

Kvantové procházky a kvantové vyhledávací algoritmy

Stanislav Skoupý

České vysoké učení technické v Praze

21. srpna 2015

- 1 Obsah
- 2 Kvantová procházka
 - Klasická náhodná procházka
 - Přechod od klasické procházky ke kvantové procházce
- 3 Vyhledávací algoritmy
 - Groverův vyhledávací algoritmus
 - Vyhledávací algoritmu založený na kvantové procházce



- Klasická procházka je náhodný proces, při kterém se chodec pohybuje po určitém typu grafu.

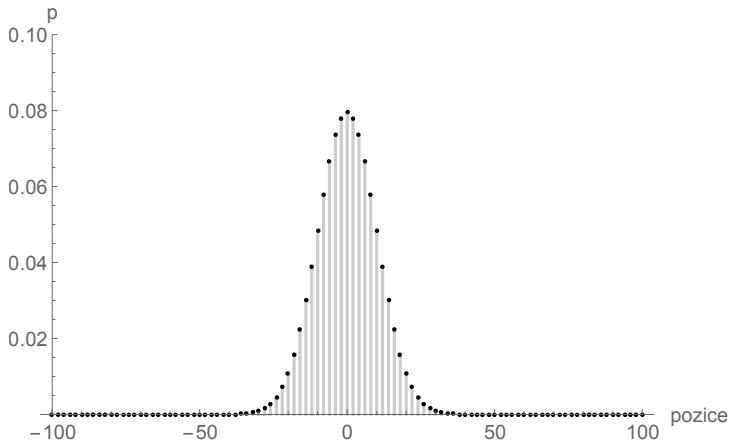


- Klasická procházka je náhodný proces, při kterém se chodec pohybuje po určitém typu grafu.
- V jednorozměrném případě se distribuce pravděpodobnosti blíží s rostoucím počtem kroků k *normálnímu rozdělení*.



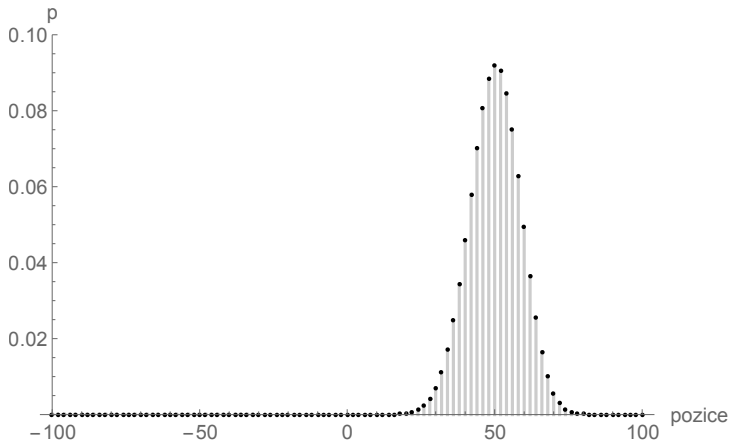
- Klasická procházka je náhodný proces, při kterém se chodec pohybuje po určitém typu grafu.
- V jednorozměrném případě se distribuce pravděpodobnosti blíží s rostoucím počtem kroků k *normálnímu rozdělení*.
- *Vyvážená procházka* = pravděpodobnost kroku doleva i doprava je rovna $\frac{1}{2}$





Obrázek: Rozdělení pravděpodobnosti pro vyváženou procházku po 100 krocích.





Obrázek: Rozdělení pravděpodobnosti pro procházku s pravděpodobností $\frac{3}{4}$ pro krok doprava po 100 krocích.



- Diskrétní přímce přiřadíme Hilbertův prostor \mathcal{H}_p s ortonormální bází $\{|i\rangle : i \in \mathbb{Z}\}$.



