Programování na kvantovém počítači

P. Scheubrein[*], M. Chmelař[†], V. Michal[‡], M. Němec[§]

Abstrakt

Předmětem našeho miniprojektu bylo seznámení se s teoretickými podklady metod kvantové informatiky a porozumění rozdílům mezi filosofiemi běžných křemíkových a kvantových počítačů. Implementací Deutschova algoritmu jsme dokázali výhodnost této technologie.

1 Úvod

S pomocí webového rozhraní kvantového počítače společnosti IBM[3] chceme implementovat nejjednodušší kvantový algoritmus, který svou podstatou ukáže vyšší efektivitu některých kvantových algoritmů ve srovnání s algoritmy běžných počítačů.

2 Qubit

Qubit je základní jednotkou kvantové informatiky. Ten, podobně jako standardní bit, může nabývat stavů $|0\rangle$ a $|1\rangle$, které se s pomocí prostředků lineární algebry dají zachytit jako vektory rovnoběžné s osami souřadné soustavy

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

Na rozdíl od běžného bitu se však může qubit nalézat i v tzv. superpozici. Matematicky lze takový stav vyjádřit jako

$$\alpha|0\rangle + \beta|1\rangle,$$

kde komplexní čísla α a β představují koeficienty jednotlivých stavů. Měřením se ztratí superpozice qubitu a ten se ocitne s pravděpodobností $|\alpha|^2$ ve stavu $|0\rangle$ a pravděpodobností $|\beta|^2$ ve stavu $|1\rangle$. Z toho důvodu musí platit rovnost $|\alpha|^2 + |\beta|^2 = 1$.

Pro dva qubity můžeme naměřit následující čtyři stavy:

$$|0\rangle|0\rangle = |00\rangle = \begin{pmatrix}1\\0\\0\\0\end{pmatrix}, \quad |0\rangle|1\rangle = |01\rangle = \begin{pmatrix}0\\1\\0\\0\end{pmatrix}, \quad |1\rangle|0\rangle = |10\rangle = \begin{pmatrix}0\\0\\1\\0\end{pmatrix}, \quad |1\rangle|1\rangle = |11\rangle = \begin{pmatrix}0\\0\\0\\1\end{pmatrix}.$$

Superpozici těchto čtyř stavů lze poté popsat výrazem

$$\alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle. \tag{2}$$

[*]Gymnázium Třebíč - petr.sche@seznam.cz

[†]Střední průmyslová škola Třebíč - chmelarm.99@spst.eu

[‡]Gymnázium Na Vítězné pláni, Praha - vojta.michall@email.cz

[§]Gymnázium Brno, Vídeňská - michaelnemec47@gmail.com

3 Kvantová hradla

Pro hradla kvantového počítače je klíčová jedna vlastnost a to reverzibilita. Tuto vlastnost nemají všechna hradla normálních křemíkových počítačů. Reverzibilita v praxi znamená, že jsme schopni zjistit počáteční stav vstupního qubitu na základě znalosti použitého hradla a výsledného stavu. Kvantová hradla se stejně jako klasická hradla rozdělují na hradla, která mají jeden vstup (unární, z klasických např. NOT) a hradla se dvěma vstupy (binární, klasicky AND).

Operace na jednom qubitu

Pauliho hradlo X je kvantovým ekvivalentem klasické negace. Na svůj jeden vstupní qubit aplikuje operaci

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \qquad \begin{aligned} X|0\rangle &\longrightarrow |1\rangle \\ X|1\rangle &\longrightarrow |0\rangle, \end{aligned} \tag{3}$$

obecně

$$X(\alpha|0\rangle + \beta|1\rangle) \longrightarrow \beta|0\rangle + \alpha|1\rangle.$$

Hadamardovo hradlo H aplikuje na svůj jeden vstupní qubit operaci

$$H = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix}, \qquad \begin{aligned} H|0\rangle &\longrightarrow \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle \\ H|1\rangle &\longrightarrow \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle, \end{aligned} \tag{4}$$

obecně

$$H(\alpha|0\rangle + \beta|1\rangle) \longrightarrow \frac{1}{\sqrt{2}}(\alpha+\beta)|0\rangle + \frac{1}{\sqrt{2}}(\alpha-\beta)|1\rangle.$$

Pomocí Hadamardova hradla je qubit v základním stavu převeden do superpozice. Aplikace dvou Hadamardových hradel za sebou zanechá qubit v nezměněném stavu.

