# Obliczenia kwantowe

## 1. Wstęp

Kwantowa teoria informacji – przetwarzanie informacji korzystające z praw fizyki kwantowej, operowanie na pojedynczych układach kwantowych, atomach, fotonach

Kwantowa teoria informacji

Komunikacja kwantowa
- kryptografia kwantowa
- teleportacja...

Obliczenia kwantowe
- komputery kwantowe
- algorytmy kwantowe

Dotychczas mówiliśmy jedynie o komunikacji kwantowej
Czas zacząć mówić o obliczeniach.

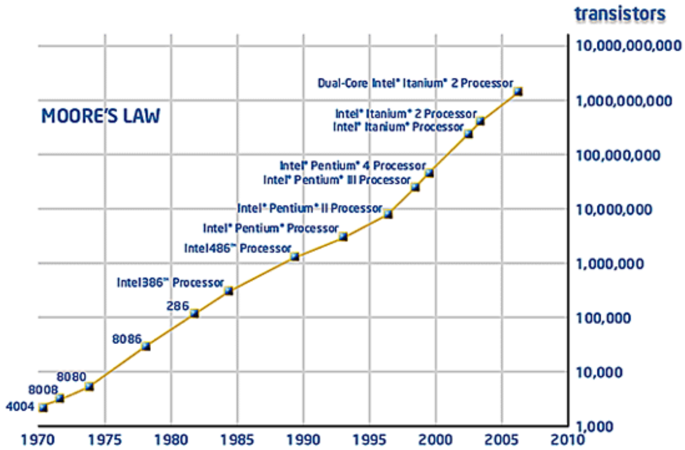
Obecne komputery też używają praw fizyki kwantowej:
struktura półprzewodników, tranzystory, momenty magnetyczne atomów (spiny)

Ale ....

– 1 bit danych na dysku twardym: (d ≈ 0,5nm)
$250nm \times 250nm \times 25nm \approx$ 12,5 mln atomów

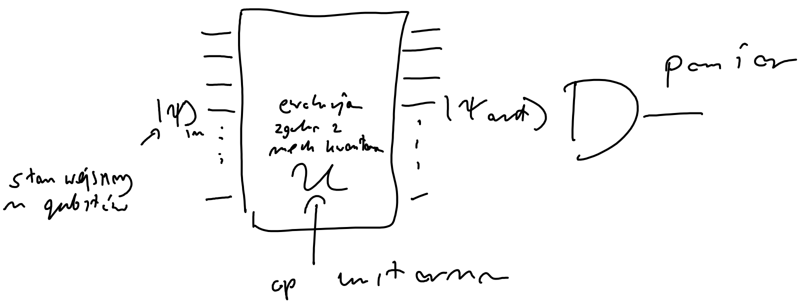
– 1 tranzystor w CPU
$50nm \times 50nm \times 25nm \approx$ 500 tys atomów

Widać dość dużo. Nie jesteśmy na etapie żeby używać pojedynczych atomów do obliczeń i wykorzystać pełne możliwości fizyki kwantowej. Kiedy zejdziemy do poziomu 1 atomu.




Rozmiar tranzystorów zmniejsza się dwa 2 razy co 2 lata
Będzie rozmiar atomu ok roku 2030-2050?
Nawet jak to nastąpi nie oznacza to, że mamy już komputer kwantowy.
Musimy umieć utrzymać kwantową superpozycję tak aby móc wykorzystać potencjał mech. kwantowej

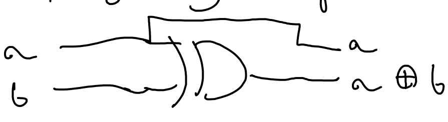
## 2. Idea




- Klasyczne komputery budujemy zwykle z prostych elementarnych bramek: NOT, AND, OR, XOR ..
  Wiadomo, że np. każdy układ logiczny można zbudować z bramek NAND.
- W klasycznych komputerach zwykle używamy bramek nieodwracalnych (chociaż można również inaczej)


nie da się odtworzyć wejścia z bitu wyjściowego

- Możemy używać w klasycznych obliczeniach bramek odwracalnych, wystarczy np.