- Diskrétní přímce přiřadíme Hilbertův prostor \mathcal{H}_p s ortonormální bází $\{|i\rangle : i \in \mathbb{Z}\}$.
- Primitivní přístup nefunguje, není unitární

$$|0\rangle \rightarrow \frac{1}{\sqrt{2}}|-1\rangle + \frac{1}{\sqrt{2}}|1\rangle \rightarrow \frac{1}{2}|-2\rangle + |0\rangle + \frac{1}{2}|2\rangle$$



- Diskrétní přímce přiřadíme Hilbertův prostor \mathcal{H}_p s ortonormální bází $\{|i\rangle : i \in \mathbb{Z}\}$.
- Primitivní přístup nefunguje, není unitární

$$|0\rangle \rightarrow \frac{1}{\sqrt{2}}|-1\rangle + \frac{1}{\sqrt{2}}|1\rangle \rightarrow \frac{1}{2}|-2\rangle + |0\rangle + \frac{1}{2}|2\rangle$$

- Přidáme další stupeň volnosti, který udává směr, \mathcal{H}_c s bází $\{|\leftarrow\rangle, |\rightarrow\rangle\}$. Výsledný prostor má tvar $\mathcal{H} = \mathcal{H}_c \otimes \mathcal{H}_p$.



- Na tomto prostoru zavedeme posunovací operátor \hat{S} , který bude provádět krok procházky, následovně

$$\hat{S} = |\rightarrow\rangle\langle\rightarrow| \otimes \sum_i |i+1\rangle\langle i| + |\leftarrow\rangle\langle\leftarrow| \otimes \sum_i |i-1\rangle\langle i|$$



- Na tomto prostoru zavedeme posunovací operátor \hat{S} , který bude provádět krok procházky, následovně

$$\hat{S} = |\rightarrow\rangle\langle\rightarrow| \otimes \sum_i |i+1\rangle\langle i| + |\leftarrow\rangle\langle\leftarrow| \otimes \sum_i |i-1\rangle\langle i|$$

- Pro netriviální procházku přidáme takzvaný hod mincí, tj. aplikace operátoru na prostor směrů.



- Na tomto prostoru zavedeme posunovací operátor \hat{S} , který bude provádět krok procházky, následovně

$$\hat{S} = |\rightarrow\rangle\langle\rightarrow| \otimes \sum_i |i+1\rangle\langle i| + |\leftarrow\rangle\langle\leftarrow| \otimes \sum_i |i-1\rangle\langle i|$$

- Pro netriviální procházku přidáme takzvaný hod mincí, tj. aplikace operátoru na prostor směrů.
- Jedním z často používaných operátorů je Hadamardova mince

$$\hat{C} = \frac{1}{\sqrt{2}} (|\rightarrow\rangle + |\leftarrow\rangle) \langle\rightarrow| + \frac{1}{\sqrt{2}} (|\rightarrow\rangle - |\leftarrow\rangle) \langle\leftarrow|$$



- Výsledný krok procházky má tvar $\hat{U} = \hat{S} (\hat{C} \otimes \hat{I})$



- Výsledný krok procházky má tvar $\hat{U} = \hat{S} (\hat{C} \otimes \hat{I})$
- Pokud bychom po každém kroku (aplikaci \hat{U}) provedli měření na prostoru směrů dostali bychom rozdělení pravděpodobnosti pro klasickou procházku.



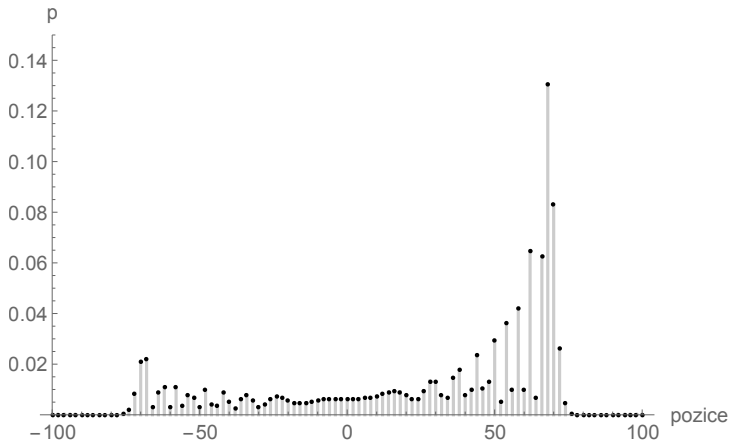
- Výsledný krok procházky má tvar $\hat{U} = \hat{S} (\hat{C} \otimes \hat{I})$
- Pokud bychom po každém kroku (aplikaci \hat{U}) provedli měření na prostoru směrů dostali bychom rozdělení pravděpodobnosti pro klasickou procházku.
- Pokud ale měření neprovádíme, začne se objevovat interference a procházka se od té klasické začne lišit už od 3. kroku.



