Lecture 3: AI and Quantum Physics and Quantum Computation

Andreas Wichert

Department of Computer Science and Engineering

Técnico Lisboa

Overview

- Unstructured Search
- Quantum Oracle
- Householder Reflection
- Grover’s Amplification
- Iterative Amplification
- Number of Iterations
- Example
- SAT Problem

Lov Kumar Grover (born 1961) is an Indian-American computer scientist.
Lov Grover received his BTech in Electrical Engineering from IIT Delhi in 1981
He is the originator of the Grover database search algorithm used in quantum computing.

To be a **Genius** You do not need a high google score!

Google Scholar SIGN IN

Lov Grover

Unknown affiliation
No verified email

FOLLOW

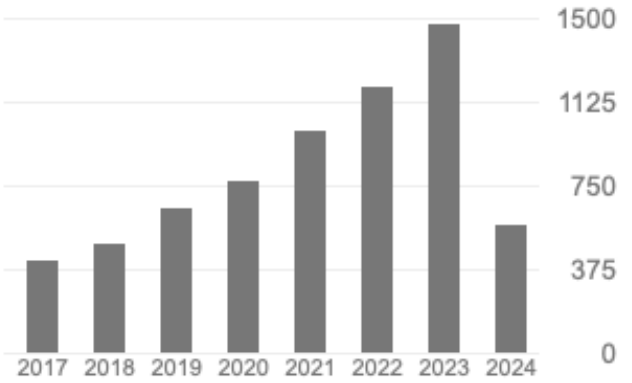
GET MY OWN PROFILE

TITLE	CITED BY	YEAR
A fast quantum mechanical algorithm for database search LK Grover Proceedings of the twenty-eighth annual ACM symposium on Theory of computing …	10588	1996

SHOW MORE

Cited by	All	Since 2019
Citations	10588	5671
h-index	1	1
i10-index	1	1

VIEW ALL



Unstructured Search

- For a function $o_\xi(x)$

$$o_\xi(x) = \begin{cases} 1 & if \quad x = \xi \\ 0 & else \end{cases}$$

- We want to find x for which $o_\xi(x) = 1$ *if* $x = \xi$. The task is equivalent to a decision problem with a binary answer ***1 = yes*** and ***0 = no*** and the instance x.
- Grover's amplification algorithm implements exhaustive search in $O(\sqrt{n})$ steps in n-dimensional Hilbert space
- It follows that using a quantum computer NP − complete problems remain NP − complete

- The search for ξ is based on the three principles of quantum computation:
  - The function $o(x)$ is represented by a quantum Boolean circuit T
  - T can be represented by a unitary operator (matrix)
  - For the function $o(x)$, the solution is encoded by $\boldsymbol{(-1)^{o(x)}}$, the **sign** of the amplitude
- If $o(x)$ is NP – complete, a quantum circuit T with a polynomial number of quantum gates T verifies for a given instance

  x_{ticket} if $o(x_{ticket}) = 1$, $x_{ticket} = \xi$

  or $o(x_{ticket}) = 0$

Solution is encoded by $(-1)^{o(x)}$, the **sign** of the amplitude

- The unitary operator *T* represents the **quantum oracle** function *o(x)* that determines if the configuration is the goal configuration
- The auxiliary qubit *c* is set to one, and the state is represented by m qubits $|0^{\otimes m}\rangle$.
  - First, we set qubits representing the states and the auxiliary qubit in superposition by the Hadamard gate for m + 1 qubits H_{m+1}, and then we execute the unitary operator *T*

$$T \cdot H_{m+1} \cdot |0^{\otimes m}\rangle|1\rangle =$$

$$= \frac{1}{\sqrt{2^{m+1}}} \cdot \sum_{x \in B^m} T \cdot |x\rangle|0\rangle - \frac{1}{\sqrt{2^{m+1}}} \cdot \sum_{x \in B^m} T \cdot |x\rangle|1\rangle$$

$$= \frac{1}{\sqrt{2^{m+1}}} \cdot \sum_{x \in B^m} |x\rangle|o(x) \oplus 0\rangle - \frac{1}{\sqrt{2^{m+1}}} \cdot \sum_{x \in B^m} |x\rangle|o(x) \oplus 1\rangle$$

$$= \frac{1}{\sqrt{2^{m+1}}} \cdot \left(\sum_{x \in B^m} |x\rangle|o(x) \oplus 0\rangle - \sum_{x \in B^m} |x\rangle|o(x) \oplus 1\rangle \right).$$

$$T \cdot H_{m+1} \cdot |0^{\otimes m}\rangle|1\rangle = \frac{1}{\sqrt{2^{m+1}}} \cdot \left(\sum_{x \in B^m} |x\rangle|o(x) \oplus 0\rangle - \sum_{x \in B^m} |x\rangle|o(x) \oplus 1\rangle \right).$$

There are four possible cases with the state $|\xi\rangle$ being the solution:

$$T \cdot |x\rangle|0\rangle = |x\rangle|o(x) \oplus 0\rangle = |x\rangle|0\rangle,$$

$$T \cdot |x\rangle|1\rangle = |x\rangle|o(x) \oplus 1\rangle = |x\rangle|1\rangle,$$

$$T \cdot |\xi\rangle|0\rangle = |\xi\rangle|f(\xi) \oplus 0\rangle = |\xi\rangle|1\rangle,$$

$$T \cdot |\xi\rangle|1\rangle = |\xi\rangle|f(\xi) \oplus 1\rangle = |\xi\rangle|0\rangle.$$

$$
\begin{aligned}
T \cdot H_{m+1} \cdot |0^{\otimes m}\rangle|1\rangle &= \frac{1}{\sqrt{2^{m+1}}} \cdot \left(\sum_{x \neq \xi} |x\rangle|0\rangle + |\xi\rangle|1\rangle - \sum_{x \neq \xi} |x\rangle|1\rangle - |\xi\rangle|0 \right. \\
&\quad \frac{1}{\sqrt{2^{m+1}}} \cdot \left(\sum_{x \neq \xi} |x\rangle \left(|0\rangle - |1\rangle\right) + |\xi\rangle \left(|1\rangle - |0\rangle\right) \right) \\
&\quad = \frac{1}{\sqrt{m}} \sum_{x \in B^m} (-1)^{o(x)} \cdot |x\rangle \otimes \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right).
\end{aligned}
$$

The value of the function $o(x)$ is encoded by $(-1)^{o(x)}$, the operation is a phase kick-back. We can set the auxiliary qubit $c = \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right)$ to zero by the Hadamard gate.

Housholder Reflection

- The Householder reflection reflects one vector |x⟩ to its negative and leaves invariant the orthogonal complement of this vectors.
- It is described by the Householder matrix Q_x with ‖|x⟩‖ = 1 representing m qubits with $n = 2^m$ and the projection matrix

$$P = |x\rangle\langle x|$$

$$Q_x = I_m - 2 \cdot P$$

- Suppose P_m is generated by the normalized vector |x⟩ indicating the direction of the bisecting line,

$$|x\rangle = \frac{1}{\sqrt{n}} \cdot |x_1\rangle + \frac{1}{\sqrt{n}} \cdot |x_2\rangle + \cdots + \frac{1}{\sqrt{n}} \cdot |x_n\rangle = \begin{pmatrix} \frac{1}{\sqrt{n}} \\ \vdots \\ \frac{1}{\sqrt{n}} \end{pmatrix}$$

then the projection matrix P_m is

$$P_m = |x\rangle\langle x| = \begin{pmatrix} \frac{1}{n} & \frac{1}{n} & \cdots & \frac{1}{n} \\ \frac{1}{n} & \frac{1}{n} & \cdots & \frac{1}{n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{n} & \frac{1}{n} & \cdots & \frac{1}{n} \end{pmatrix}$$

Householder Reflection

$$P_m = |x\rangle\langle x| = \begin{pmatrix} \frac{1}{n} & \frac{1}{n} & \cdots & \frac{1}{n} \\ \frac{1}{n} & \frac{1}{n} & \cdots & \frac{1}{n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{n} & \frac{1}{n} & \cdots & \frac{1}{n} \end{pmatrix}$$

The projection matrix P computes for each dimension the mean value

$$\begin{pmatrix} \frac{\sum_{i=1}^{n} x_i}{n} \\ \frac{\sum_{i=1}^{n} x_i}{n} \\ \vdots \\ \frac{\sum_{i=1}^{n} x_i}{n} \end{pmatrix} = \begin{pmatrix} \frac{1}{n} & \frac{1}{n} & \cdots & \frac{1}{n} \\ \frac{1}{n} & \frac{1}{n} & \cdots & \frac{1}{n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{n} & \frac{1}{n} & \cdots & \frac{1}{n} \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

and the Householder reflection

$$Q_x = I_m - 2 \cdot P_m.$$

computes the following mapping,

$$x_i^{new} = x_i^{old} - 2 \cdot \frac{\sum_{i=1}^{n} x_i^{old}}{n}.$$

Q_x is unitary,

$$Q_x \cdot Q_x^* = (I_m - 2 \cdot |x\rangle\langle x|) \cdot (I_m - 2 \cdot |x\rangle\langle x|)^*$$

$$Q_x \cdot Q_x^* = I_m - 2 \cdot |x\rangle\langle x| - 2 \cdot |x\rangle\langle x| + 4 \cdot |x\rangle\langle x| \cdot |x\rangle\langle x|$$

$$Q_x \cdot Q_x^* = I_m - 4 \cdot |x\rangle\langle x| + 4 \cdot \langle x|x\rangle \cdot |x\rangle\langle x| = I_m$$

$\langle x|x\rangle = 1$ because $\||x\rangle\| = 1$. With

$$P = |x\rangle\langle x|.$$

Grover's Amplification

Grover's amplification is based on $-Q_x$. It is a unitary operator with

$$G_m := -Q_x = -I_m + 2 \cdot P_m = 2 \cdot P_m - I_m$$

the mapping is defined as,

$$x_i^{new} = 2 \cdot \frac{\sum_{i=1}^{n} x_i^{old}}{n} - x_i^{old}.$$

Suppose only one amplitude of x_j is negative and the other one are positive. Then the corresponding amplitude grows with

$$x_j^{new} = 2 \cdot \frac{\sum_{i=1}^{n} x_j^{old}}{n} + x_i^{old}$$

$$\begin{pmatrix} 2 \cdot \frac{\sum_{i=1}^{n} x_i}{n} - x_1 \\ 2 \cdot \frac{\sum_{i=1}^{n} x_i}{n} + x_2 \\ \vdots \\ 2 \cdot \frac{\sum_{i=1}^{n} x_i}{n} - x_n \end{pmatrix} = \begin{pmatrix} \frac{2}{n} - 1 & \frac{2}{n} & \cdots & \frac{2}{n} \\ \frac{2}{n} & \frac{2}{n} - 1 & \cdots & \frac{2}{n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{2}{n} & \frac{2}{n} & \cdots & \frac{2}{n} - 1 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ -x_2 \\ \vdots \\ x_n \end{pmatrix}$$

The probability of measuring the solution depending on the size n is

$$p(solution) = \left| \frac{3}{\sqrt{n}} - \frac{4}{n \cdot \sqrt{n}} \right|^2$$

and non solution

$$p(non\ solution) = \left| \frac{1}{\sqrt{n}} - \frac{4}{n \cdot \sqrt{n}} \right|^2 .$$

Why?