Kosztem komplikacji obwodów. Zazwyczaj się tego nie robi. Pamiętajmy jednak, że fizyka w zasadzie jest odwracalna. Nieodwracalność bierze się z tego że po prostu ignorujemy jakieś stopnie swobody

- Myśląc o komputerach kwantowych mamy operacje unitarne - które są odwracalne. Można oczywiście też zrobić nieodwracalne np. śledząc po pomiarze z wyjściowych qubitów, ale to zazwyczaj niszczy kwantowe superpozycje, więc nie ma sensu. Ograniczmy się więc do operacji unitarnych.

Chcemy znaleźć elementarne bramki z których można złożyć dowolny op. $U$. Bramki też muszą być odwracalne.

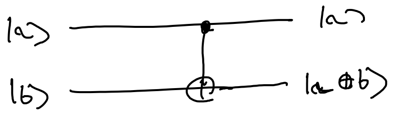
- Bramka CNOT

$|0\rangle|0\rangle \longrightarrow |0\rangle|0\rangle$

$|0\rangle|1\rangle \longrightarrow |0\rangle|1\rangle$

$|1\rangle|0\rangle \longrightarrow |1\rangle|1\rangle$

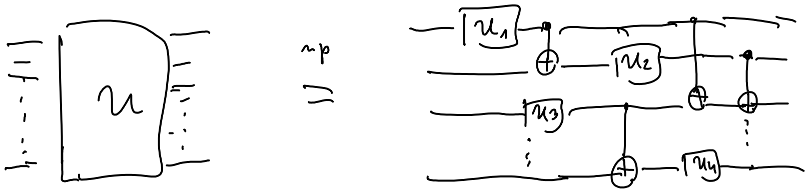
$|1\rangle|1\rangle \longrightarrow |1\rangle|0\rangle$




$$U_{CNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

w bazie $|0\rangle|0\rangle, |0\rangle|1\rangle, |1\rangle|0\rangle, |1\rangle|1\rangle$.
Myśląc klasycznie to jest taki odwracalny XOR

<u>Fakt</u> Każda wieloqubitowa $U$ może być rozłożona na jednoqubitowe op. unitarne i bramki CNOT




W ogólności potrzebujemy wielu bramek jednoqubitowych (np. wielu obrotów sfery Blocha) można np. wybrać: ..

—[H]— bramka Hadamarda $H = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$ } $H^\dagger H = 1$

—[$\varphi$]— opóźnienie fazowe $\varphi$ (bramka fazowa) $U_\varphi = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\varphi} \end{bmatrix}$ (np. $\varphi = \frac{\pi}{2}$ ćwierćfalówka)

3. Kwantowy Parallelizm – dlaczego komputer kwantowy ma szansę liczyć szybciej?

Idea: $f: \{0,1\} \to \{0,1\}$ jedno bitowa funkcja $f(0), f(1)$

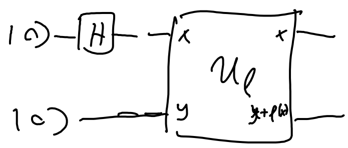
Wyobraźmy sobie że każdej funkcji $f$ w bramce kwantowej $U_f$

$$|x\rangle|y\rangle \xrightarrow{U_f} |x\rangle|y \oplus f(x)\rangle$$

$|x\rangle$ – qubit z argumentem funkcji; $|y\rangle$ – qubit z wynikiem funkcji; $\oplus$ – dodawanie mod 2

$$\left\{\begin{array}{l} |0\rangle|0\rangle \\ |0\rangle|1\rangle \\ |1\rangle|0\rangle \\ |1\rangle|1\rangle \end{array}\right. \xrightarrow{U_f} \begin{array}{l} |0\rangle|0\oplus f(0)\rangle \\ |0\rangle|1\oplus f(0)\rangle \\ |1\rangle|0\oplus f(1)\rangle \\ |1\rangle|1\oplus f(1)\rangle \end{array}$$