Začneme se stavem $|\rightarrow\rangle \otimes |0\rangle$ a aplikujeme třikrát \hat{U}

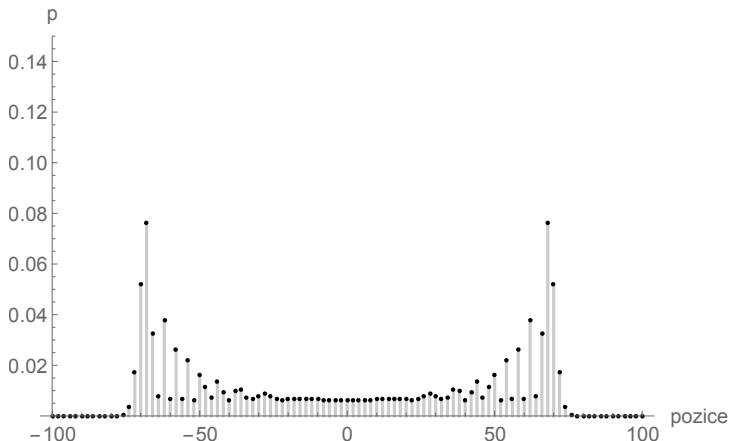
$$\begin{aligned}\hat{U}^3 (|\rightarrow\rangle \otimes |0\rangle) &= \hat{U}^2 \left(\frac{1}{\sqrt{2}} |\rightarrow\rangle \otimes |1\rangle + \frac{1}{\sqrt{2}} |\leftarrow\rangle \otimes |-1\rangle \right) = \\ &= \hat{U} \left(\frac{1}{2} |\rightarrow\rangle \otimes |2\rangle + \frac{1}{2} (|\rightarrow\rangle + |\leftarrow\rangle) \otimes |0\rangle - \frac{1}{2} |\leftarrow\rangle \otimes |-2\rangle \right) = \\ &= \frac{1}{2\sqrt{2}} |\rightarrow\rangle \otimes |3\rangle + \frac{1}{2\sqrt{2}} (2|\rightarrow\rangle + |\leftarrow\rangle) \otimes |1\rangle \\ &\quad - \frac{1}{2\sqrt{2}} |\rightarrow\rangle \otimes |-1\rangle + \frac{1}{2\sqrt{2}} |\leftarrow\rangle \otimes |-3\rangle\end{aligned}$$





Obrázek: Rozdělení pravděpodobnosti pro procházku s počátečním stavem $|\rightarrow\rangle \otimes |0\rangle$ po 100 krocích.





Obrázek: Rozdělení pravděpodobnosti pro procházku s počátečním stavem $\frac{1}{\sqrt{2}} (|\rightarrow\rangle + i|\leftarrow\rangle) \otimes |0\rangle$ po 100 krocích.



- Diskrétní kvantovou procházka je možné zobecnit na d -regulární grafy, což jsou grafy, kde z každého vrcholu vychází d hran.



- Diskrétní kvantovou procházka je možné zobecnit na d -regulární grafy, což jsou grafy, kde z každého vrcholu vychází d hran.
- Báze příslušného Hilbertova prostoru je $\{|i\rangle \otimes |v\rangle : v \in G, i \in 1 \dots d\}$.
Posunovací operátor má tvar $\hat{S}|i\rangle \otimes |v\rangle = |i\rangle \otimes |w\rangle$, kde vrchol v a w jsou spojeny hranou označenou i .



