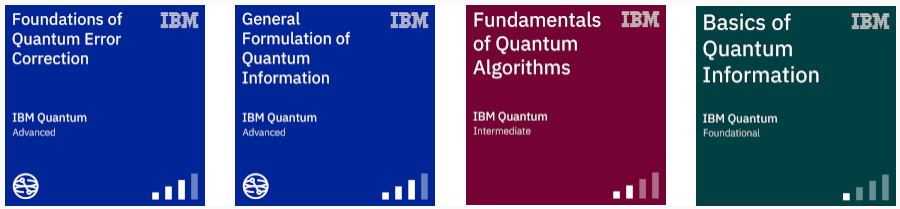
Quantum Computation and Information
The Algorithms

Dr. Thorsten Hehn



Outline

Reminder:
Bra-ket Notation by Dirac

Quantum Algorithms

Quantum Advantage and Q-Day

Outlook

Reminder:
Bra-ket Notation by Dirac

Reminder: Bra-ket Notation by Dirac

Basis states:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\langle 0| = |0\rangle^H = \begin{pmatrix} 1 & 0 \end{pmatrix}, \quad \langle 1| = |1\rangle^H = \begin{pmatrix} 0 & 1 \end{pmatrix}$$

Well-known vector-calculus rules apply:

$$\begin{aligned} \langle 0|\,|0\rangle &= \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = 1 \\ |0\rangle\,\langle 0| &= \begin{pmatrix} 1 \\ 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \end{aligned}$$

Quantum Algorithms

Quantum Algorithms - An overview

1985–1993	**Foundations** Deutsch Algorithm, Deutsch–Jozsa Algorithm, Bernstein–Vazirani Problem, Simon's Algorithm
1994–1996	**Breakthrough** Shor, Grover
1998–2010	**Expansion** Amplitude Amplification HHL (Harrow-Hassidim-Lloyd) to solve linear equations
2014–today	**Noisy Intermediate Scale Quantum (NISQ) Era** (50 - 1000 qubits, still noisy) Variational Quantum Eigensolver (VQE), Quantum Machine Learning, Error Mitigation

Deutsch's Problem

- Published by David Deutsch in 1985
- Tailored to demonstrate an advantage of Quantum Computers

Problem statement: four possible functions $f(x), \{0,1\} \rightarrow \{0,1\}$

x	$f_1(x)$
0	0
1	0
Constant	

x	$f_2(x)$
0	1
1	1
Constant	

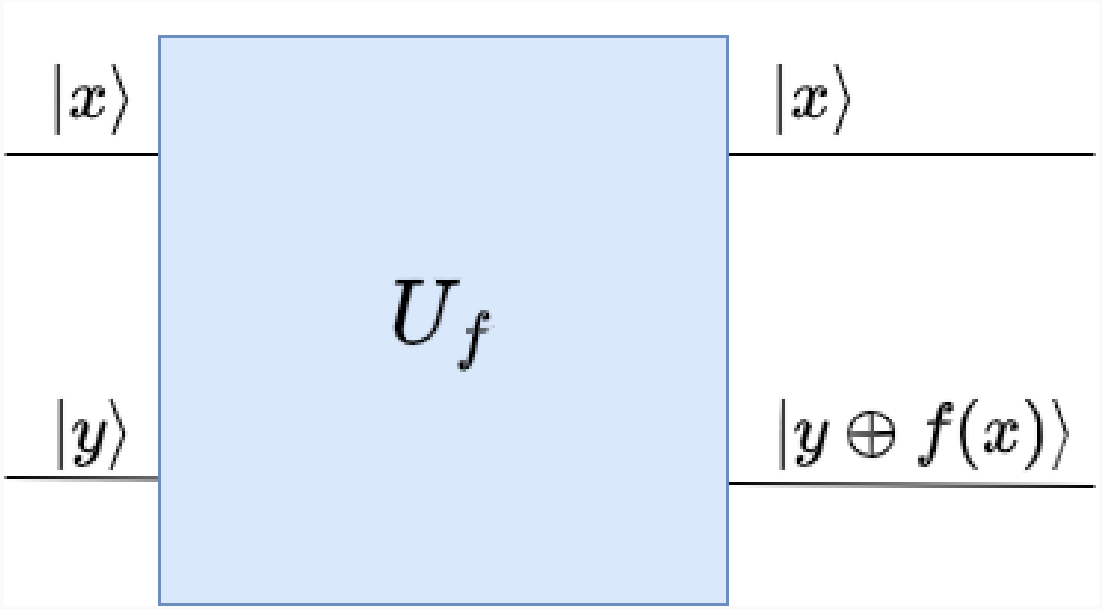
x	$f_3(x)$
0	0
1	1
Balanced	

x	$f_4(x)$
0	1
1	0
Balanced	

- Classify unknown $f(x)$ as constant or balanced
- Classical computing: Always two function queries necessary