Projection computes the mean value

$$P_m = |x\rangle\langle x| = \begin{pmatrix} \frac{1}{n} & \frac{1}{n} & \dots & \frac{1}{n} \\ \frac{1}{n} & \frac{1}{n} & \dots & \frac{1}{n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{n} & \frac{1}{n} & \dots & \frac{1}{n} \end{pmatrix}$$

$$\begin{pmatrix} \frac{\sum_{i=1}^{n} x_i}{n} \\ \frac{\sum_{i=1}^{n} x_i}{n} \\ \vdots \\ \frac{\sum_{i=1}^{n} x_i}{n} \end{pmatrix} = \begin{pmatrix} \frac{1}{n} & \frac{1}{n} & \dots & \frac{1}{n} \\ \frac{1}{n} & \frac{1}{n} & \dots & \frac{1}{n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{n} & \frac{1}{n} & \dots & \frac{1}{n} \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

With

$$|y\rangle = \begin{pmatrix} \frac{1}{\sqrt{n}} \\ \vdots \\ \frac{1}{\sqrt{n}} \\ -\frac{1}{\sqrt{n}} \\ \frac{1}{\sqrt{n}} \\ \vdots \\ \frac{1}{\sqrt{n}} \end{pmatrix} = \frac{(-1)^{f(x)}}{\sqrt{n}} \cdot \begin{pmatrix} 1 \\ \vdots \\ 1 \\ 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} = \frac{(-1)^{f(x)}}{\sqrt{n}} \cdot |\mathbf{1}\rangle$$

the new amplitude distribution is computed by

$$G_m \cdot |y\rangle = (2 \cdot P_m - I_m) \cdot |y\rangle = 2 \cdot P_m \cdot |y\rangle - I_m \cdot |y\rangle.$$

First two times the average amplitude is computed. The amplitude values for non solution are $\frac{1}{\sqrt{n}}$ and for one marked solution $-\frac{1}{\sqrt{n}}$, it follows

$$2 \cdot P_m \cdot |y\rangle = A \cdot |\mathbf{1}\rangle = \frac{2}{n} \cdot \left(n \cdot \frac{1}{\sqrt{n}} - \frac{1}{\sqrt{n}} + \left(-\frac{1}{\sqrt{n}} \right) \right) \cdot |\mathbf{1}\rangle$$

$$A \cdot |\mathbf{1}\rangle = \frac{2}{n} \cdot \left((n-1) \cdot \frac{1}{\sqrt{n}} - \frac{1}{\sqrt{n}} \right) \cdot |\mathbf{1}\rangle$$

$$A = \frac{2}{\sqrt{n}} \cdot \left(1 - \frac{2}{n} \right) = \frac{2 \cdot n - 4}{n^{\frac{3}{2}}}$$

and the amplitude distribution is

$$G_m \cdot |y\rangle = A \cdot |\mathbf{1}\rangle - I_m \cdot |y\rangle.$$

The amplitude of the state $|\tau\rangle$ indicating the solution in the dimension i is

$$\tau_i = A + \frac{1}{\sqrt{n}} = \frac{3 \cdot n - 4}{n^{\frac{3}{2}}} = \frac{3}{\sqrt{n}} - \frac{4}{n \cdot \sqrt{n}}$$

and the non solution in the dimension j with $j \neq i$

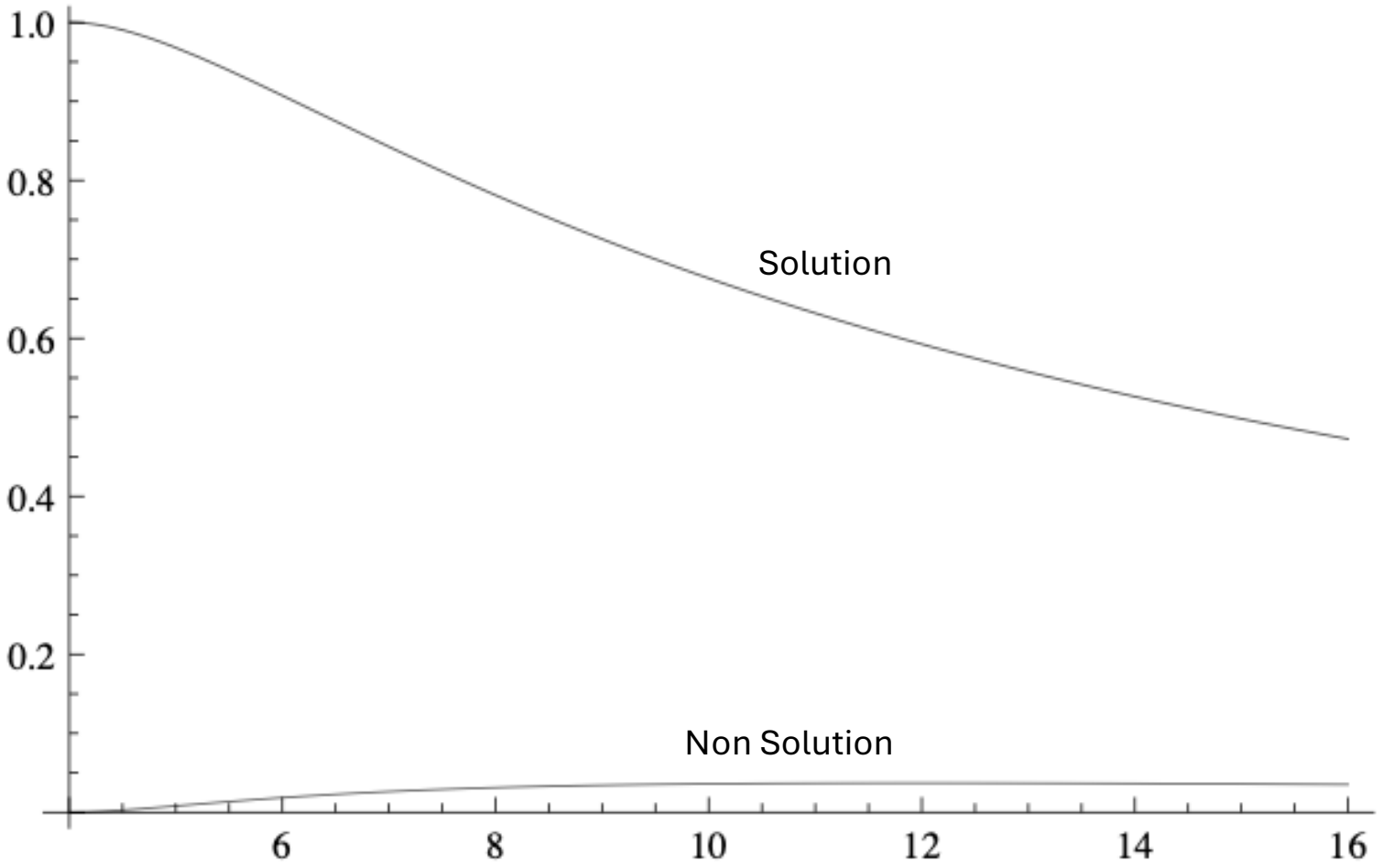
$$\tau_j = A - \frac{1}{\sqrt{n}} = \frac{n - 4}{n^{\frac{3}{2}}} = \frac{1}{\sqrt{n}} - \frac{4}{n \cdot \sqrt{n}}.$$

For $n = 4$ $\tau_i = 1$ and $\tau_j = 0$, for $n = 2^8$ $\tau_i = 0.186523$ and $\tau_j = 0.0615234$.
The probability of measuring the solution depending on the size n is

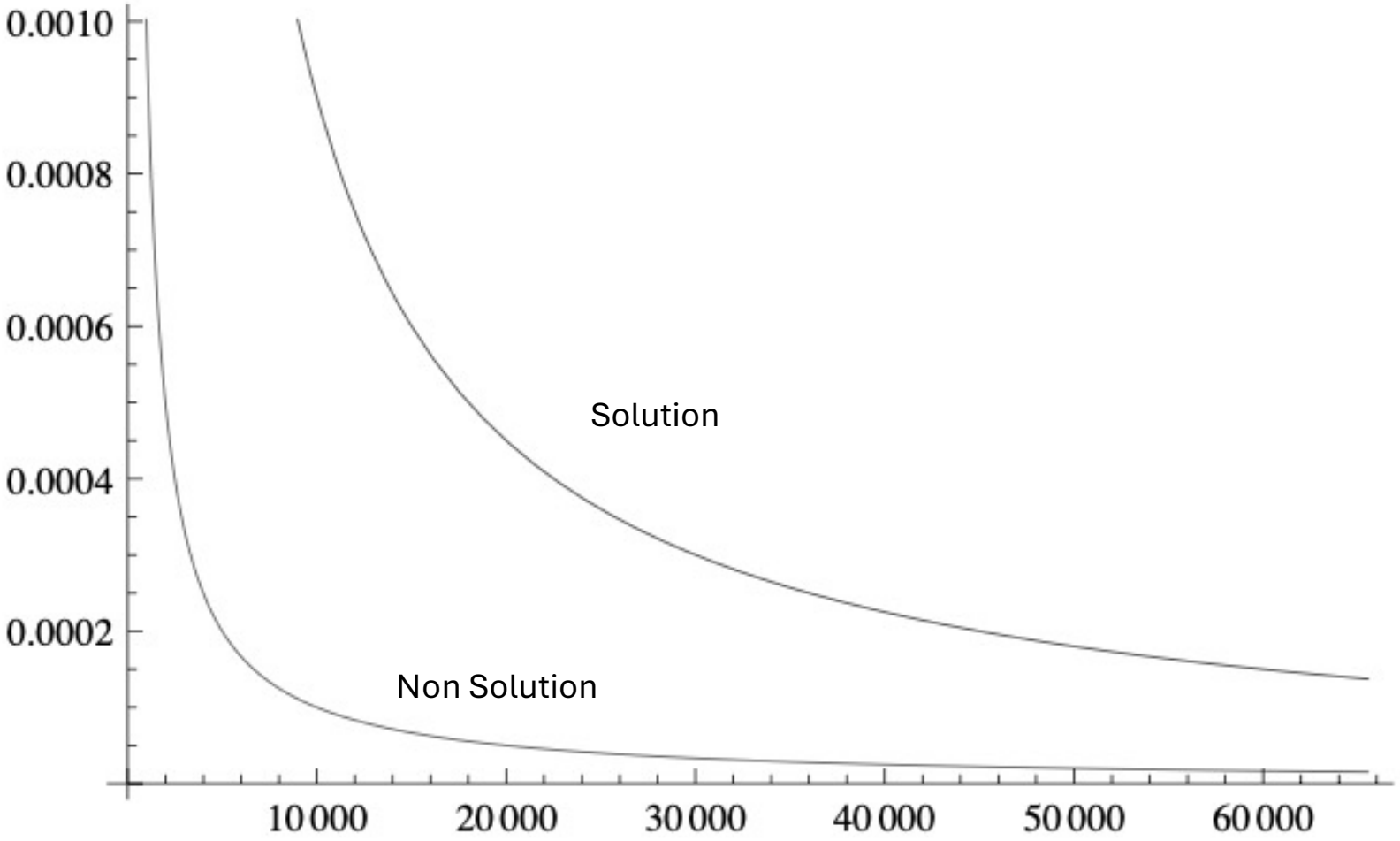
$$p(solution) = \left| \frac{3}{\sqrt{n}} - \frac{4}{n \cdot \sqrt{n}} \right|^2$$

and non solution

$$p(non\ solution) = \left| \frac{1}{\sqrt{n}} - \frac{4}{n \cdot \sqrt{n}} \right|^2.$$



The *probability* of measuring **one** present solution and **one** non solution for $4 \leq n \leq 16$



The *probability* of measuring **one** present solution and **one** non solution for $2^8 \leq n \leq 2^{16}$

Iterative Amplification

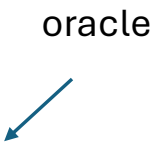
- The probability of seeing one solution should be as close as possible to **1** and the number of iterations should be small
- The number of iterations *r* is the largest integer not greater than t^*

$$r = \lfloor t^* \rfloor = \left\lfloor \frac{\pi}{4} \cdot \sqrt{\frac{2^m}{k}} \right\rfloor$$

- The value of r depends on the relation of $n=2^m$ versus *k*, *k*=number of solutions
  - For $n = 4$ and $k = 1$ we need **only** one rotation
  - For $k=n/4$ we need **only** one rotation to measure **one** solution

Iterative Amplification

With the definition



$$\Gamma_m := (G_m \otimes I_1) \cdot \boldsymbol{T}$$

the resulting state is $|\tau\rangle$. We describe several amplifications by

$$\left(\prod_{t=1}^{r} \Gamma_m\right) \cdot H_{m+1} \cdot |0^{\oplus m}\rangle|1\rangle = \Gamma_m \cdot \Gamma_m \cdot \cdots \cdot \Gamma_m \cdot H_{m+1} \cdot |0^{\oplus m}\rangle|1\rangle =$$

$$= |\tau\rangle \otimes \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right).$$

How many iterations do we need to preform, or what is the suitable value for r? The probability of seeing one solution should be as close as possible to 1 and r should be as small as possible.

- Suppose that the function o(x) has k solutions with $n \gg k \geq 1$,

$$o_\xi(x) = \begin{cases} 1 & if \quad x = \xi_i \quad i \in \{1, 2, \cdots, k\} \\ 0 & else \end{cases}$$

- The amplitude for the solution at the iteration t will be indicated by α_t non solution by β_t
- For $t = 0$ before the first iteration of Γ_m

$$\alpha_0 = \frac{1}{\sqrt{n}} \qquad \beta_0 = \frac{1}{\sqrt{n}}.$$

- In the first iteration Γ_m two times the average amplitude is computed

$$A = \frac{2}{n} \cdot \left(n \cdot \frac{1}{\sqrt{n}} - \frac{k}{\sqrt{n}} - \frac{k}{\sqrt{n}} \right) = \frac{2}{n} \cdot \left((n-k) \cdot \frac{1}{\sqrt{n}} - \frac{k}{\sqrt{n}} \right)$$

$$A = \frac{2}{n} \cdot \left((n-k) \cdot \beta_0 - k \cdot \alpha_0 \right).$$

The amplitude of the solution is

$$\alpha_1 = A + \frac{1}{\sqrt{n}} = A + \alpha_0 = \frac{2}{n} \cdot ((n-k) \cdot \beta_0 - k \cdot \alpha_0) + \alpha_0,$$

$$\alpha_1 = \frac{1}{n} \cdot (\alpha_0 \cdot (n - 2 \cdot k) + \beta_0 \cdot (2 \cdot n - 2 \cdot k),$$

$$\alpha_1 = \alpha_0 \cdot (1 - \frac{2 \cdot k}{n}) + \beta_0 \cdot (2 - \frac{2 \cdot k}{n}),$$

and the non solution

$$\beta_1 = A - \frac{1}{\sqrt{n}} = A - \beta_0 = \frac{2}{n} \cdot ((n-k) \cdot \beta_0 - k \cdot \alpha_0) - \beta_0,$$

$$\beta_1 = -\alpha_0 \cdot \frac{2 \cdot k}{n} + \beta_0 \cdot (1 - \frac{2 \cdot k}{n}).$$