- Diskrétní kvantovou procházka je možné zobecnit na d -regulární grafy, což jsou grafy, kde z každého vrcholu vychází d hran.
- Báze příslušného Hilbertova prostoru je $\{|i\rangle \otimes |v\rangle : v \in G, i \in 1 \dots d\}$.
Posunovací operátor má tvar $\hat{S}|i\rangle \otimes |v\rangle = |i\rangle \otimes |w\rangle$, kde vrchol v a w jsou spojeny hranou označenou i .
- Operátor kroku je $\hat{U} = \hat{S} \cdot \sum_{v \in G} \hat{C}_v \otimes |v\rangle\langle v|$, protože obecně může být v každém vrcholu jiná mince.



- V kvantovém počítání se používá pojem *qubit*, což je kvantový ekvivalent klasického bitu.



- V kvantovém počítání se používá pojem *qubit*, což je kvantový ekvivalent klasického bitu.
- Hilbertův prostor jednoho qubitu má bázi $\{|0\rangle, |1\rangle\}$. Tento prostor je izomorfní \mathbb{C}^2 .



- V kvantovém počítání se používá pojem *qubit*, což je kvantový ekvivalent klasického bitu.
- Hilbertův prostor jednoho qubitu má bázi $\{|0\rangle, |1\rangle\}$. Tento prostor je izomorfní \mathbb{C}^2 .
- Hilbertův prostor kvantového registru dostaneme jako tenzorový součin prostorů odpovídajícím jednotlivým qubitům. Pro zjednodušení značení budeme používat $|i\rangle \otimes |j\rangle = |ij\rangle$.



Groverův algoritmus

- Kvantový algoritmus určený pro vyhledávání prvku v neseřazené databázi.



Groverův algoritmus

- Kvantový algoritmus určený pro vyhledávání prvku v neseřazené databázi.
- Oproti klasickému algoritmu se složitostí $O\left(\frac{N}{2}\right)$ má Groverův algoritmus složitost $O\left(\sqrt{N}\right)$



- Předpokládejme, že máme kvantový registr s n qubity. Hilbertův prostor tohoto registru má bázi $\{|i_1 i_2 \dots i_n\rangle; i_k \in \{0, 1\}\}$, která má $N = 2^n$ prvků.



- Předpokládejme, že máme kvantový registr s n qubity. Hilbertův prostor tohoto registru má bázi $\{|i_1 i_2 \dots i_n\rangle; i_k \in \{0, 1\}\}$, která má $N = 2^n$ prvků.
- Prvky této báze přeznačíme $|1\rangle, |2\rangle, \dots, |N\rangle$.



- Předpokládejme, že máme kvantový registr s n qubity. Hilbertův prostor tohoto registru má bázi $\{|i_1 i_2 \dots i_n\rangle; i_k \in \{0, 1\}\}$, která má $N = 2^n$ prvků.
- Prvky této báze přeznačíme $|1\rangle, |2\rangle, \dots, |N\rangle$.
- Necht' dále existuje unikátní stav, který označíme $|m\rangle$ a který splňuje podmínku $P(|m\rangle) = 1$ a ostatní stavy splňují $P(|i\rangle) = 0$ pro $i \neq m$.



1. Uvedeme registr do stavu

$$|\varphi_0\rangle = \sum_{i=1}^N \frac{1}{\sqrt{N}} |i\rangle.$$



2. Opakujeme následující operace $O(\sqrt{N})$ -krát

- Rotujeme fázi $|i\rangle$ o π pokud $P(|i\rangle) = 1$. Pokud $P(|i\rangle) = 0$, ponecháme fázi nezměněnou. To lze provést aplikací operátoru $\hat{I} - 2|m\rangle\langle m|$.