jest to op. unitarna



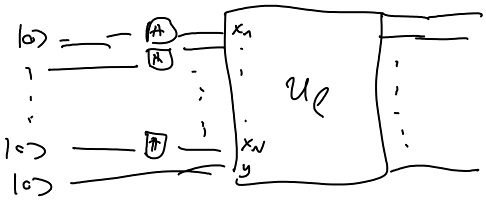

$$|\psi\rangle = \frac{1}{\sqrt2}\left(|0\rangle|f(0)\rangle + |1\rangle|f(1)\rangle\right)$$

$$|0\rangle|0\rangle \xrightarrow{H\otimes 1} \frac{1}{\sqrt2}(|0\rangle+|1\rangle)|0\rangle = \frac{1}{\sqrt2}(|0\rangle|0\rangle+|1\rangle|0\rangle) \xrightarrow{U_f} \frac{1}{\sqrt2}\cdot(|0\rangle|f(0)\rangle + |1\rangle|f(1)\rangle)$$

Użyliśmy tylko raz obliczenia $f$ a mamy stan $|\psi\rangle$ w którym pokazane jest $f$ zarówno dla $f(0)$ i $f(1)$.
„Liczy równolegle” $f(0)$ i $f(1)$ dzięki temu, że wykorzystaliśmy superpozycję.

Ogólniej: $f: \{0,1\}^N \to \{0,1\}$ funkcja na $N$ bitach

$$|x_1, \dots, x_N, y\rangle \xrightarrow{U_f} |x_1 \dots x_N, y \oplus f(x_1,\dots,x_N)\rangle$$




$$|0\rangle^{\otimes N}|0\rangle \xrightarrow{H^{\otimes N}\otimes 1} \left(\frac{1}{\sqrt2}(|0\rangle+|1\rangle)\right)^{\otimes N} \otimes |0\rangle =$$

$$= \frac{1}{\sqrt{2^N}}\Big(\underbrace{|0\rangle\cdots|0\rangle + |0\rangle\cdots|1\rangle + \dots + |1\rangle\cdots|1\rangle}_{\text{wszystkie liczby } N \text{ bitowe}}\Big)\otimes|0\rangle \xrightarrow{U_f}$$

$$= \frac{1}{\sqrt{2^N}}\cdot\left(|0\rangle\cdots|0\rangle\otimes|f(0,\dots,0)\rangle + |0\rangle\cdots|1\rangle\otimes|f(0,\dots,1)\rangle + \dots + |1\rangle\cdots|1\rangle\otimes|f(1,\dots,1)\rangle\right)$$

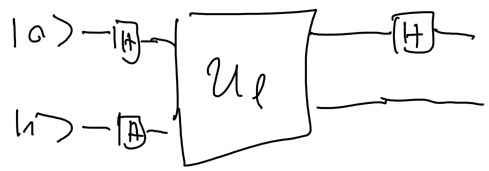
Policzyliśmy „w jednym obliczeniu wartości funkcji $f$ dla

„Policzyliśmy" w jednym obliczeniu wartości funkcji $f$ dla wszystkich $2^N$ możliwych danych wejściowych.
Nadzieja na wykładnicze przyspieszenie obliczeń !
Ale nie tak szybko – nie istnieje pomiar pozwalający jednocześnie odczytać wszystkie wartości $f$. Mierząc w bazie $|0\ldots0\rangle, \ldots, |1\ldots1\rangle$. Stan zredukuje się nam na jeden z wektorów superpozycji i poznamy tylko jedną wartość $f$....

Ale może są problemy w których takie obliczenie wszystkich $f$ jest elementem pośrednim a na końcu odczytujemy jakąś funkcję tych $f$ i wystarczy jeden pomiar dający pożądany wynik.