- We can describe the evolution of the amplitudes in time *t* by two coupled *recurrence* equations.
- They represent a discrete **dynamical system** of two **difference equations**

$$\alpha_{t+1} = \alpha_t \cdot \left(1 - \frac{2 \cdot k}{n}\right) + \beta_t \cdot \left(2 - \frac{2 \cdot k}{n}\right)$$

$$\beta_{t+1} = -\alpha_t \cdot \frac{2 \cdot k}{n} + \beta_t \cdot \left(1 - \frac{2 \cdot k}{n}\right).$$

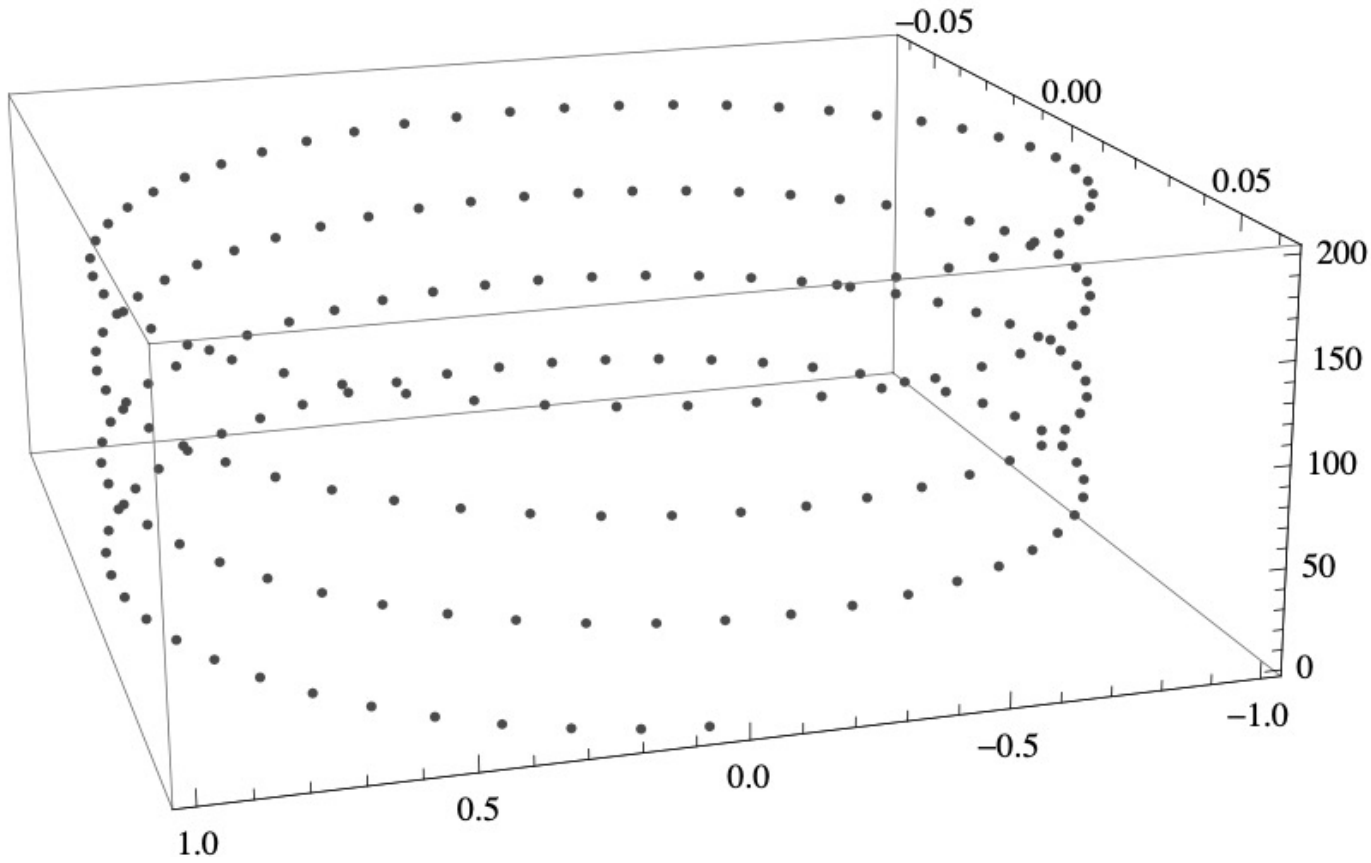


Fig. 10.4 Phase space ofα_t, β_t and t with the boundary condition of $\alpha_0 = \beta_0 = \frac{1}{\sqrt{n}}$. The values are $n = 256$, $k = 1$ and $1 \leq t \leq 200$. The x-axis indicates α_t, the y-axis β_t and the z-axis t.

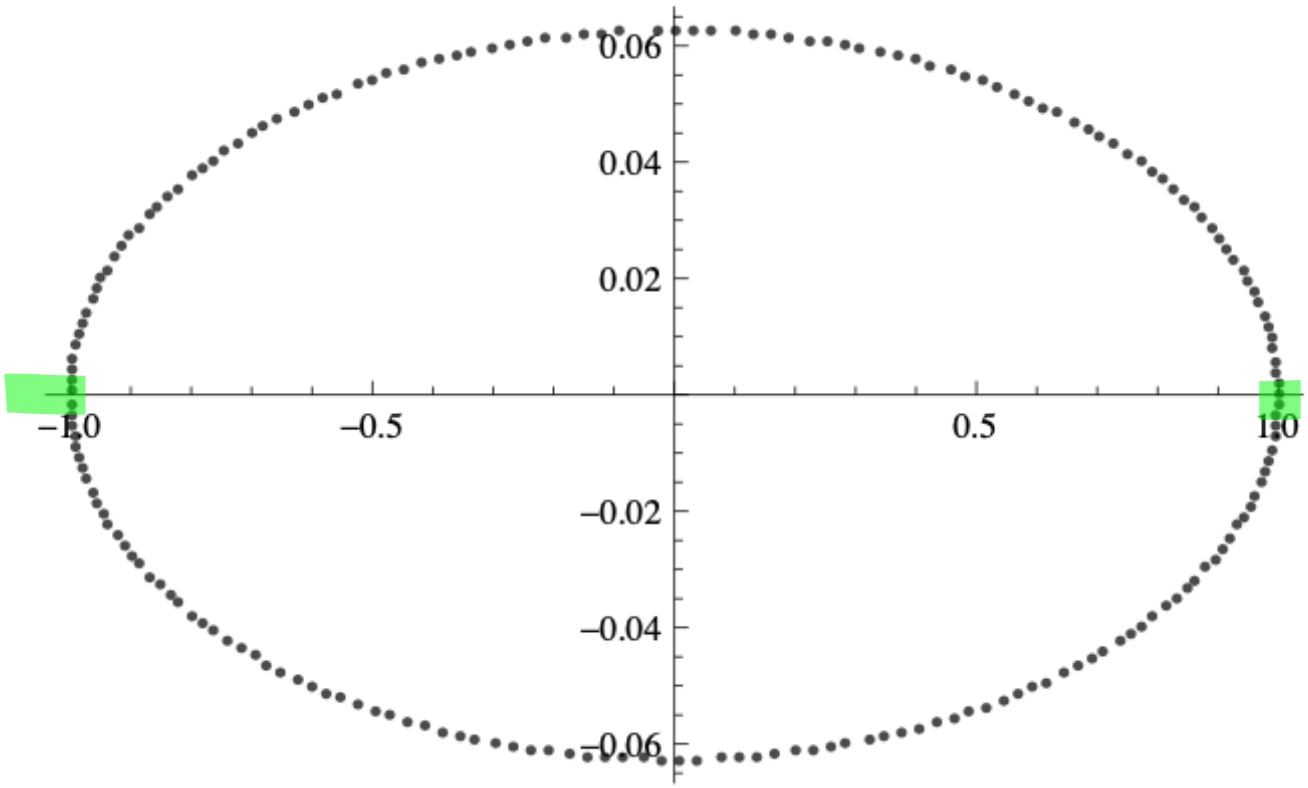


Fig. 10.5 The projected orbit in the two dimensional subspace α_t, β_t with the boundary condition of $\alpha_0 = \beta_0 = \frac{1}{\sqrt{n}}$. The values are $n = 256$, $k = 1$ and $1 \leq t \leq 200$. The x-axis indicates α_t and the y-axis β_t.

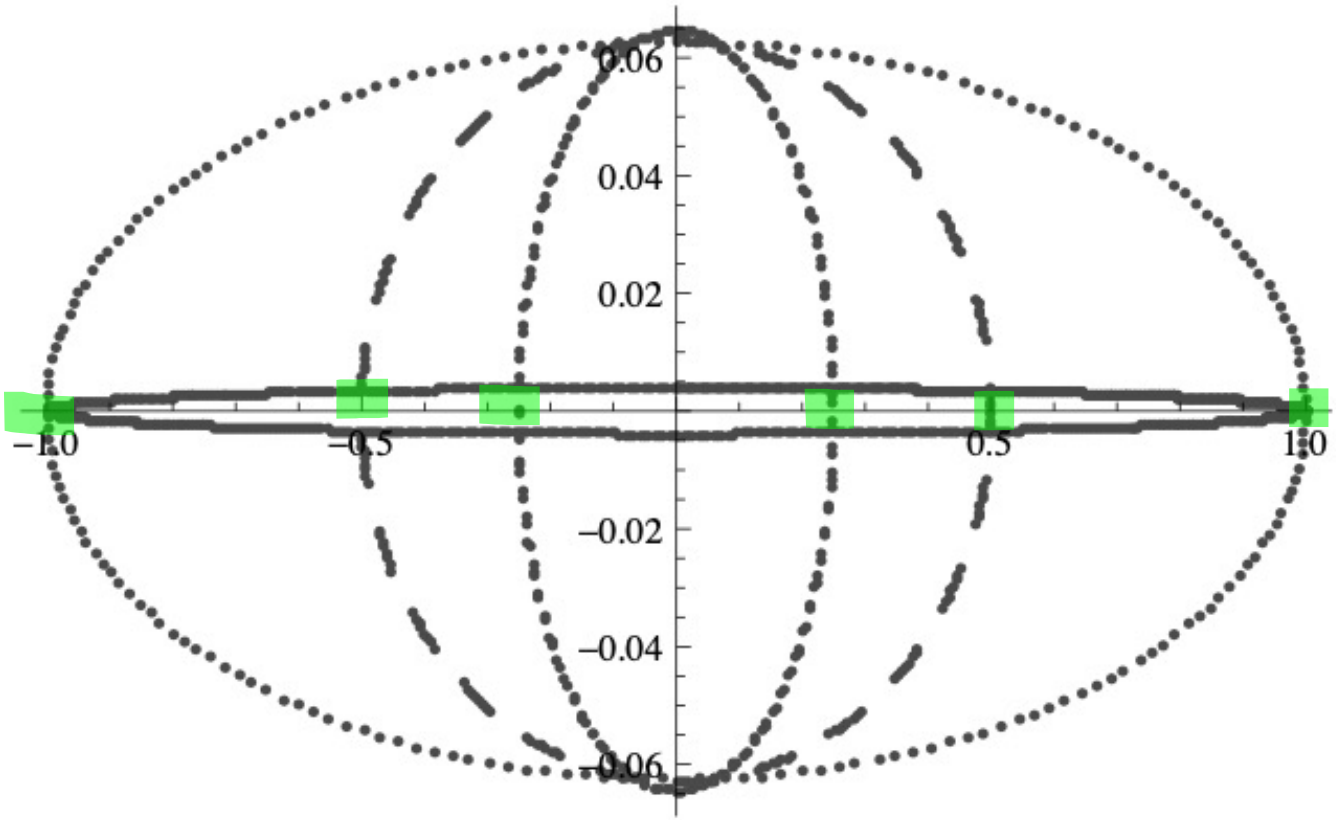


Fig. 10.6 The projected orbit in the two dimensional subspace α_t, β_t with the boundary condition of $\alpha_0 = \beta_0 = \frac{1}{\sqrt{n}}$. The x-axis indicates α_t, the y-axis β_t. The outer ellipse has the values $n = 256$, $k = 1$ as before, the two ellipses with diminishing x-axis radius corresponds to increased k values $k = 4$ and $k = 16$. For all three ellipses with $n = 256$, 200 iterations were done. In the fourth ellipse k is one and $n = 65536 = 2^{16}$, y-axis radius diminish. 1000 iterations were done.