2. Opakujeme následující operace $O(\sqrt{N})$ -krát

- Rotujeme fázi $|i\rangle$ o π pokud $P(|i\rangle) = 1$. Pokud $P(|i\rangle) = 0$, ponecháme fázi nezměněnou. To lze provést aplikací operátoru $\hat{I} - 2|m\rangle\langle m|$.
- Aplikujeme difuzní operátor \hat{D} , který je definován pomocí matice $D = (D_{ij})$ takto:

$$\langle i|\hat{D}|j\rangle = D_{ij} = \frac{2}{N} \text{ pokud } i \neq j \text{ a } \langle i|\hat{D}|i\rangle = D_{ii} = -1 + \frac{2}{N}$$



3. Změříme výsledný stav. S pravděpodobností větší než 0,5 naměříme stav $|m\rangle$.



Ukážeme, že Groverův algoritmus funguje

- Zavedeme vektor $|\varphi_1\rangle = \frac{\sqrt{N}}{\sqrt{N-1}}|m\rangle - \frac{1}{\sqrt{N-1}}|\varphi_0\rangle$



Ukážeme, že Groverův algoritmus funguje

- Zavedeme vektor $|\varphi_1\rangle = \frac{\sqrt{N}}{\sqrt{N-1}}|m\rangle - \frac{1}{\sqrt{N-1}}|\varphi_0\rangle$
- Potom lze ukázat, že vektory $|\omega_{\pm}\rangle = \frac{1}{\sqrt{2}}(|\varphi_0\rangle \pm i|\varphi_1\rangle)$ jsou vlastními vektory operátoru $\hat{D} \left(\hat{I} - 2|m\rangle\langle m| \right)$ s vlastními čísly $e^{\pm i\omega}$, kde pro ω platí vztahy $\sin(\omega) = \frac{2\sqrt{N-1}}{N}$ a $\cos(\omega) = 1 - \frac{2}{N}$.



- Vektory $|\varphi_0\rangle$ a $|\varphi_1\rangle$ se dají vyjádřit pomocí vektorů $|\omega_{\pm}\rangle$

$$|\varphi_0\rangle = \frac{1}{\sqrt{2}} (|\omega_+\rangle + |\omega_-\rangle), |\varphi_1\rangle = \frac{-i}{\sqrt{2}} (|\omega_+\rangle - |\omega_-\rangle)$$



- Vektory $|\varphi_0\rangle$ a $|\varphi_1\rangle$ se dají vyjádřit pomocí vektorů $|\omega_{\pm}\rangle$

$$|\varphi_0\rangle = \frac{1}{\sqrt{2}} (|\omega_+\rangle + |\omega_-\rangle), |\varphi_1\rangle = \frac{-i}{\sqrt{2}} (|\omega_+\rangle - |\omega_-\rangle)$$

- Aplikujeme t -krát operátor $\hat{D} \left(\hat{I} - 2|m\rangle\langle m| \right)$ na počáteční stav $|\varphi_0\rangle$

$$\left[\hat{D} \left(\hat{I} - 2|S_m\rangle\langle S_m| \right) \right]^t |\varphi_0\rangle = \frac{1}{\sqrt{2}} (e^{i\omega t} |\omega_+\rangle + e^{-i\omega t} |\omega_-\rangle)$$



- Pokud zvolíme t tak, aby $t\omega = \frac{\pi}{2}$, potom dostaneme vektor $|\varphi_1\rangle$.



- Pokud zvolíme t tak, aby $t\omega = \frac{\pi}{2}$, potom dostaneme vektor $|\varphi_1\rangle$.
- Protože platí $\omega \sim \frac{1}{\sqrt{N}}$, dostaneme odhad kroků $t = O(\sqrt{N})$.



- Pokud zvolíme t tak, aby $t\omega = \frac{\pi}{2}$, potom dostaneme vektor $|\varphi_1\rangle$.
- Protože platí $\omega \sim \frac{1}{\sqrt{N}}$, dostaneme odhad kroků $t = O(\sqrt{N})$.
- Lze ukázat, že kvantové vyhledávací algoritmy potřebují nejméně $\Omega(\sqrt{N})$ kroků.