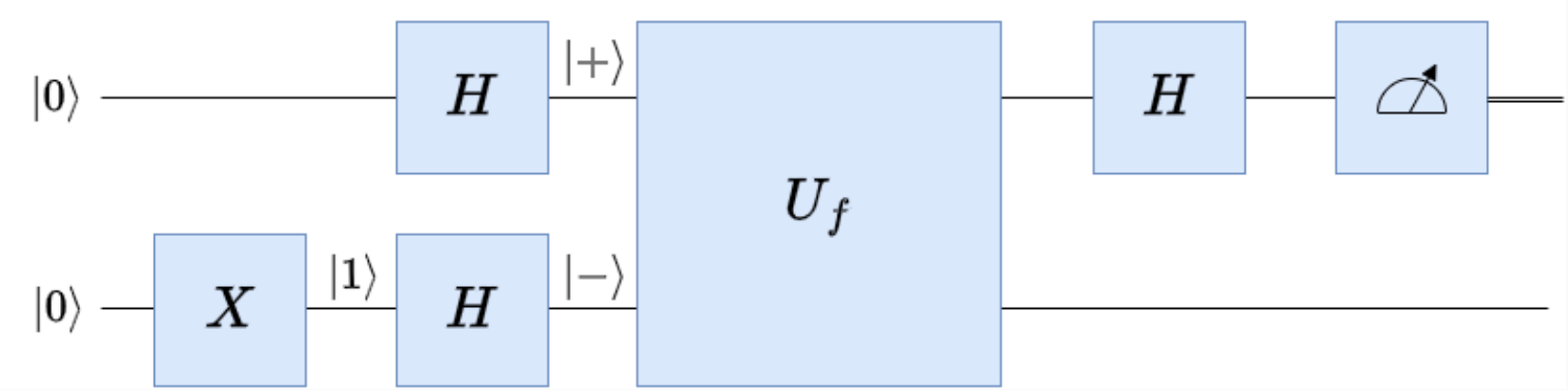
Quantum Query Model

Query model as unitary function ($U_f U_f^H = I$)



Deutsch's Algorithm

Remarkable result: Deutsch's algorithm requires only one function query. The key method is superposition.

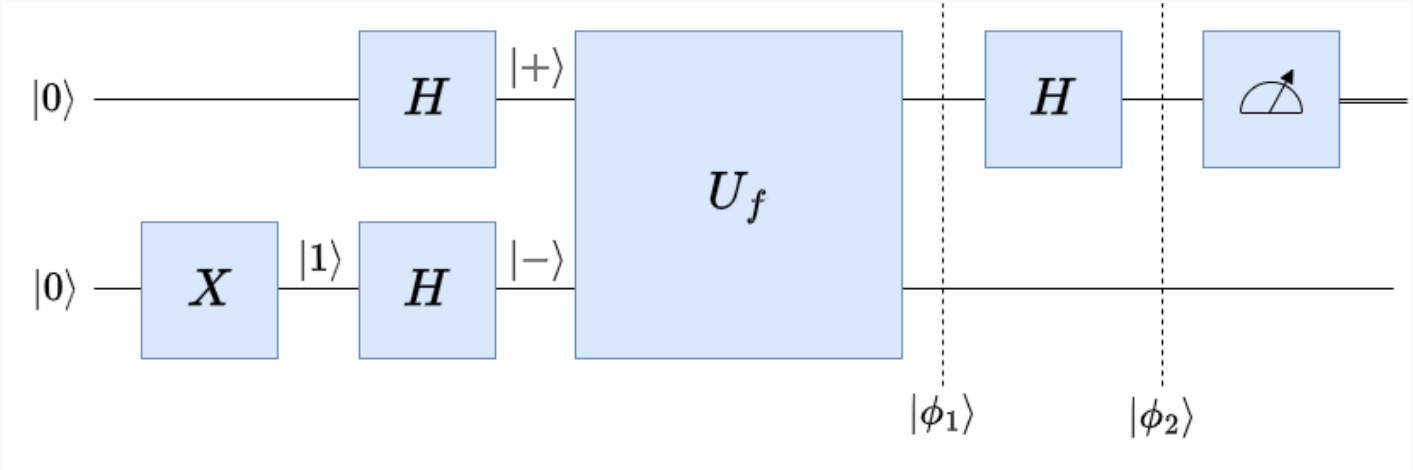


X: Not gate

H: Hadamard gate

$|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \quad |-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$

Deutsch's Algorithm - Analysis



Qiskit notation: left-to-right means bottom-to-top.