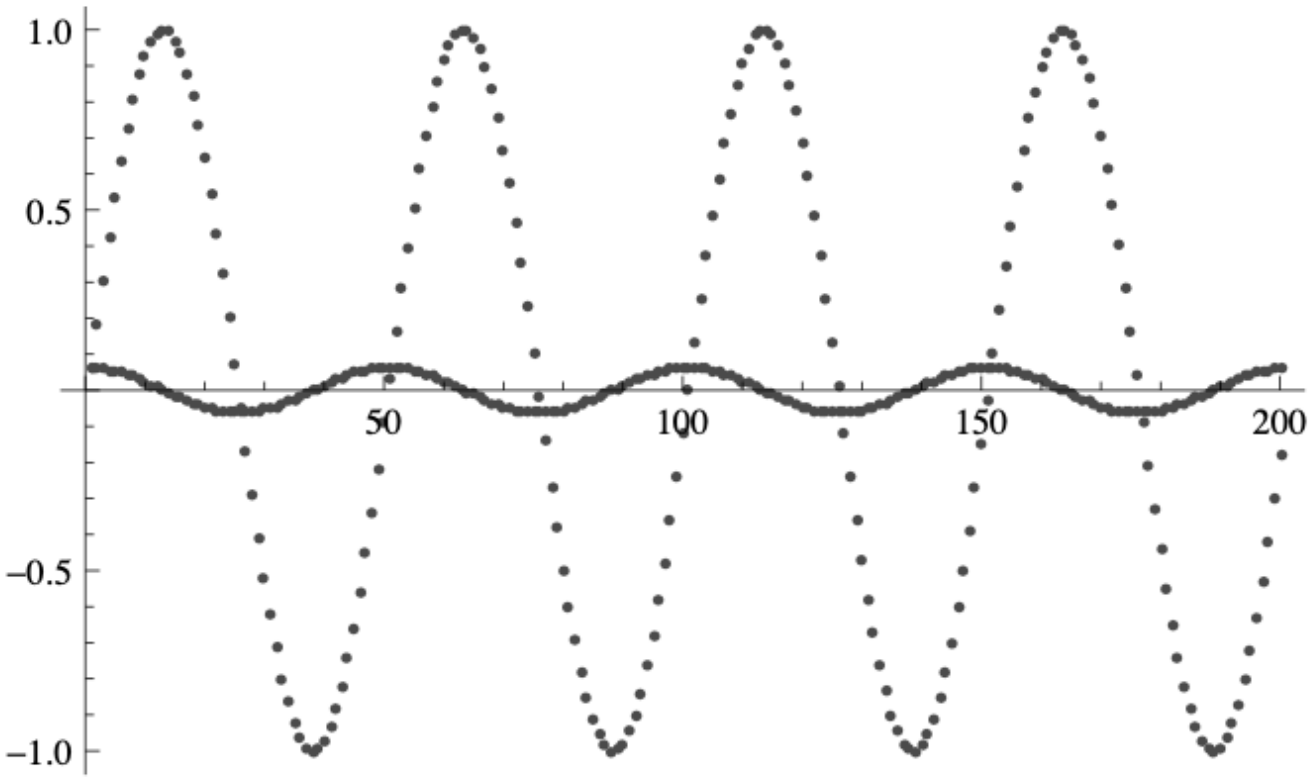
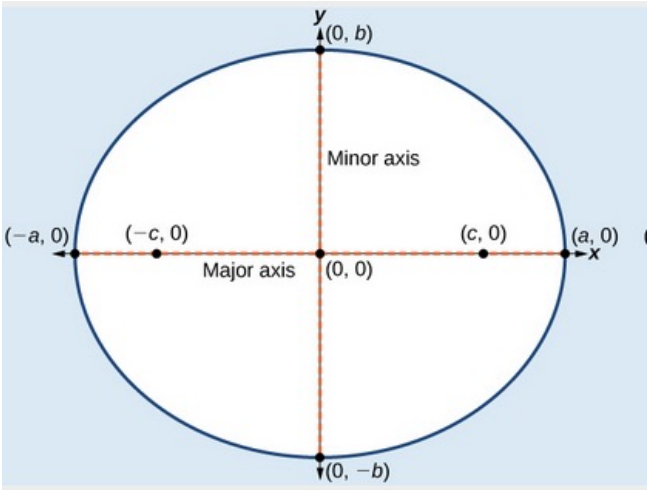


Fig. 10.7 The projected orbit in the two dimensional subspace of amplitude and time t represents a periodic function described by α_t (the dotted curve) and β_t (the continuous curve). The x-axis indicates t and the y-axis the amplitude. The values are $n = 256$, $k = 1$ and $1 \leq t \leq 200$.

- Such a linear and periodic system is usually described by sine and cosine equations
- The ellipse in a Cartesian system is represented by the equation



$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$$

- Because α_t and βt are real, following equation representing an ellipse is given by

$$k \cdot \alpha_t^2 + \ (n-k) \cdot \beta_t^2 = 1.$$

Using the the Pythagorean identity

$$\sin^2\theta + \cos^2\theta = 1$$

we can rewrite the equation representing the ellipse as

$$k \cdot \left(\sqrt{\frac{1}{k}} \cdot \sin\theta_t \right)^2 + (n-k) \cdot \left(\sqrt{\frac{1}{n-k}} \cdot \cos\theta_t \right)^2 = 1$$

$$\alpha_{t+1} = \alpha_t \cdot \left(1 - \frac{2 \cdot k}{n}\right) + \beta_t \cdot \left(2 - \frac{2 \cdot k}{n}\right)$$

$$\beta_{t+1} = -\alpha_t \cdot \frac{2 \cdot k}{n} + \beta_t \cdot \left(1 - \frac{2 \cdot k}{n}\right).$$

it follows that

$$\alpha_t = \sqrt{\frac{1}{k}} \cdot \sin\theta_t \quad \textit{and} \quad \beta_t = \sqrt{\frac{1}{n-k}} \cdot \cos\theta_t$$

and we can rewrite the the two coupled recurrence equations as

$$\sqrt{\frac{1}{k}} \cdot \sin\theta_{t+1} = \sqrt{\frac{1}{k}} \cdot \sin\theta_t \cdot \left(1 - \frac{2 \cdot k}{n}\right) + \sqrt{\frac{1}{n-k}} \cdot \cos\theta_t \cdot \left(2 - \frac{2 \cdot k}{n}\right)$$

$$\sqrt{\frac{1}{n-k}} \cdot \cos\theta_{t+1} = -\sqrt{\frac{1}{k}} \cdot \sin\theta_t \cdot \frac{2 \cdot k}{n} + \sqrt{\frac{1}{n-k}} \cdot \cos\theta_t \cdot \left(1 - \frac{2 \cdot k}{n}\right)$$

simplified as

$$\sin\theta_{t+1} = \sin\theta_t \cdot \left(1 - \frac{2\cdot k}{n}\right) + \cos\theta_t \cdot \frac{2\cdot\sqrt{k}\cdot\sqrt{n-k}}{n}$$

$$\cos\theta_{t+1} = -\sin\theta_t \cdot \frac{2\cdot\sqrt{k}\cdot\sqrt{n-k}}{n} + \cos\theta_t \cdot \left(1 - \frac{2\cdot k}{n}\right).$$

Trigonometric simplification

Because

$$-1 \leq \left(1 - \frac{2 \cdot k}{n}\right) \leq 1$$

we can represent it as

$$\cos \omega = 1 - \frac{2 \cdot k}{n}.$$

Because of the Pythagorean identity

$$\left(1 - \frac{2 \cdot k}{n}\right)^2 + \left(\frac{2 \cdot \sqrt{k} \cdot \sqrt{n-k}}{n}\right)^2 = 1$$

it follows that

$$\sin \omega = \frac{2 \cdot \sqrt{k} \cdot \sqrt{n-k}}{n}$$

$$\sin(\alpha+\beta) = \sin\alpha\cos\beta + \cos\alpha\sin\beta$$
$$\sin(\alpha-\beta) = \sin\alpha\cos\beta - \cos\alpha\sin\beta$$
$$\cos(\alpha+\beta) = \cos\alpha\cos\beta - \sin\alpha\sin\beta$$
$$\cos(\alpha-\beta) = \cos\alpha\cos\beta + \sin\alpha\sin\beta$$

- We can rewrite again the the two coupled recurrence equations as

$$\sin\theta_{t+1} = \sin\theta_t \cdot \cos\omega + \cos\theta_t \cdot \sin\omega$$

$$\cos\theta_{t+1} = -\sin\theta_t \cdot \sin\omega + \cos\theta_t \cdot \cos\omega$$

- Because of the trigonometric identities, addition and subtraction theorem, the two coupled recurrence equations with the boundary condition θ_0 are

$$\sin\theta_{t+1} = \sin(\theta_t + \omega) = \sin(\theta_0 + t\cdot\omega + \omega)$$

$$\cos\theta_{t+1} = \cos(\theta_t + \omega) = \cos(\theta_0 + t\cdot\omega + \omega)$$

$$\sin\theta_{t+1} = \sin(\theta_t + \omega) = \sin(\theta_0 + t \cdot \omega + \omega)$$

$$\cos\theta_{t+1} = \cos(\theta_t + \omega) = \cos(\theta_0 + t \cdot \omega + \omega)$$

we can rewrite the recurrence equations into two simple equations

$$\alpha_t = \sqrt{\frac{1}{k}} \cdot \sin\theta_t \longrightarrow \alpha_t = \frac{1}{\sqrt{k}} \cdot \sin(\theta_0 + t \cdot \omega)$$

$$\beta_t = \sqrt{\frac{1}{n-k}} \cdot \cos\theta_t \longrightarrow \beta_t = \frac{1}{\sqrt{n-k}} \cdot \cos(\theta_0 + t \cdot \omega).$$

- With the boundary conditions

$$\alpha_0 = \beta_0 = \frac{1}{\sqrt{n}}$$

$$\alpha_0 = \sqrt{\frac{1}{k}} \cdot \sin\theta_0$$

$$\beta_0 = \frac{1}{\sqrt{n-k}} \cdot \cos\theta_0$$

it follows that:

$$\sin^2\theta_0 = \frac{k}{n} \longrightarrow \theta_0 = \sin^{-1}\left(\sqrt{\frac{k}{n}}\right)$$

$$\cos^2\theta_0 = \frac{n-k}{n} = 1 - \frac{k}{n}$$

and because of the double angle formula

$$\cos(2 \cdot x) = 2 \cdot \cos^2 x - 1 = 1 - 2 \cdot \sin^2 x$$

and

$$\sin(2 \cdot x) = 2 \cdot \sin x \cdot \cos x$$

it follows that

$$\cos \omega = 1 - \frac{2 \cdot k}{n} = 1 - 2 \cdot \sin^2 \theta_0 = \cos(2 \cdot \theta_0).$$

$$\sin \omega = \frac{2 \cdot \sqrt{k} \cdot \sqrt{n-k}}{n} = 2 \sin \theta_0 \cdot \cos_0 \theta_0 = \sin(2 \cdot \theta_0).$$

$$\alpha_t = \frac{1}{\sqrt{k}} \cdot \sin(\theta_0 + t \cdot \omega)$$

$$\beta_t = \frac{1}{\sqrt{n-k}} \cdot \cos(\theta_0 + t \cdot \omega).$$

$$\sin(\alpha+\beta) = \sin\alpha\cos\beta + \cos\alpha\sin\beta$$
$$\sin(\alpha-\beta) = \sin\alpha\cos\beta - \cos\alpha\sin\beta$$
$$\cos(\alpha+\beta) = \cos\alpha\cos\beta - \sin\alpha\sin\beta$$
$$\cos(\alpha-\beta) = \cos\alpha\cos\beta + \sin\alpha\sin\beta$$

Using the trigonometric identities

$$\alpha_t = \frac{1}{\sqrt{k}} \cdot (\sin\theta_0 \cdot \cos(t \cdot \omega) + \cos\theta_0 \cdot \sin(t \cdot \omega))$$

$$\beta_t = \frac{1}{\sqrt{n-k}} \cdot (-\sin\theta_0 \cdot \sin(t \cdot \omega) + \cos\theta_0 \cdot \cos(t \cdot \omega))$$

Solution:

...with boundary conditions: $\alpha_0 = \beta_0 = \frac{1}{\sqrt{n}}$

$$\alpha_t = \frac{1}{\sqrt{k}} \cdot \sin(\theta_0 + t \cdot 2 \cdot \theta_0) = \frac{1}{\sqrt{k}} \cdot \sin(\theta_0 \cdot (2 \cdot t + 1))$$

$$\beta_t = \frac{1}{\sqrt{n-k}} \cdot \cos(\theta_0 + t \cdot 2 \cdot \theta_0) = \frac{1}{\sqrt{n-k}} \cdot \cos(\theta_0 \cdot (2 \cdot t + 1))$$

with

$$\theta_0 = \sin^{-1}\left(\sqrt{\frac{k}{n}}\right)$$

Number of Iterations

$$\alpha_t = \frac{1}{\sqrt{k}} \cdot \sin(\theta_0 \cdot (2 \cdot t + 1))$$

- The probability of seeing one solution should be as close as possible to 1 and the number of iterations *r* should be as small as possible.
  - Because there are *k* solution, the probability of measuring a state that represents a solution is

$$k \cdot \alpha_t^2 = \sin^2(\theta_0 \cdot (2 \cdot t + 1)) = 1$$

$$\theta_0 = \sin^{-1}\left(\sqrt{\frac{k}{n}}\right)$$

$$\theta_0 \cdot (2 \cdot t + 1) = \frac{\pi}{2}$$

after t^* iterations the probability of measuring a solution is nearly one

$$t^* := t = \frac{\pi}{4 \cdot \theta_0} - \frac{1}{2} = \frac{\pi}{4} \cdot \sqrt{\frac{n}{k}} - \frac{1}{2} = \frac{\pi}{4} \cdot \sqrt{\frac{2^m}{k}} - \frac{1}{2}$$

$$t^* = 0.785398 \cdot \sqrt{\frac{2^m}{k}} - 0.5.$$

Number of Iterations

- The number of iterations r is the largest integer not greater than t^*,

$$r = \lfloor t^* \rfloor = \left\lfloor \frac{\pi}{4} \cdot \sqrt{\frac{2^m}{k}} \right\rfloor$$

The value of r depends on the relation of n versus k. For $n = 4$ and $k = 1$ we need only one rotation, we need as well only one rotation for

$$\frac{n}{4} = k$$

to find **one** of the k solutions. For 16 qubits and one solution, $k = 1$, $n = 65536 = 2^{16}$, $t^* = 200.562$. In this case we need two hundred rotations. The probability of measuring a state that represents a solution is nearly one.