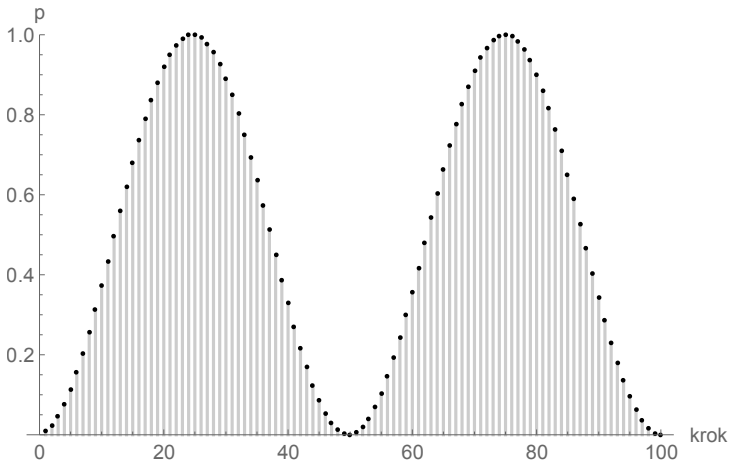


- Pokud zvolíme t tak, aby $t\omega = \frac{\pi}{2}$, potom dostaneme vektor $|\varphi_1\rangle$.
- Protože platí $\omega \sim \frac{1}{\sqrt{N}}$, dostaneme odhad kroků $t = O(\sqrt{N})$.
- Lze ukázat, že kvantové vyhledávací algoritmy potřebují nejméně $\Omega(\sqrt{N})$ kroků.
- To znamená, že Groverův vyhledávací algoritmu je optimální.



- Pokud zvolíme t tak, aby $t\omega = \frac{\pi}{2}$, potom dostaneme vektor $|\varphi_1\rangle$.
- Protože platí $\omega \sim \frac{1}{\sqrt{N}}$, dostaneme odhad kroků $t = O(\sqrt{N})$.
- Lze ukázat, že kvantové vyhledávací algoritmy potřebují nejméně $\Omega(\sqrt{N})$ kroků.
- To znamená, že Groverův vyhledávací algoritmus je optimální.
- Více informací o tomto algoritmu můžete najít v článku L. Grover: *A fast quantum mechanical algorithm for database search* (1996), arXiv:quant-ph/9605043





Obrázek: Vývoj pravděpodobnosti naměření označeného stavu pro 1024 položek.



Vyhledávací algoritmu založený na kvantové procházce pro hledání na hyperkrychli

- Hyperkrychle je krychle v n -dimenzionálním prostoru s jednotkovou hranou.



Vyhledávací algoritmu založený na kvantové procházce pro hledání na hyperkrychli

- Hyperkrychle je krychle v n -dimenzionálním prostoru s jednotkovou hranou.
- Každému vrcholu přiřadíme řetězec \vec{a} tvořený 0 a 1 odpovídající jeho souřadnicím.



Vyhledávací algoritmu založený na kvantové procházce pro hledání na hyperkrychli

- Hyperkrychle je krychle v n -dimenzionálním prostoru s jednotkovou hranou.
- Každému vrcholu přiřadíme řetězec \vec{a} tvořený 0 a 1 odpovídající jeho souřadnicím.
- Celkový Hilbertův prostor je $\mathcal{H} = \mathcal{H}^n \otimes \mathcal{H}^{2^n}$.



Vyhledávací algoritmu založený na kvantové procházce pro hledání na hyperkrychli

- Hyperkrychle je krychle v n -dimenzionálním prostoru s jednotkovou hranou.
- Každému vrcholu přiřadíme řetězec \vec{a} tvořený 0 a 1 odpovídající jeho souřadnicím.
- Celkový Hilbertův prostor je $\mathcal{H} = \mathcal{H}^n \otimes \mathcal{H}^{2^n}$.
- Založeno na článku: N. Shenvi, J. Kempe, K. B. Whaley: *A quantum random walk search algorithm*, Phys. Rev. A 67, 052307 (2003)



- Posunovací operátor má tvar

$$\hat{S} = \sum_{d=1}^n \sum_{\vec{a}} |d, \vec{a}_d\rangle \langle d, \vec{a}|.$$



- Posunovací operátor má tvar

$$\hat{S} = \sum_{d=1}^n \sum_{\vec{a}} |d, \vec{a}_d\rangle \langle d, \vec{a}|.$$

