrix}$

Remarks on Grover's algorithm

- Has a probabilistic character – it gives the desired result with a high probability (the correctness of the result should be checked and repeated if necessary)
- May be adapted to situations where there are many solutions
- It is the best possible quantum algorithm for this problem
- It has a number of extensions/generalizations, e.g. amplitude amplification
- May be adapted to optimize black-box $O\left(\sqrt{N}\right)$
    - Select any x
    - **Repeat**
        - Find y: $f(y) < f(x)$ using the Grover's algorithm
        - If found, then $x = y$, otherwise x is a minimum

Shor's factorization algorithm

- Quantum part – the use quantum Fourier transformation to find the period of a function

Quantum algorithms building blocks/patterns/primitives

Data Encoding	Quantum processing	Post-processing and readout
Computational Basis Encoding Amplitude Encoding Angle Encoding Qsample Encoding Multiple Qubit Registers QRAM QROM Matrix Encoding (input)	Quantum Amplitude Amplification Quantum Amplitude Estimation Quantum Fourier Transform Quantum Phase Estimation Applying a Hermitian Operator Linear Combination of Unitaries Quantum (random) walks Hamiltonian simulation Postselection Quantum Signal Processing Quantum Singular Value Transformations	Basic states measurement Probability estimation Quantum tomography Shadow tomography Swap Test and Sample Mean Estimation Postselected learning of quantum states.

How to use quantum computing in your research

1. An attempt to construct a new algorithm using available tools:
   - Setting initial qubit values
   - Unitary transformations
   - Complete or partial measurement
   - Classic algorithms
   - Building blocks/patterns/primitives

   How?
     - Manually
     - Genetic programming
2. Application of one of the existing quantum algorithms
3. "Attacking" one of the subproblems
   - E.g. synthesis of quantum circuits, quantum tomography, Postselected learning of quantum states, …
4. …

ELSEVIER

Annual Reviews in Control
Available online 25 May 2022
In Press, Corrected Proof

Annual Reviews in Control

Vision article

Quantum estimation, control and learning: Opportunities and challenges ☆

Daoyi Dong [a], Ian R. Petersen [b]

Show more

Add to Mendeley Share Cite

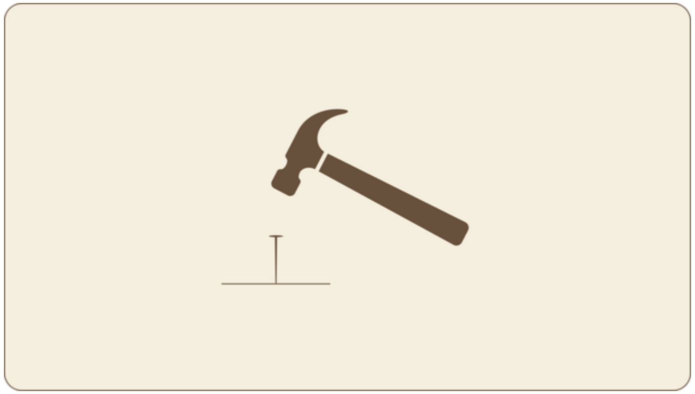
https://doi.org/10.1016/j.arcontrol.2022.04.011

Abstract

The development of estimation and control theories for quantum systems is a fundamental task for practical quantum technology. This vision article presents a brief introduction to challenging problems and potential opportunities in the emerging areas of quantum estimation, control and learning. The topics cover quantum state estimation, quantum parameter identification, quantum filtering, quantum open-loop control, quantum feedback control, machine learning for estimation and control of quantum systems, and quantum machine learning.

Quantum Algorithm Zoo

- https://quantumalgorithmzoo.org/
- Factoring
- Quantum Fourier transform
- Discrete-log
- Gauss Sums
- Primality Proving
- Solving Exponential Congruences
- Matrix Elements of Group Representations



Quantum Algorithm Zoo

- Verifying Matrix Products
- Subset-sum
- Decoding
- Formula Evaluation
- Polynomial interpolation
- Pattern matching
- Ordered Search
- Graph Properties in the Adjacency Matrix Model

Quantum Algorithm Zoo

- Graph Properties in the Adjacency List Model
- Welded Tree
- Collision Finding and Element Distinctness
- Graph Collision
- Matrix Commutativity
- Center of Radial Function
- Group Order and Membership
- Group Isomorphism

Quantum Algorithm Zoo

- Statistical Difference
- Matrix Rank
- Matrix Multiplication over Semirings
- Subset finding
- Search with Wildcards
- Network flows
- Electrical Resistance
- Junta Testing and Group Testing

Quantum Algorithm Zoo

- Quantum Approximate Optimization
- Semidefinite Programming
- Zeta Functions
- Weight Enumerators
- Simulated Annealing
- String Rewriting
- Matrix Powers
- Constraint Satisfaction

Quantum Algorithm Zoo

- Gradients, Structured Search, and Learning Polynomials
- Linear Systems
- Machine Learning
- Tensor Principal Component Analysis
- Solving Differential Equations
- Quantum Dynamic Programming

Other issues in quantum computing

- Adiabatic Quantum Computation
- Synthesis of quantum circuits
- Relationship with reversible calculations
- Analysis of the complexity of quantum algorithms
- Quantum correction codes
- Quantum-inspired classic algorithms
- Quantum telecommunications, teleportation
- Quantum and post-quantum cryptography
- Building quantum computers
- Libraries, programming languages
- …