- It is possible to adapt the iterations in such a way that the probability of finding a solution is exactly one.
  - One changes θ_0 of the difference equations either in the last step or continuously
- The resulting speed up remains quadratic
- The iterative amplification algorithm requires the value of k in order to determine the number of iterations
- We can determine the value of k by the quantum counting algorithm

Lower Bound

- If the operator T is represented by $O(m)$ gates, then we can get a speed up of maximum $O(\sqrt{n})$ steps, in fact $\Omega(\sqrt{n})$ *is* the **lower bound**.
- The proof for lower bound $\Omega(\sqrt{n})$ for T oracle-based search is based on a sequence in a Hilbert space and the Euclidean norm properties of Hilbert space
- The definition of the probabilities as normed squared amplitudes

- **No better algorithm exists!**

It follows that using a quantum computer NP – complete problems remain NP – complete

Unitary Representation

$$P_m = |x\rangle\langle x| = \begin{pmatrix} \frac{1}{n} & \frac{1}{n} & \cdots & \frac{1}{n} \\ \frac{1}{n} & \frac{1}{n} & \cdots & \frac{1}{n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{n} & \frac{1}{n} & \cdots & \frac{1}{n} \end{pmatrix}$$

$$Q_x = I_m - 2 \cdot P_m.$$

Grover's amplification (for $m \geq 2$) is based on $G_m = -Q_x$. The unitary operator Λ_m reverses the sign of $|0\rangle$

$$\Lambda_m \cdot |0\rangle = -|0\rangle$$

and for $|x\rangle \neq |0\rangle$

$$\Lambda_m \cdot |x\rangle = |x\rangle.$$

Then we can write

$$G_m = -Q_x = -I_m + 2 \cdot P_m = 2 \cdot P_m - I_m = -(H_m \cdot \Lambda_m \cdot H_m) \tag{5.17}$$

with

$$H_m \cdot \Lambda_m \cdot H_m = \begin{pmatrix} 1 - \frac{2}{n} & -\frac{2}{n} & \cdots & -\frac{2}{n} \\ -\frac{2}{n} & 1 - \frac{2}{n} & \cdots & -\frac{2}{n} \\ \vdots & \vdots & \ddots & \vdots \\ -\frac{2}{n} & -\frac{2}{n} & \cdots & 1 - \frac{2}{n} \end{pmatrix}. \tag{5.18}$$

- We will demonstrate the example of $m = 3$, means $8 = 2^3$ different states
  - *G=-H3*L3*H3*

```
H3=
 [[ 0.3536  0.3536  0.3536  0.3536  0.3536  0.3536  0.3536  0.3536]
 [ 0.3536 -0.3536  0.3536 -0.3536  0.3536 -0.3536  0.3536 -0.3536]
 [ 0.3536  0.3536 -0.3536 -0.3536  0.3536  0.3536 -0.3536 -0.3536]
 [ 0.3536 -0.3536 -0.3536  0.3536  0.3536 -0.3536 -0.3536  0.3536]
 [ 0.3536  0.3536  0.3536  0.3536 -0.3536 -0.3536 -0.3536 -0.3536]
 [ 0.3536 -0.3536  0.3536 -0.3536 -0.3536  0.3536 -0.3536  0.3536]
 [ 0.3536  0.3536 -0.3536 -0.3536 -0.3536 -0.3536  0.3536  0.3536]
 [ 0.3536 -0.3536 -0.3536  0.3536 -0.3536  0.3536  0.3536 -0.3536]]
```

```
L3=np.matrix([[-1., 0., 0., 0., 0., 0., 0., 0.],
        [0., 1., 0., 0., 0., 0., 0., 0.],
        [0., 0., 1., 0., 0., 0., 0., 0.],
        [0., 0., 0., 1., 0., 0., 0., 0.],
        [0., 0., 0., 0., 1., 0., 0., 0.],
        [0., 0., 0., 0., 0., 1., 0., 0.],
        [0., 0., 0., 0., 0., 0., 1., 0.],
        [0., 0., 0., 0., 0., 0., 0., 1.]])
```

*G=-H3*L3*H3*

m = 3, n=8 = 2^3

$$G_m = \begin{pmatrix} \frac{2}{n}-1 & \frac{2}{n} & \cdots & \frac{2}{n} \\ \frac{2}{n} & \frac{2}{n}-1 & \cdots & \frac{2}{n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{2}{n} & \frac{2}{n} & \cdots & \frac{2}{n}-1 \end{pmatrix}$$

$$G3 = \begin{pmatrix} -\frac{3}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & -\frac{3}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & -\frac{3}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & -\frac{3}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & -\frac{3}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & -\frac{3}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & -\frac{3}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & -\frac{3}{4} \end{pmatrix}$$

In the next step we represent the state vector with equally distributed amplitudes and mark the solution with a minus sign

```
a=1/np.sqrt(8)
x1=np.array([a,a,a,a,-a,a,a,a])
print("x1=\n",x1)
x1=
 [ 0.3536  0.3536  0.3536  0.3536 -0.3536  0.3536  0.3536  0.3536]
```

and perform a step in Grover's amplification as with the resulting amplitudes

```
x1=G.dot(x1)
print("x1=\n",x1)
x1=
 [[0.1768 0.1768 0.1768 0.1768 0.8839 0.1768 0.1768 0.1768]]
```

and the second iteration with the resulting amplitudes

```
x2=np.array([0.1768, 0.1768, 0.1768, 0.1768, -0.8839, 0.1768, 0.1768, 0.1768])
x2=G.dot(x2)
print("x2=\n",x2)
x2=
 [[-0.0884 -0.0884 -0.0884 -0.0884  0.9723 -0.0884 -0.0884 -0.0884]].
```

With two rotations we achieved the maximal amplitude value that corresponds to the probability value $0.945367 = |0.9723|^2$. After the third rotation the amplitudes diminish because of the periodic property of Grover's amplification

Decomposition

With the Grove's amplification we have

$$G_m = -(H_m \cdot \Lambda_m \cdot H_m).$$

How can we decompose Λ_m by quantum gates? We note that for one qubit wi

$$\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \cdot \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

and

$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \cdot \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \cdot \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}$$

For three qubits we can define Λ_3 accordingly with

$$X_3 = X \otimes X \otimes X$$

$$H_0 = I \otimes I \otimes H$$

$$\Lambda_3 = X_3 \cdot H_0 \cdot CCX \cdot H_0 \cdot X_3$$

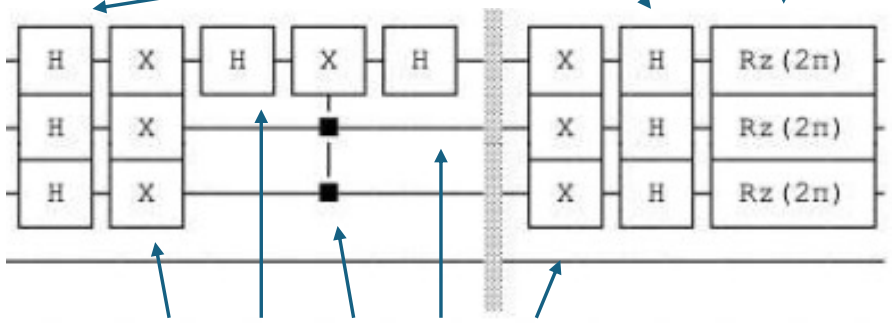
and for the minus sign operation

$$RZ(2 \cdot \pi)_3 = RZ(2 \cdot \pi) \otimes RZ(2 \cdot \pi) \otimes RZ(2 \cdot \pi)$$

it follows

$$G_m = -(H_3 \cdot \Lambda_3 \cdot H_3) = H_3 \cdot \Lambda_3 \cdot H_3 \cdot RZ(2 \cdot \pi)_3$$

"Grover Diffuser"



$$\Lambda_3 = X_3 \cdot H_0 \cdot CCX \cdot H_0 \cdot X_3$$

Example

- In the next step we apply Grover's amplification to a marked state of three qubits
- Our solution corresponds to the Boolean formula $\neg x \wedge y \wedge \neg z$
- for which it evaluates true, which is the case for
  - $x = 0$, $y = 1$ and $z = 0$
- In this case the state determined by the oracle function is
  - $o(010) = 1$ with the solution encoded by $(-1)^{o(x)}$
- The unitary operator T

$$T = (X \otimes I \otimes X \otimes I) \cdot MCX \cdot (X \otimes I \otimes X \otimes I)$$

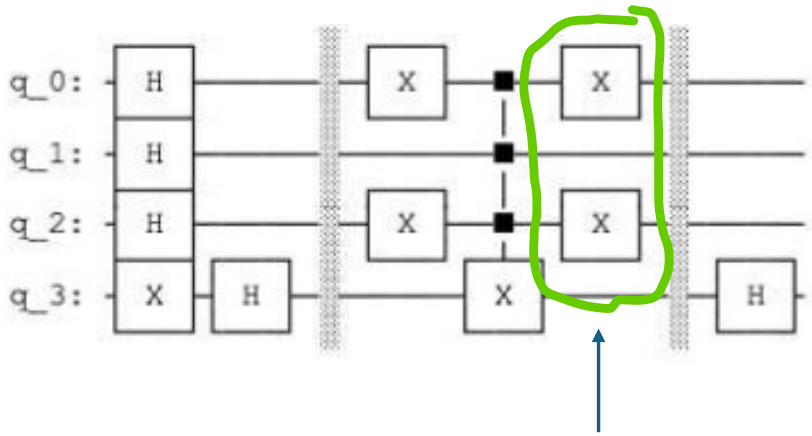
Un-Computation

$$H_0 = I \otimes I \otimes I \otimes H$$

and

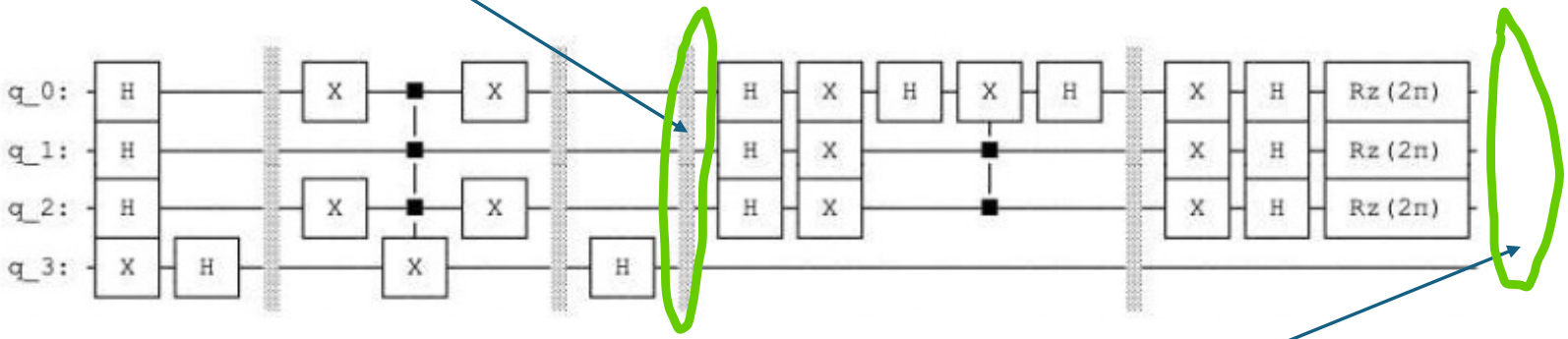
$$H_0 \cdot T \cdot H_4 \cdot |0001\rangle = \frac{1}{\sqrt{8}} \sum_{x \in B^3} (-1)^{o(x)} \cdot |x\rangle \otimes |0\rangle$$

the value of the function $o(x)$ is encoded by $(-1)^{o(x)}$, see as well Figure 5



$$Statevector = \left[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ \frac{1}{\sqrt{8}}\ \frac{1}{\sqrt{8}}\ -\frac{1}{\sqrt{8}}\ \frac{1}{\sqrt{8}}\ \frac{1}{\sqrt{8}}\ \frac{1}{\sqrt{8}}\ \frac{1}{\sqrt{8}}\ \frac{1}{\sqrt{8}}\right]$$

$$Statevector = \left[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ \frac{1}{\sqrt{8}}\ \frac{1}{\sqrt{8}}\ -\frac{1}{\sqrt{8}}\ \frac{1}{\sqrt{8}}\ \frac{1}{\sqrt{8}}\ \frac{1}{\sqrt{8}}\ \frac{1}{\sqrt{8}}\ \frac{1}{\sqrt{8}}\right]$$



$$Statevector = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0.1768\ 0.1768\ 0.8839\ 0.1768\ 0.1768\ 0.1768\ 0.1768\ 0.1768]$$