Input:

$$|-\rangle|+\rangle = \frac{1}{2}(|0\rangle - |1\rangle)(|0\rangle + |1\rangle) = \frac{1}{2}(|0\rangle - |1\rangle)|0\rangle + \frac{1}{2}(|0\rangle - |1\rangle)|1\rangle$$

Output:

$$|\phi_1\rangle = \frac{1}{2}(|0 \oplus f(0)\rangle - |1 \oplus f(0)\rangle)|0\rangle + \frac{1}{2}(|0 \oplus f(1)\rangle - |1 \oplus f(1)\rangle)|1\rangle$$

Deutsch's Algorithm - Analysis

$$
\begin{aligned}
|\phi_1\rangle &= \frac{1}{2}\left(|0 \oplus f(0)\rangle - |1 \oplus f(0)\rangle\right)|0\rangle + \frac{1}{2}\left(|0 \oplus f(1)\rangle - |1 \oplus f(1)\rangle\right)|1\rangle \\
&= \frac{1}{2}(-1)^{f(0)}\left(|0\rangle - |1\rangle\right)|0\rangle + \frac{1}{2}(-1)^{f(1)}\left(|0\rangle - |1\rangle\right)|1\rangle \\
&= |-\rangle\left(\frac{(-1)^{f(0)}|0\rangle + (-1)^{f(1)}|1\rangle}{\sqrt{2}}\right) \\
&= (-1)^{f(0)}|-\rangle\left(\frac{|0\rangle + (-1)^{f(0)\oplus f(1)}|1\rangle}{\sqrt{2}}\right) \\
&= \begin{cases} (-1)^{f(0)}|-\rangle|+\rangle & \text{if } f(0) = f(1) \\ (-1)^{f(0)}|-\rangle|-\rangle & \text{if } f(0) \neq f(1) \end{cases} \\
|\phi_2\rangle &= \begin{cases} (-1)^{f(0)}|-\rangle|0\rangle & \text{if } f(0) = f(1) \\ (-1)^{f(0)}|-\rangle|1\rangle & \text{if } f(0) \neq f(1) \end{cases}
\end{aligned}
$$

Deutsch's Algorithm - Interpretation

- One function query suffices to classify the function
- Measurement of the upper qubit yields the result
- Note that the upper qubit is a pass-through of the input. The phenomenon of information being encoded in its phase (before the final Hadamard gate) is called phase kickback
- Deutsch's Algorithm has no practical application, but inspired many algorithms with such application

Integer Factorization

State of play:

- RSA-250d (RSA-829) challenge factored in 2020
  (250 digit number represented by 829 bits)
- Thousands of years of CPU-time

RSA-829

$$N = 214032465024074496126442307283933356300861471514475501779775492088141802344714013664334551909580467961099285187247091458768739626192155736304745477052080511905649310668769159001975940569345745223058932597669747168173806936489469987157849497593749793 7$$

$$= p \cdot q$$

$$p = 641352894770715802787901901705773890848250147429434472081168596320245323446302386235987526683477087376619255856946397988533 67$$

$$q = 333720275949781565562260106053551142279407603447675546667845209870238417292100370802574486732968818775657189862580369320627 11$$

Integer Factorization

State of play:

- RSA-1024, RSA-2048, etc currently not factored

RSA-2048

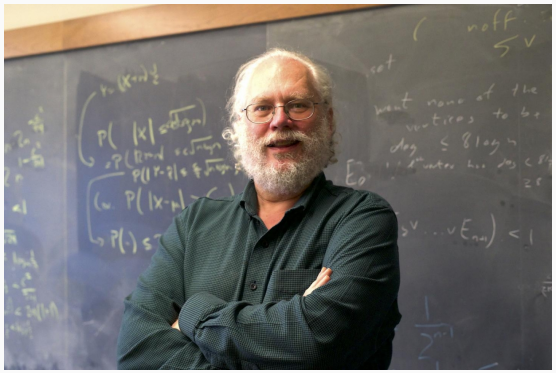
$$
\begin{aligned}
N = {} & 25195908475657893494027183240048398571429282126204032027777137836043662020 \\
& 75955562640185258807844069182906412495150821892985591491761845028084891200 \\
& 72844992687392807287776735971418347270261896375014971824691165077613379859 \\
& 09570009733045974880842840179742910064245869181719511874612151517265463228 \\
& 22168699875491824224336372590851418654620435767984233871847744479207399342 \\
& 36584823824281198163815010674810451660377306056201619676256133844143603833 \\
& 90441495263443219011465754445417842402092461651572335077870774981712577246 \\
& 79629263863563732899121548314381678998850404453640235273819513786365643912 \\
& 120103971228221207203 57 \\
= {} & p \cdot q
\end{aligned}
$$