- Minci zvolíme následovně

$$\hat{C}_0 = -\hat{I} + 2|s^C\rangle \langle s^C|, \text{ kde } |s^C\rangle = \frac{1}{\sqrt{n}} \sum_{d=1}^n |d\rangle.$$



- Posunovací operátor má tvar

$$\hat{S} = \sum_{d=1}^n \sum_{\vec{a}} |d, \vec{a}_d\rangle \langle d, \vec{a}|.$$

- Minci zvolíme následovně

$$\hat{C}_0 = -\hat{I} + 2|s^C\rangle \langle s^C|, \text{ kde } |s^C\rangle = \frac{1}{\sqrt{n}} \sum_{d=1}^n |d\rangle.$$

- Evoluční operátor kvantové procházky je $\hat{U} = \hat{S} (\hat{C}_0 \otimes \hat{I})$.



- Chceme použít algoritmus na nalezení jednoho určitého stavu. Díky velké symetrii hyperkrychle můžeme předpokládat, že tento stav odpovídá $|0\rangle$.



- Chceme použít algoritmus na nalezení jednoho určitého stavu. Díky velké symetrii hyperkrychle můžeme předpokládat, že tento stav odpovídá $|0\rangle$.
- Algoritmus používá operátor kroku následujícího tvaru

$$\hat{U}' = \hat{U} - 2\hat{S}(|s^C\rangle\langle s^C| \otimes |0\rangle\langle 0|)$$



Algoritmus se skládá z následujících kroků:

- 1 Uvedeme system do počátečního stavu

$$|\varphi_0\rangle = \sum_{d=1}^n \sum_{a=0}^{2^n-1} \frac{1}{\sqrt{n2^n}} |d, \vec{a}\rangle,$$



Algoritmus se skládá z následujících kroků:

- 1 Uvedeme system do počátečního stavu

$$|\varphi_0\rangle = \sum_{d=1}^n \sum_{a=0}^{2^n-1} \frac{1}{\sqrt{n2^n}} |d, \vec{a}\rangle,$$

- 2 Aplikujeme operátor \hat{U}' $t = \left\lfloor \frac{\pi}{2} \sqrt{N} \right\rfloor$ -krát.



Algoritmus se skládá z následujících kroků:

- 1 Uvedeme system do počátečního stavu

$$|\varphi_0\rangle = \sum_{d=1}^n \sum_{a=0}^{2^n-1} \frac{1}{\sqrt{n2^n}} |d, \vec{a}\rangle,$$

- 2 Aplikujeme operátor \hat{U}' $t = \left\lfloor \frac{\pi}{2} \sqrt{N} \right\rfloor$ -krát.
- 3 Naměříme výsledný stav. S pravděpodobností $\frac{1}{2} - O\left(\frac{1}{n}\right)$ naměříme stav $|0\rangle$.



Při vyšetřování algoritmu se postupuje následovně

- Díky velké symetrii hyperkrychle a vlastnostem operátoru \hat{U}' se ukáže, že algoritmus lze převést na nevyváženou procházku na $2n$ -dimenzionální diskretní přímce.



Při vyšetřování algoritmu se postupuje následovně

- Díky velké symetrii hyperkrychle a vlastnostem operátoru \hat{U}' se ukáže, že algoritmus lze převést na nevyváženou procházku na $2n$ -dimenzionální diskretní přímce.
- Ukáže se, že vektor $|\varphi_0\rangle$ je aproximovaným vlastním vektorem \hat{U}' s vlastním číslem 1. Pak se najde stav $|\varphi_1\rangle$, který je také aproximovaným vlastním vektorem \hat{U}' s vlastním číslem 1. Také se ukáže, že pravděpodobnost naměření označeného vektoru, když je systém ve stavu $|\varphi_1\rangle$, je $\frac{1}{2} - O\left(\frac{1}{n}\right)$.



- Ukáže se, že vektory $|\varphi_0\rangle$ a $|\varphi_1\rangle$ se dají zapsat jako přibližná lineární kombinace dvou vlastních vektorů \hat{U}' s komplexně sdruženými vlastními čísly, které jsou velmi blízku jedničce.