## Algorytm Deutscha

Najprostszy: całkowicie nieprzydatny ale ciekawy dydaktycznie.
Rozważmy funkcję $f: \{0,1\} \to \{0,1\}$.
Pytamy się czy funkcja jest różnowartościowa?
Klasycznie funkcję musimy policzyć 2 razy.
A jeśli mamy komputer kwantowy wystarczy raz.




$$U_f|x, y\rangle = |x, y \oplus f(x)\rangle$$

$$\frac{1}{\sqrt{2}}(|0\rangle+|1\rangle)\frac{1}{\sqrt{2}}(|0\rangle-|1\rangle) \xrightarrow{U_f} \frac{1}{2}\Big(|0\rangle\big(|f(0)\rangle - |1\oplus f(0)\rangle\big) + |1\rangle\big(|f(1)\rangle - |1\oplus f(1)\rangle\big)\Big) =$$

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

$$= \frac{1}{2}\Big((-1)^{f(0)}|0\rangle + (-1)^{f(1)}|1\rangle\Big)\big(|0\rangle-|1\rangle\big) \xrightarrow{H\otimes I}$$

$$= \frac{1}{2\sqrt{2}}\Big((-1)^{f(0)}(|0\rangle+|1\rangle) + (-1)^{f(1)}(|0\rangle-|1\rangle)\Big) \otimes (|0\rangle-|1\rangle) =$$

$$= \Big(\underbrace{\tfrac{1}{2}\big[(-1)^{f(0)}+(-1)^{f(1)}\big]}_{\substack{\| \\ 1 - f \text{ stała} \\ 0 - f \text{ różnowart}}}|0\rangle + \tfrac{1}{2}\underbrace{\big[(-1)^{f(0)} - (-1)^{f(1)}\big]}_{\substack{\| \\ 1 - f \text{ różnowart.} \\ 0 - f \text{ stała}}}|1\rangle\Big) \otimes \frac{1}{\sqrt{2}}(|0\rangle-|1\rangle)$$

0 - f. zrównoważ.  0 - f stała

Mierzymy pierwszy qubit jeśli wynik $|0\rangle \Rightarrow$ f. stała
$|1\rangle \Rightarrow$ f zrównoważona

Dzięki temu że policzyliśmy jednocześnie dla obu wartości wejściowych wystarczyło tylko raz użyć pudełka.

# Algorytm Grovera

Kwantowe przeszukiwanie nieuporządkowanej bazy danych. Niech baza składa się z $N$ $(N=2^n)$ elementów.

Niech $f$ będzie funkcją identyfikującą poszukiwany element:

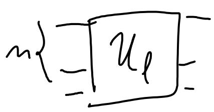
$f(x)=1$ gdy $x$ jest poszukiwanym elementem

$f(x)=0$ w przeciwnym razie

Jeśli szukamy jednego elementu, musimy średnio obliczyć funkcję $f$, $\sim \frac{N}{2}$ razy. Czy kwantowo da się lepiej?

Załóżmy że mamy kwantową wersję $f$

$$U_f \underbrace{|x\rangle}_{n \text{ qubitów}} \underbrace{|y\rangle}_{1 \text{ qubit}} = |x\rangle|y \oplus f(x)\rangle$$



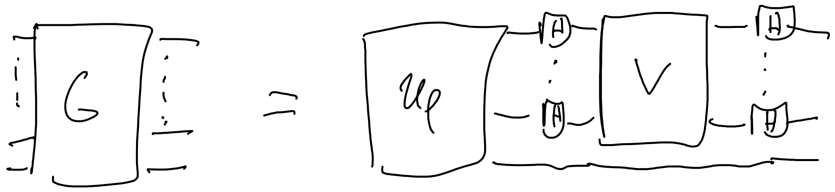
Zauważmy że:

$$U_f |x\rangle \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = (-1)^{f(x)} \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$$

W dalszym ciągu możemy ignorować ostatni qubit (on cały czas będzie pozostawał w stanie $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$)

Schemat: k razy


jedna iteracja w alg. Grovera




V – op. unitarny t. że $V|0\rangle = |0\rangle$
$V|x\rangle = -|x\rangle$, $x > 0$