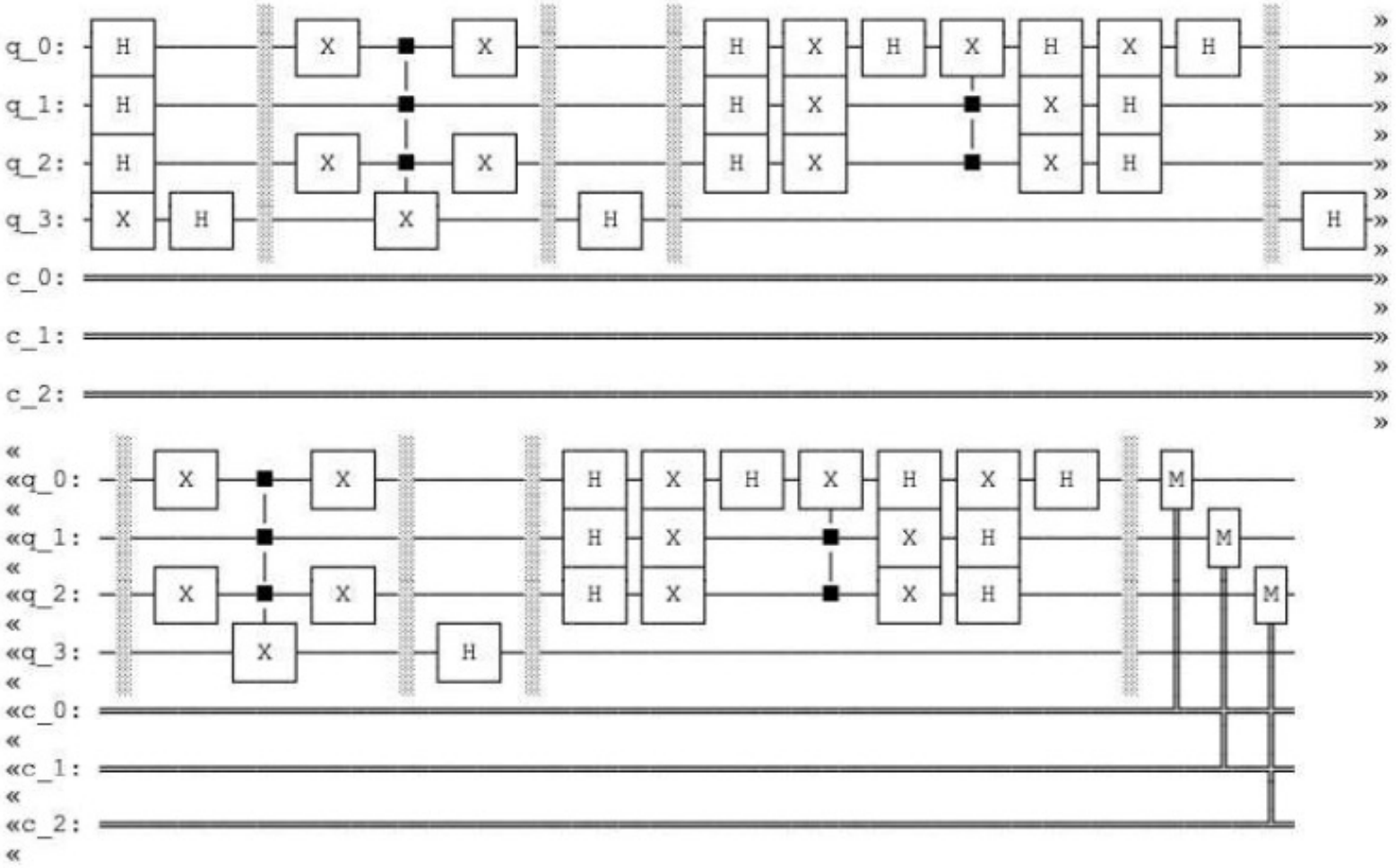
Two Rotations

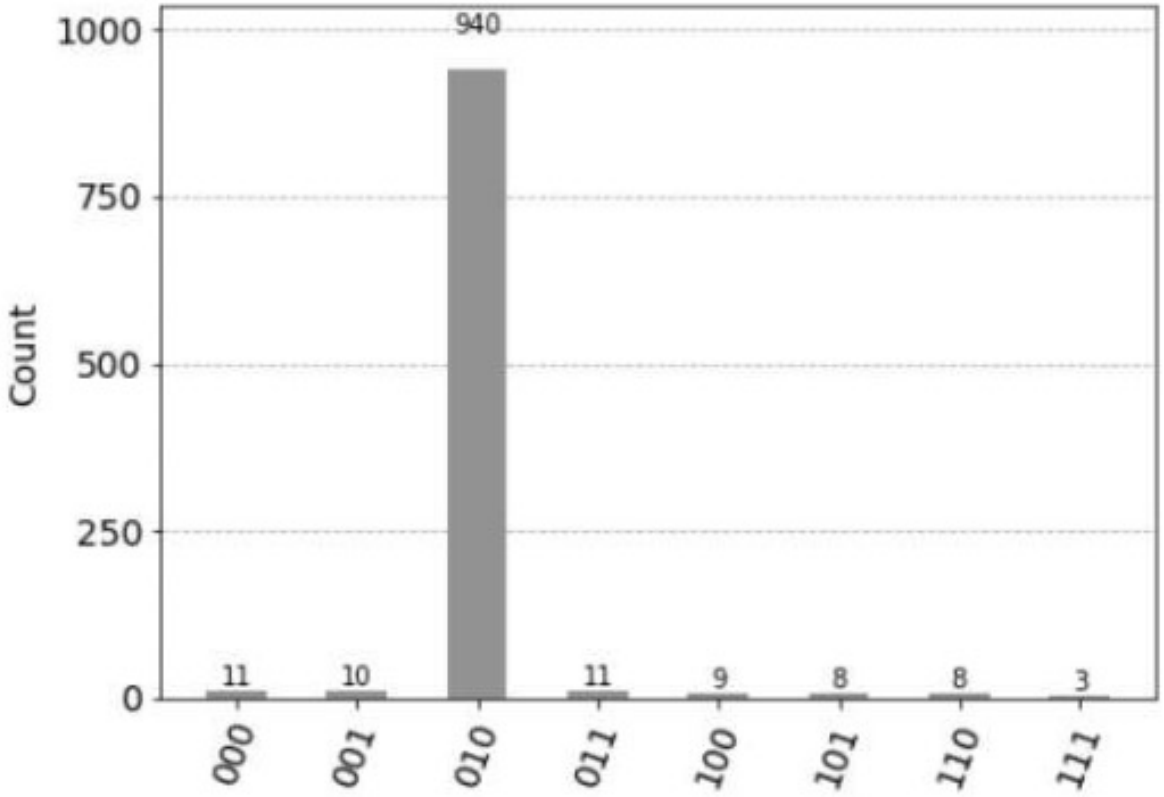
In the next step we will perform two rotations using *qasm simulator*. We do not minus sign operation

$$G_3 = -H_3 \cdot \Lambda_3 \cdot H_3$$

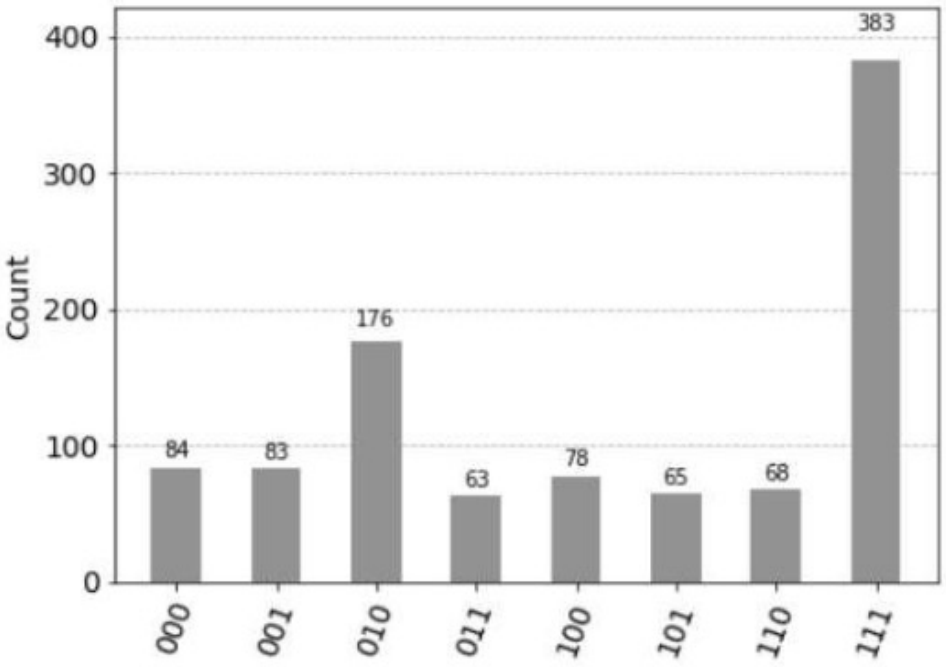
to get the correct result, what changes is the minus sign of over all amplitudes. This has no consequence for the resulting probabilities, so the Grover's amplification can be simplified to

$$G_3 = H_3 \cdot \Lambda_3 \cdot H_3$$


q_0:
q_1:
q_2:
q_3:
c_0:
c_1:
c_2:
H
X
M


1000
750
500
250
0
Count
940
11
10
11
9
8
8
3
000
001
010
011
100
101
110
111

- We may ask what would happen if we do not un-compute and define our oracle simply as $T = MCX \cdot (X \otimes I \otimes X \otimes I).$



- Histogram of Counts after two rotation of Grover's amplification without the un-computation of the oracle
- The resulting histogram of cost does not indicate the correct solution

For m qubits we can define Λ_m accordingly using the MCX gate

```
gate = MCXGate(m)
qc.append(gate, [0, 1, 2,...., m, m+1])
```

with

$$X_m = \underbrace{X \otimes X \otimes \cdots \otimes X}_{M \text{ times}}$$

$$H_0^m = \underbrace{I \otimes I \otimes \cdots \otimes I}_{M-1 \text{ times}} \otimes H$$

$$\Lambda_m = X_m \cdot H_0^m \cdot MCX \cdot H_0^m \cdot X_m$$

Alternatively Λ_m can be implemented efficiently with $f_0(x)$

$$f_0(x) = \begin{cases} 1 & if \quad x = 0 \\ 0 & else \end{cases}$$

as

$$\frac{1}{\sqrt{m}} \sum_{x \in B^m} (-1)^{f_0(x)} \cdot |x\rangle \otimes \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right).$$

SAT Problem

- The Boolean satisfiability problem (SAT problem) is the problem of determining if there exists an interpretation that satisfies a given Boolean formula, whether the formula evaluates true

The formula satisfiability problem is as follows, is a formula ϕ composed of

- m boolean variables: $x_1, x_2, \cdots, x_m$;
- k boolean connectivities: $\wedge$ (AND), $\vee$ OR $\neg$ (NOT), $\rightarrow$ (implication), $\leftrightarrow$ (if and only if);
- parentheses.

x	y	$x \to y$	$\neg x \vee y$
1	1	1	1
1	0	0	0
0	1	1	1
0	0	1	1

x	y	$x \leftrightarrow y$	$\neg(x \; XOR \; y)$
1	1	1	1
1	0	0	0
0	1	0	0
0	0	1	1

$$\phi = ((x_1 \to x_2) \vee \neg((\neg x_1 \leftrightarrow x_3) \vee x_4)) \wedge \neg x_2$$

has the interpretation $x_1 = 0$, $x_2 = 0$, $x_3 = 1$ and $x_4 = 1$ that satisfies ϕ with

$$\phi = ((0 \to 0) \vee \neg((\neg 0 \leftrightarrow 1) \vee 1)) \wedge \neg 0$$

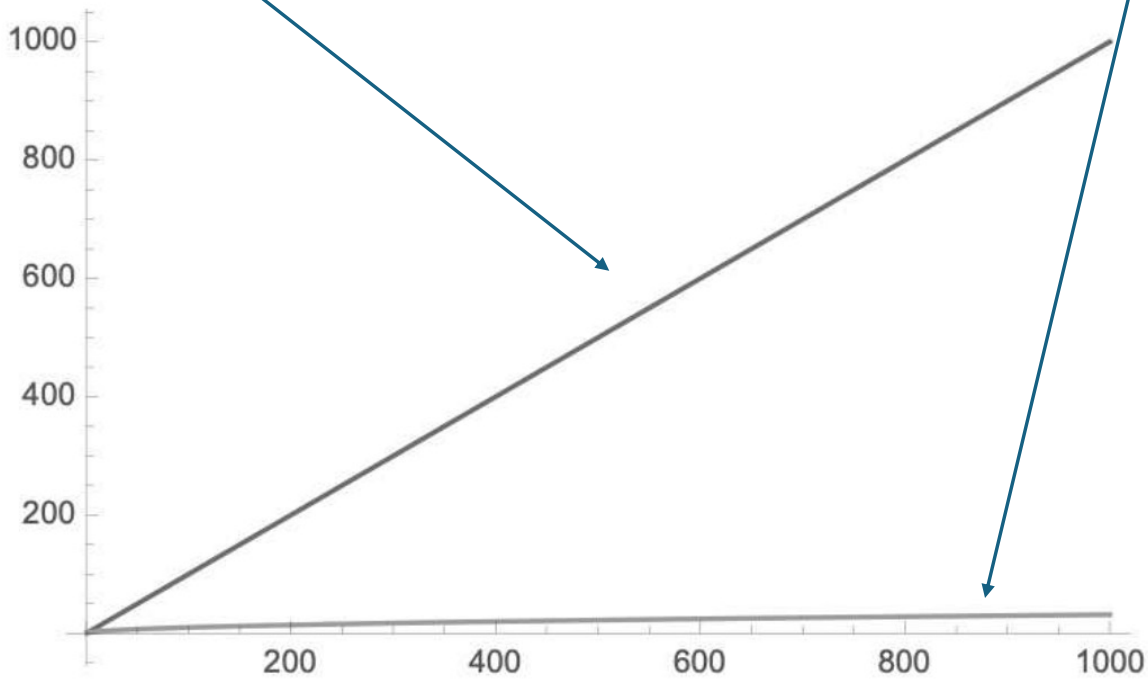
$$\phi = (1 \vee \neg(1 \vee 1)) \wedge 1$$

$$\phi = 1$$

There are 2^m possible assignments in a formula ϕ with m variables, the checking every assignment requires 2^m time on a conventional computer.