$p = ?$

$q = ?$

Shor's Algorithm

- Published by Peter Shor in 1994
- Novel method for factorization of large integers
- Famous for breaking public-key cryptography once a capable quantum computer exists
- Demonstrating exponential quantum speedup



Peter Shor, Image source: CERN

It is possible that a classical algorithm with similar properties exists. But none has been found so far.

Shor's Algorithm

Main idea for factoring N:

- Probabilistic algorithm, succeeds with at least probability $1/2$ when N is odd and not a prime power.
- Ingredients: Greatest common divisor (gcd), Order finding
- GCD: Efficient algorithms exist (Euclid's)
- Order finding is exponentially hard on classical computers

Shor's Algorithm

$$\mathbb{Z}_N^* = \{a \in \mathbb{Z}_N : gcd(a, N) = 1\}$$

For every $a \in \mathbb{Z}_N^*$ there must exist a positive integer k such that $a^k = 1 \,(\mathrm{mod} N)$. The smallest k fulfilling this property is called the order of a in $\mathbb{Z}_N^*$.

Shor's Algorithm solves the order finding problem efficiently:

- Quantum Phase Estimation (Cost $O(n^3)$, n: encoding length)
- Quantum Fourier Transform (Cost $O(n^2)$)
- Overall cost is polynomial in encoding length n

Quantum Advantage and Q-Day

Quantum Advantage

Quantum advantage refers to performing an information processing task more efficiently, more cost-effectively, or more accurately using a quantum computer than is known to be possible with classical computers alone.

- The comparison is against the *best known classical methods*.
- Advantage may refer to *runtime*, *cost*, or *accuracy*.
- It may be *theoretical* (predicted by algorithms and complexity arguments) or *experimental* (demonstrated on hardware).
- **Practical quantum advantage** for broadly useful applications is still an active research goal.

Q-Day

Q-Day marks the "advent" of a

Cryptographically-Relevant Quantum Computer (CRQC).

Cryptographic relevance is given when a standard cryptographic method, such as RSA-2048 or ECC-256, can be broken within a few days. These requires multiple major advances. Q-Day predictions taken from `postquantum.com`

Requirements: Logical Qubit Capacity (LQC) and number of physical qubits: based on attack vector and error correction scheme

- LQC requirement Shor paper: $3\log_2(N) = 6144$ for $N = 2048$
- LQC requirement of refined quantum attacks (e.g. Gidney 2025): ≈ 1400 qubits
- Physical qubit requirement: $\approx 10^6$ qubits
- Note: stark contrast to current implementation level (≈ 150

Q-Day

Q-Day marks the "advent" of a

Cryptographically-Relevant Quantum Computer (CRQC).

Cryptographic relevance is given when a standard cryptographic method, such as RSA-2048 or ECC-256, can be broken within a few days. These requires multiple major advances. Q-Day predictions taken from `postquantum.com`

Capabilities: Base Technology to implement qubits

- Trapped-Ion Qubits: $\rightarrow$ 2034
- Superconducting Qubits: $\rightarrow$ 2044

Q-Day

Q-Day marks the "advent" of a

Cryptographically-Relevant Quantum Computer (CRQC).

Cryptographic relevance is given when a standard cryptographic method, such as RSA-2048 or ECC-256, can be broken within a few days. These requires multiple major advances. Q-Day predictions taken from `postquantum.com`

Relation to Quantum Advantage:

Outlook

Outlook

- Today's Algorithms
  (Noisy Intermediate-Scale Quantum, NISQ)
  - Variational Quantum Eigensolver (VQE): Chemistry and Life Sciences
  - Quantum Annealing: Logistics, Routing, Network Optimization
  - Quantum Approximate Optimization Algorithm (QAOA): Combinatorial problems
- Future
  (Fault-Tolerant Quantum Computing, FTQC)
  - Grover's: unstructured search, databases
  - Shor's: revolutionize public-key cryptography
  - Hamiltonian Simulation
- Longer-Term Research
  - Solving Linear Systems
  - Future Applications e.g. molecular simulations