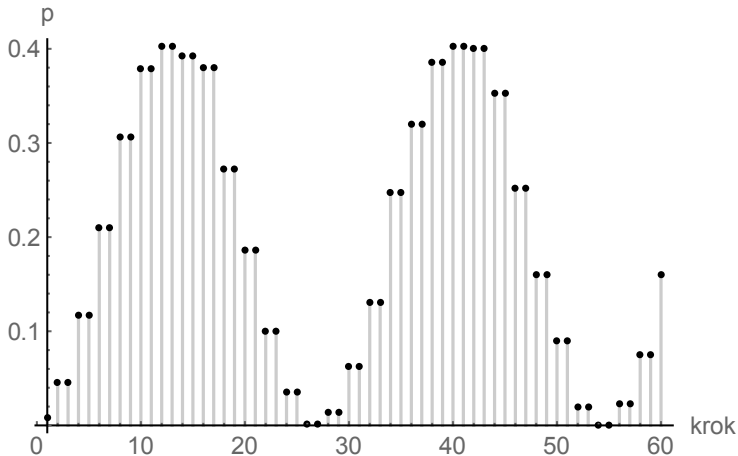


- Ukáže se, že vektory $|\varphi_0\rangle$ a $|\varphi_1\rangle$ se dají zapsat jako přibližná lineární kombinace dvou vlastních vektorů \hat{U}' s komplexně sdruženými vlastními čísly, které jsou velmi blízku jedničce.
- Operátor \hat{U}' pak v daném algoritmu působí jako přibližná rotace mezi stavy $|\varphi_0\rangle$ a $|\varphi_1\rangle$.



- Ukáže se, že vektory $|\varphi_0\rangle$ a $|\varphi_1\rangle$ se dají zapsat jako přibližná lineární kombinace dvou vlastních vektorů \hat{U}' s komplexně sdruženými vlastními čísly, které jsou velmi blízku jedničce.
- Operátor \hat{U}' pak v daném algoritmu působí jako přibližná rotace mezi stavy $|\varphi_0\rangle$ a $|\varphi_1\rangle$.
- Počet kroků, které algoritmus potřebuje k nalezení označeného prvku, je nepřímo úměrný úhlu rotace.





Obrázek: Vývoj pravděpodobnosti naměření označeného stavu pro hyperkrychli o dimenzi 7.



Porovnání Groverova algoritmu a algoritmu pro hyperkrychli založeného na kvantových procházkách.

- Oba algoritmy mají stejnou složitost $O(\sqrt{N})$ a oba začínají se superpozicí všech bazických stavů systému.



Porovnání Groverova algoritmu a algoritmu pro hyperkrychli založeného na kvantových procházkách.

- Oba algoritmy mají stejnou složitost $O(\sqrt{N})$ a oba začínají se superpozicí všech bazických stavů systému.
- Na rozdíl od Groverova algoritmu, který se dá reprezentovat jako rotace v 2-dimenzionálním systému, v procházkovém algoritmu je to pouze aproximovaná rotace. V prvním případě je prostor rotace tvořen počátečním a označeným stavem zatímco ve druhém případě má druhý vektor, který tvoří prostor rotace, pouze velký překryv s označeným stavem.



Porovnání Groverova algoritmu a algoritmu pro hyperkrychli založeného na kvantových procházkách.

- Oba algoritmy mají stejnou složitost $O(\sqrt{N})$ a oba začínají se superpozicí všech bazických stavů systému.
- Na rozdíl od Groverova algoritmu, který se dá reprezentovat jako rotace v 2-dimenzionálním systému, v procházkovém algoritmu je to pouze aproximovaná rotace. V prvním případě je prostor rotace tvořen počátečním a označeným stavem zatímco ve druhém případě má druhý vektor, který tvoří prostor rotace, pouze velký překryv s označeným stavem.
- Další rozdíl je v použití difuzního operátoru. Groverův algoritmus ho používá na celém N -dimenzionálním prostoru, zatímco procházkový algoritmus ho používá pouze na n -dimenzionálním podprostoru. Tohoto lze využít při různých typech fyzikální implementace.



Děkuji za pozornost.