- A problem is easy if a conventional computer can determine the instances related to the input for the answer “Yes” in polynomial time
  - Polynomial-time algorithms are said to be fast since they can be executed in an acceptable time on a conventional computer and are called P
- Otherwise, we state that the problem is hard, means the time required grows exponentially and cannot be executed in an acceptable time on a conventional computer, such a problem is called NP
- A problem is called N P – complete if all possible instances must be examined by the conventional computer
  - A conventional computer can only solve a problem by checking all possible instances one after the other one

Despite the fact the saving of Grover's algorithm of $= O(\sqrt{n}) = O(2^{\frac{m}{2}})$ compared to $O(n) = O(2^m)$ is huge, $NP - complete$ problems remain $NP - complete$ on a quantum computer.



Why Huge and another Complexity Class?

saving of $O(2^{\frac{m}{2}})$ compared to $O(2^m)$ is huge, $2^m \neq O(2^{\frac{m}{2}})$ means

$$\Theta(2^{\frac{m}{2}}) \neq \Theta(2^m).$$

For $2^m \neq O(2^{\frac{m}{2}})$ we assume there exist a constant c, that for certain value $m_0 > 0$

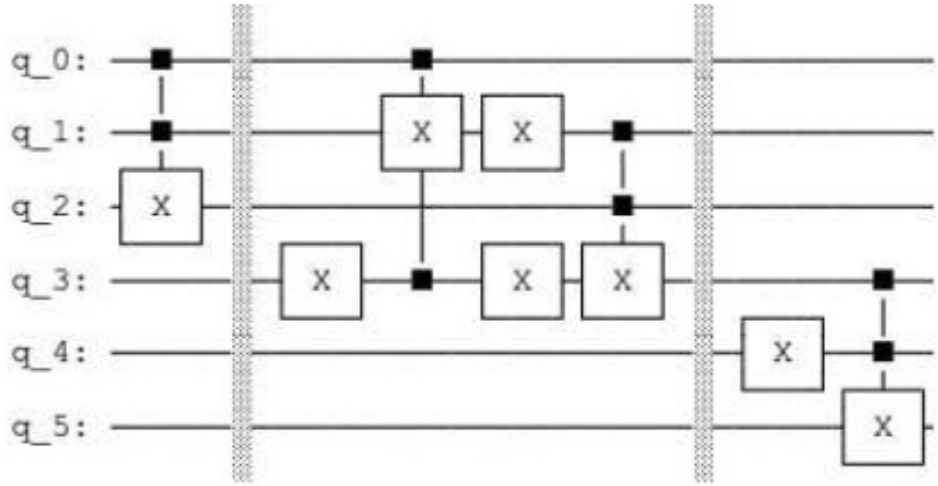
$$0 \leq 2^m \leq c \cdot 2^{\frac{m}{2}} \quad \forall m \geq m_0.$$

However such a constant does not exist because from

$$2^m = 2^{\frac{m}{2}} \cdot 2^{\frac{m}{2}} \leq c \cdot 2^{\frac{m}{2}}$$

follows the simple contradiction

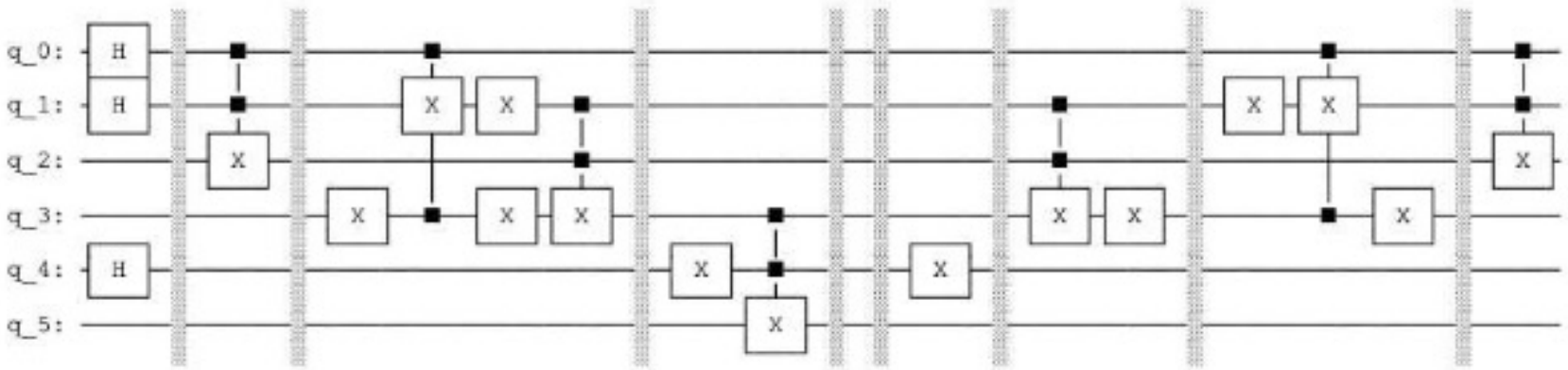
$$2^{\frac{m}{2}} \leq c.$$



In the next step we apply Grover's amplification to a marked state of three qubits. Our solution corresponds to the Boolean formula

$$\phi = (x_1 \leftrightarrow x_2) \wedge (x_1 \wedge x_2) \wedge \neg x_3$$

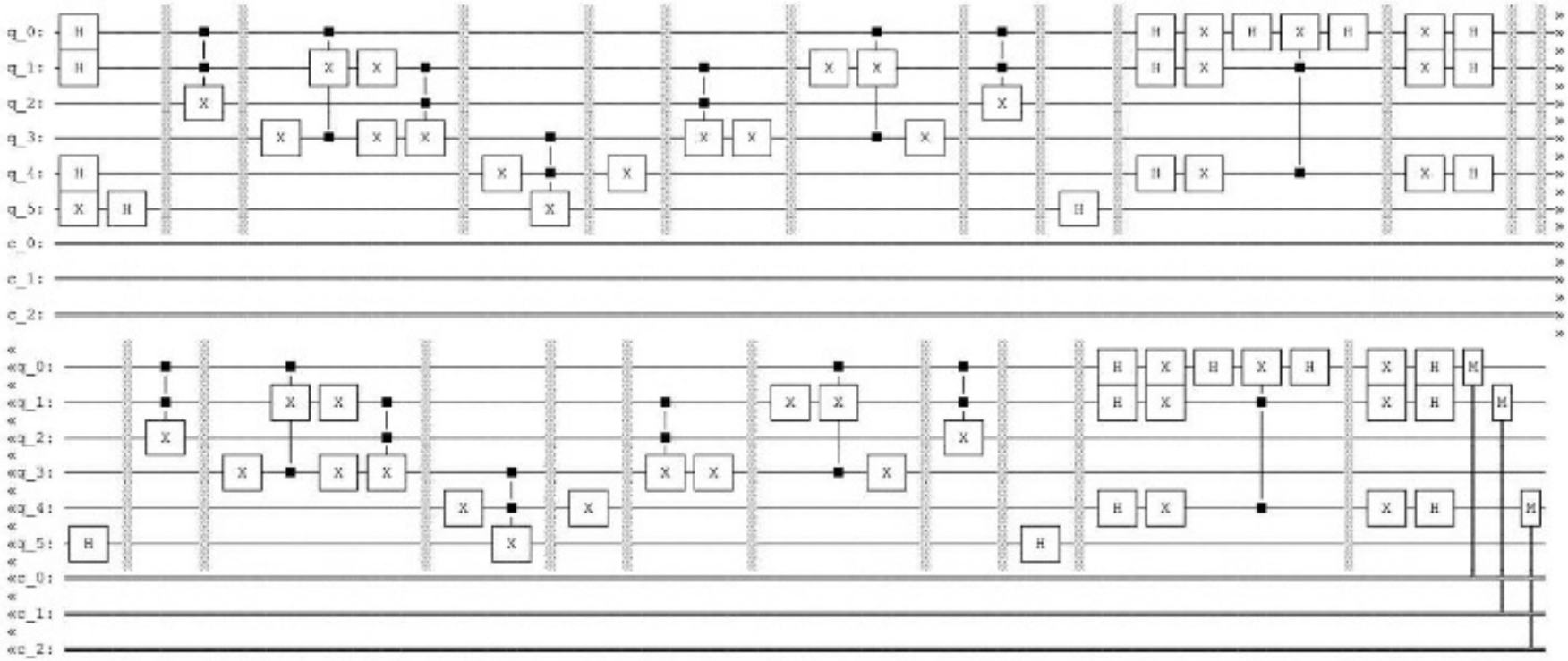
that evaluates true, which is the case for $x_1 = 1$, $x_2 = 1$ and $x_3 = 0$. In this case, the state determined by the oracle function with the solution encoded by $(-1)^{o(x)}$.

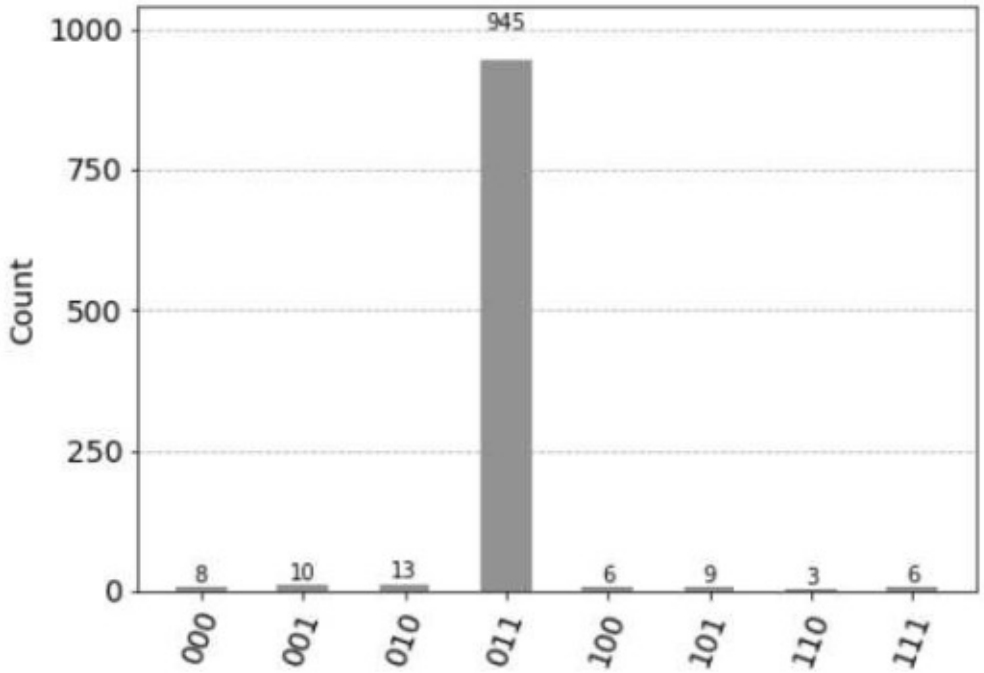


- The circuit representing the Boolean formula $\varphi = (x_1 \leftrightarrow x_2) \wedge (x_1 \wedge x_2) \wedge \neg x_3$.