$V = 2|0\rangle\langle 0| - I$

$H^{\otimes N} V H^{\otimes N} = 2|\psi\rangle\langle\psi| - I$, gdzie

$|\psi\rangle = \frac{1}{\sqrt{N}} \sum_x |x\rangle$

Czyli: $G = \left(2|\psi\rangle\langle\psi| - I\right) U_f$

Niech $|x_0\rangle$ – będzie odpowiedzią szukanym stanem

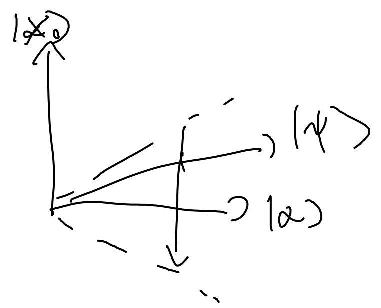
Niech $|\alpha\rangle = \frac{1}{\sqrt{N-1}} \sum_{x \neq x_0} |x\rangle$ będzie superpozycją pozostałych stanów.

Rozważmy jak działa G w superpozycji $|x_0\rangle$ i $|\alpha\rangle$

Zauważmy że $|\psi\rangle = \sqrt{\frac{N-1}{N}}|\alpha\rangle + \frac{1}{\sqrt{N}}|x_0\rangle$
podprzestrzeni.

$$G\left(a|\alpha\rangle + b|x_0\rangle\right) = \left(2|\psi\rangle\langle\psi| - I\right)\left(a|\alpha\rangle - b|x_0\rangle\right)$$

odbicie względem kierunku $|\alpha\rangle$




$$= 2|\psi\rangle\left(\sqrt{\frac{N-1}{N}}\cdot a - \frac{1}{\sqrt{N}} b\right) - \left(a|\alpha\rangle - b|x_0\rangle\right) =$$

odbicie względem $|\psi\rangle$

$$= 2\,\frac{N-1}{N}\,a|\alpha\rangle - 2\,\frac{1}{N}\,b|x_0\rangle + \frac{2\sqrt{N-1}}{N}\,a|x_0\rangle - \frac{2\sqrt{N-1}}{N}\,b|\alpha\rangle$$

$$- \left(a|\alpha\rangle - b|x_0\rangle\right) =$$

$$= \left(2a - \frac{2a}{N} - \frac{2\sqrt{N-1}}{N} b - a\right)|\alpha\rangle + \left(b - \frac{2}{N} b + \frac{2\sqrt{N-1}}{N} a\right)|x_0\rangle$$

$$= \left(a\left(1-\frac{2}{N}\right) - b\,\frac{2\sqrt{N-1}}{N}\right)|\alpha\rangle + \left(b\left(1-\frac{2}{N}\right) + a\,\frac{2\sqrt{N-1}}{N}\right)|x_0\rangle$$

Czyli po prostu obrót o kąt $\theta$:

$$\cos\theta = 1 - \frac{2}{N} \qquad \theta = \arccos\left(1 - \frac{2}{N}\right)$$

W Alg. Grovera startujemy ze stanu $|\psi\rangle$

$$|\psi\rangle = \sqrt{\frac{N-1}{N}}|\alpha\rangle + \frac{1}{\sqrt{N}}|x_0\rangle =$$

$$= \cos\frac{\theta}{2}|\alpha\rangle + \sin\frac{\theta}{2}|x_0\rangle$$

I w każdym kroku obracamy się o kąt $\theta$

Czyli po k iteracjach:

$$G^k|\psi\rangle = \cos\left(\frac{2k+1}{2}\theta\right)|\alpha\rangle + \sin\left(\frac{2k+1}{2}\theta\right)|x_0\rangle$$

Jeśli $N$ b. duże $\quad \theta \approx \frac{2\sqrt{N-1}}{N} \approx \frac{2}{\sqrt{N}}$

Jeśli $N$ b. duże $\quad \Theta \approx \frac{2\sqrt{N-1}}{N} \approx \frac{2}{\sqrt{N}}$

Chcemy, aby $\frac{2k+1}{2}\Theta \approx \frac{\pi}{2}$

$$(2k+1)\cdot\frac{2}{\sqrt{N}} = \pi \qquad k \approx \sqrt{N}$$

Czyli kwadratowe przyspieszenie w porównaniu z algorytmem klasycznym.