- In quantum computation, it is not possible to reset the information to the pattern representing the initial state
- Instead, we un-compute the output back to the input
  - The input are the qubits 0, 1 and 4 mapped in superposition by Hadamard gates
  - The output is represented in the qubit 5 indicating by the value 1 the presence of the solution

Two Rotations

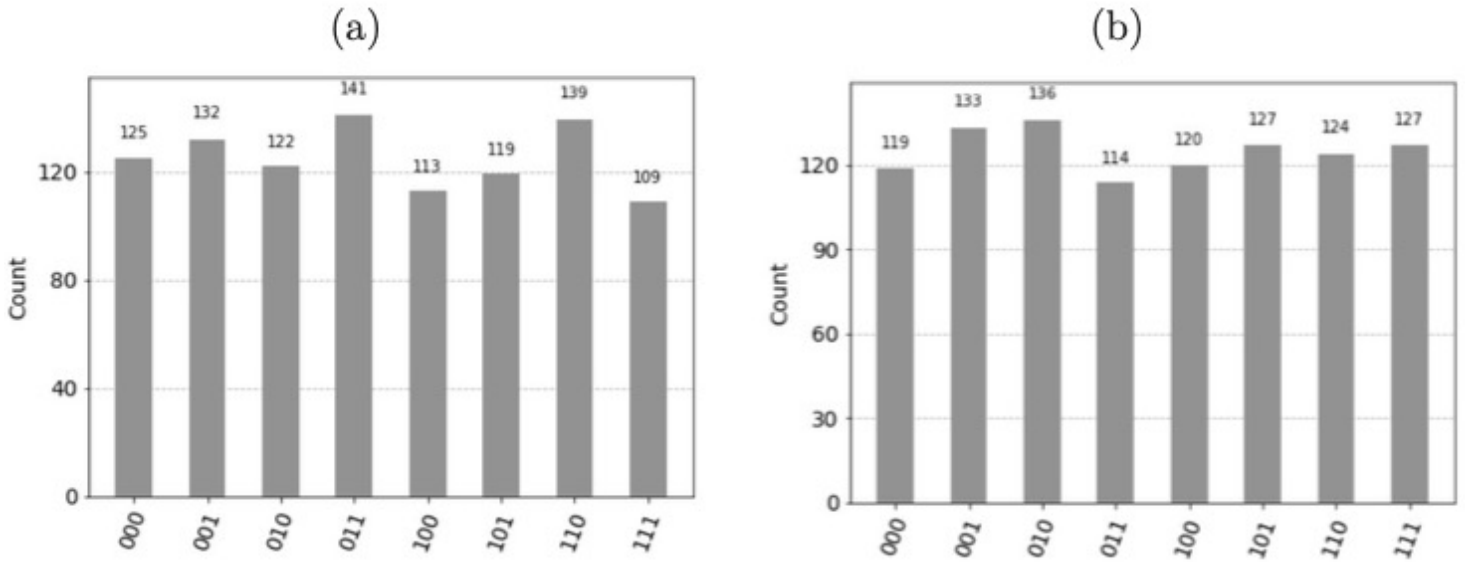




- The qubits 0, 1 and 4 indicate the eight possible instantiations for x_1, $x2$ and x_3 for the $\phi = (x_1 \leftrightarrow x_2) \wedge (x_1 \wedge x_2) \wedge \neg x_3$ SAT problem
- The solution after two rotations for the SAT problem has the highest count with the measured state $|011\rangle$

- For the Boolean formula
  - $\phi=(x_1 \leftrightarrow x_2) \wedge (\neg x_1 \wedge x_2) \wedge \neg x_3$
- no solution exists
  - the formula can never evaluate to the value true.

  - (a) one rotation, (b) two rotations



Grover's Amplitude Amplification

- Grovers Algorithm requires uniform distribution
- Why?
- $n=2^m$

$$|\psi\rangle = \frac{1}{\sqrt{n}} \cdot |x_1\rangle + \frac{1}{\sqrt{n}} \cdot |x_2\rangle + \cdots + \frac{1}{\sqrt{n}} \cdot |x_n\rangle = \begin{pmatrix} \frac{1}{\sqrt{n}} \\ \vdots \\ \frac{1}{\sqrt{n}} \end{pmatrix}$$

$$G_{\text{uniform}} = 2 \cdot |\psi\rangle\langle\psi| - I_m$$

$$G_m = H^{\otimes m}(2 \cdot |0^{\otimes m}\rangle\langle 0^{\otimes m}| - I_m)H^{\otimes m}$$

$$G_{\text{uniform}} = H^{\otimes m}(2|0\rangle\langle 0| - I)H^{\otimes m}$$

General Amplitude Amplification

Quantum Amplitude Amplification and Estimation; Gilles Brassard, Peter Hoyer, Michele Mosca, Alain Tapp, 2000, **https://arxiv.org/abs/quant-ph/0005055**

$$G_{\text{uniform}} = H^{\otimes m}(2|0\rangle\langle 0| - I)H^{\otimes m}$$

For non uniform distribution

$$A \neq H^{\otimes m}$$

Gilles Brassard

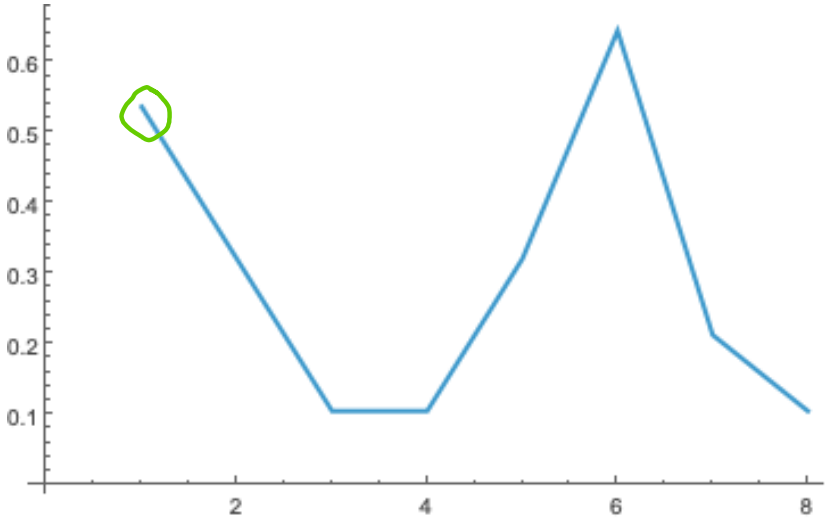
$$A|0\rangle = |\psi\rangle$$

$$D = A(2|0\rangle\langle 0| - I)A^{\dagger}$$

Example for non-uniform distribution

- $|\psi\rangle = \{\{\frac{5}{\sqrt{86}}, \frac{3}{\sqrt{86}}, \frac{1}{\sqrt{86}}, \frac{1}{\sqrt{86}}, \frac{3}{\sqrt{86}}, 3\sqrt{\frac{2}{43}}, \sqrt{\frac{2}{43}}, \frac{1}{\sqrt{86}}\}\}^T$

$$2 \cdot |\psi\rangle\langle\psi| - I_m$$



$$D = \begin{pmatrix} -\frac{18}{43} & \frac{15}{43} & \frac{5}{43} & \frac{5}{43} & \frac{15}{43} & \frac{30}{43} & \frac{10}{43} & \frac{5}{43} \\ \frac{15}{43} & -\frac{34}{43} & \frac{3}{43} & \frac{3}{43} & \frac{9}{43} & \frac{18}{43} & \frac{6}{43} & \frac{3}{43} \\ \frac{5}{43} & \frac{3}{43} & -\frac{42}{43} & \frac{1}{43} & \frac{3}{43} & \frac{6}{43} & \frac{2}{43} & \frac{1}{43} \\ \frac{5}{43} & \frac{3}{43} & \frac{1}{43} & -\frac{42}{43} & \frac{3}{43} & \frac{6}{43} & \frac{2}{43} & \frac{1}{43} \\ \frac{15}{43} & \frac{9}{43} & \frac{3}{43} & \frac{3}{43} & -\frac{34}{43} & \frac{18}{43} & \frac{6}{43} & \frac{3}{43} \\ \frac{30}{43} & \frac{18}{43} & \frac{6}{43} & \frac{6}{43} & \frac{18}{43} & -\frac{7}{43} & \frac{12}{43} & \frac{6}{43} \\ \frac{10}{43} & \frac{6}{43} & \frac{2}{43} & \frac{2}{43} & \frac{6}{43} & \frac{12}{43} & -\frac{39}{43} & \frac{2}{43} \\ \frac{5}{43} & \frac{3}{43} & \frac{1}{43} & \frac{1}{43} & \frac{3}{43} & \frac{6}{43} & \frac{2}{43} & -\frac{42}{43} \end{pmatrix}$$

$$|\psi\rangle = \{-\frac{5}{\sqrt{86}}, \frac{3}{\sqrt{86}}, \frac{1}{\sqrt{86}}, \frac{1}{\sqrt{86}}, \frac{3}{\sqrt{86}}, 3\sqrt{\frac{2}{43}}, \sqrt{\frac{2}{43}}, \frac{1}{\sqrt{86}}\}^T$$

$|\psi\rangle^{\text{one rotation}}$ ={0.990557, -0.0526625, -0.0175542, -0.0175542, -0.0526625, -0.105325, -0.0351083, -0.0175542}^{T}

$$|\psi\rangle = \sqrt{a}\,|\text{good}\rangle + \sqrt{1-a}\,|\text{bad}\rangle$$

$$\sin^2(\theta) = a \quad \Rightarrow \quad \theta = \arcsin(\sqrt{a})$$

$$\text{Success probability after } k \text{ steps} = \sin^2\left((2k+1)\theta\right)$$

Solving:

$$r \approx \frac{\pi}{4\theta} - \frac{1}{2}$$

Substitute $\theta = \arcsin(\sqrt{a})$:

$$r \approx \frac{\pi}{4\,\arcsin(\sqrt{a})} - \frac{1}{2}$$

If $a \ll 1$, then:

$$\arcsin(\sqrt{a}) \approx \sqrt{a}$$

So:

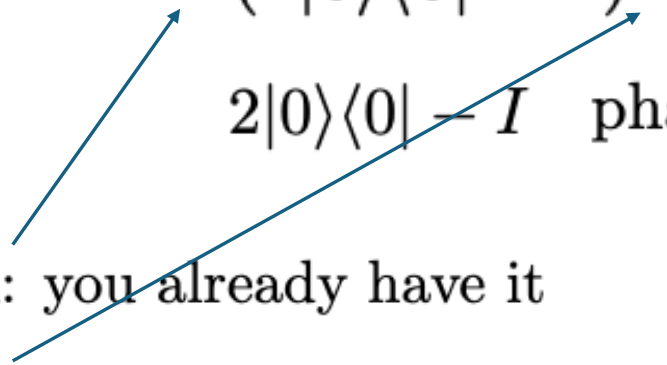
$$r \approx \frac{\pi}{4\sqrt{a}} \approx \frac{1}{\sqrt{a}}$$

How to Implement

- We **have to know** $A|0\rangle = |\psi\rangle$

$$D = A(2|0\rangle\langle 0| - I)A^\dagger$$

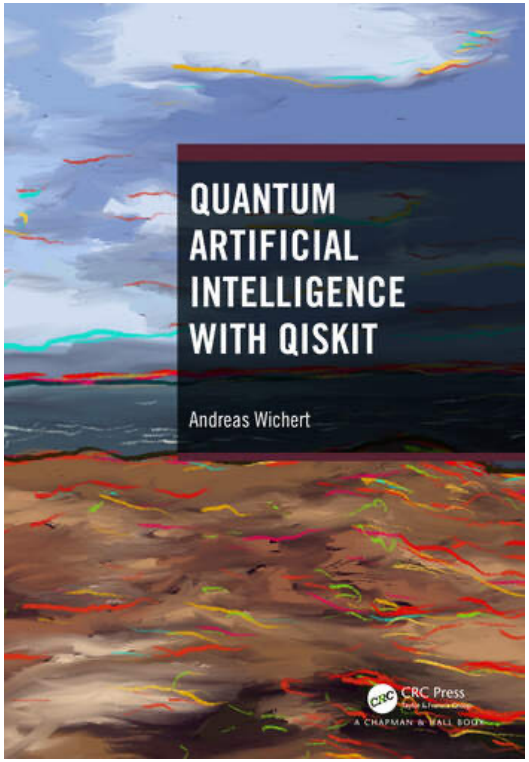
$2|0\rangle\langle 0| - I$ phase flip on $|0\rangle$



A: you already have it

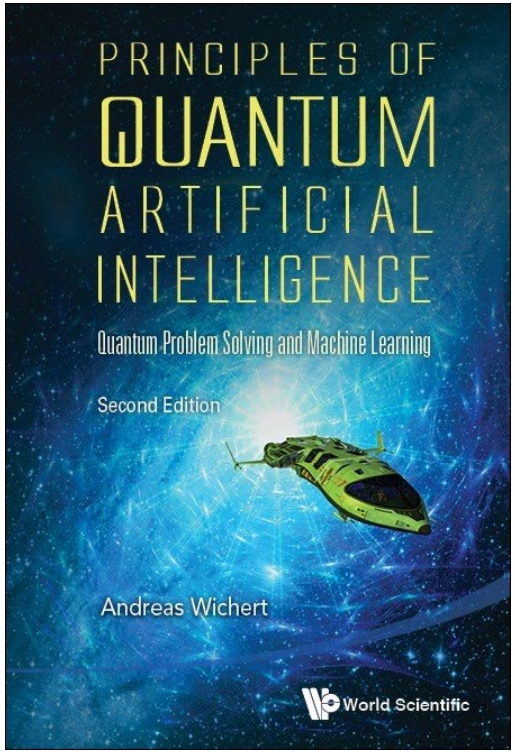
$A^\dagger$: just reverse the circuit

Reflection about your actual state implemented by "uncomputing" with $A^\dagger$



- Chapter 5
- Chapter 6

Quantum Artificial Intelligence with Qiskit, A. Wichert, Chapman and Hall/CRC, 2024



- Chapter 10

Principles of Quantum Artificial Intelligence: Quantum Problem Solving and Machine Learning, 2nd Edition, A. Wichert, World Scientific, 2020