Operace na dvou qubitech

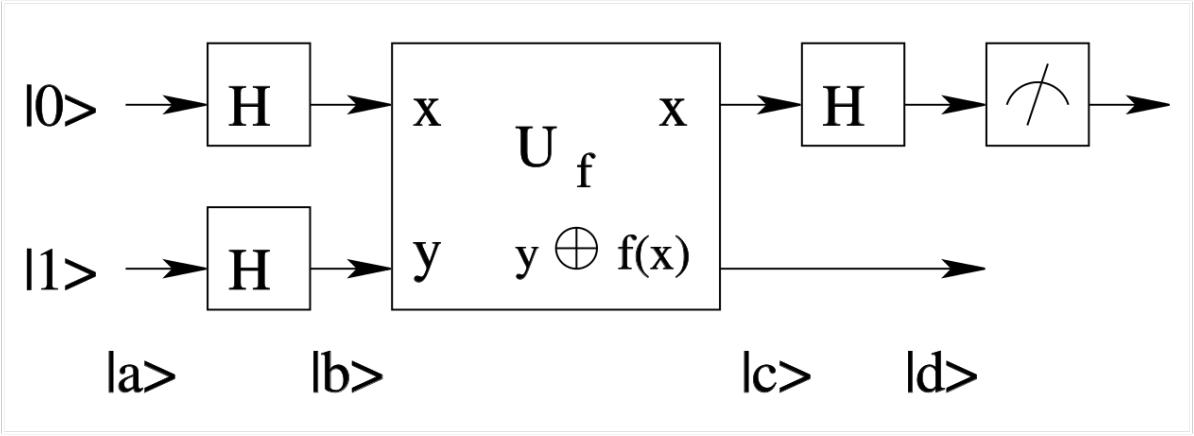
Hradlo CNOT provádí podmíněnou negaci, která změní stav druhého qubitu, pokud je první qubit ve stavu $|1\rangle$. Dochází tudíž k aplikaci následující operace

$$CNOT = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \qquad \begin{aligned} CNOT|00\rangle &\longrightarrow |00\rangle \\ CNOT|01\rangle &\longrightarrow |01\rangle \\ CNOT|10\rangle &\longrightarrow |11\rangle \\ CNOT|11\rangle &\longrightarrow |10\rangle, \end{aligned} \tag{5}$$

obecně

$$CNOT|x\rangle|y\rangle \longrightarrow |x\rangle|x \oplus y\rangle.$$

je patrné, že hradlo CNOT provádí výlučnou disjunkci svých dvou operandů a je tudíž analogií brány XOR běžných počítačů.



Obrázek 1: Schéma Deutschova algoritmu.

4 Deutschův algoritmus

Deutschův algoritmus byl vytvořen jako jednoduchý příklad řešení problému, který nelze vyřešit pomocí klasických algoritmů tak efektivně, jako může být vyřešen použitím kvantového počítače [1].

Problém je definován následovně: Funkce $f : \{0,1\} \to \{0,1\}$ je zaručeně buď konstantní, nebo balancovaná. Úkolem je zjistit, kterou z těchto vlastností taková funkce má [2].

Uvažujeme-li funkce pouze s jedním parametrem, pak je konstantní funkce taková, která nezávisle na vstupních parametrech vrací stále stejnou hodnotu a balancovaná funkce naproti tomu taková, která vrací 1 pro jednu hodnotu parametru 0 pro druhou hodnotu.

Pokud bychom chtěli řešit tento problém pomocí klasického algoritmu, musíme nezbytně vyčíslit funkci pro obě hodnoty parametru a porovnat oba výsledky. Rychlost výpočtu tedy bude limitována především dvojím vyhodnocováním funkce f. S tím nám může pomoci právě Deutschův algoritmus. Pro jeho implementaci ovšem nejdříve musíme implementovat i funkci f v kontextu kvantových hradel. Takové funkce jsou 4:

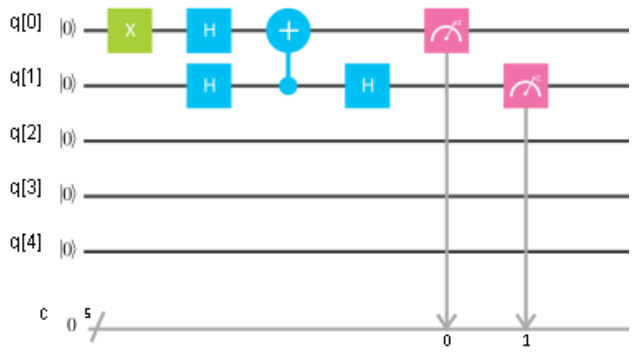
- Identita: $f(x) = x$
- Negace: $f(x) = \neg x$
- Pravda: $f(x) = 1$
- Nepravda: $f(x) = 0$

Konstantní funkce (pravda, nepravda) zároveň nemůže operovat na jednom qubitu, protože by nemohla být reverzibilní (nemůžeme určit jaké ze dvou původních hodnot nabýval parametr pokud výstup funkce nabývá pouze jedné hodnoty). Proto pro Deutschův algoritmus budeme pracovat se dvěma qubity x a y, kde x je vstup funkce f, a hledat funkci U_f jejíž výstup bude $y \oplus f(x)$ (viz Obrázek 1).

Na vstupu Deutschova algoritmu nastavíme qubit x na stav $|0\rangle$ a y na stav $|1\rangle$. Po průchodu algoritmem skončí qubit y v superpozici $\frac{|1\rangle+|0\rangle}{\sqrt{2}}$ a qubit x ve stavu $|0\rangle$ pokud byla funkce f balancovaná (3a) a $|1\rangle$ pokud byla konstantní (3b). Vlastnost funkce tedy určíme už po jednom spuštění algoritmu (3c), narozdíl od dvou spuštění (pro $f(1)$ a pro $f(0)$) pro klasický algoritmus.

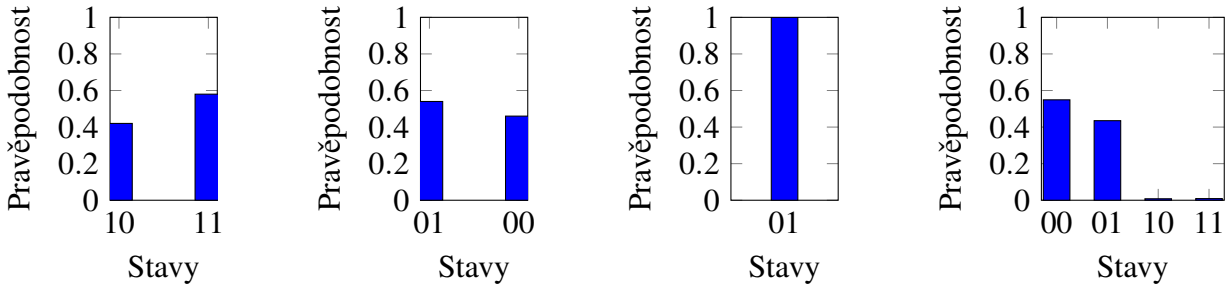
Tuto skutečnost jsme testovali implementováním algoritmů v prostředí IBM Q, kde jsme algoritmy pro všechny funkce simulovali a pro některé funkce i spouštěli na kvantovém počítači (3d).

Obrázek 2: Zapojení Deutchova algoritmu v IBM Q Experience s funkcí CNOT (balancovaná).



Obrázek 3: Výsledky simulací a reálného měření Deutchova algoritmu(různé funkce)

(a) Balancovaná fce. 100x (b) Konstantní fce. 100x (c) Konstantní fce. 1x (d) ibmqx4 1024x



5 Shrnutí

Teoreticky jsme navrhli, výpočtem ověřili správnost a v simulačním prostředí společnosti IBM úspěšně namodelovali několik kvantových obvodů. Námi implementovaný Deutschův algoritmus skutečně problém vyřešil efektivněji než běžný křemíkový procesor.

Poděkování

Chtěli bychom poděkovat naší odborné garantce Ing. Tereze Štefkové za skvělé předání vědomostí dané problematiky a za výbornou motivaci našeho týmu.

Reference

[1] DEUTSCH, D., RICHARD, J. Rapid solution of problems by quantum computation. *Proceedings of the Royal Society A*, 1992, č. 439. ISSN 1364-5021.

[2] CLEVE R., EKERT, A. Quantum Algorithms Revisited. *Philosophical Transactions of the Royal Society*, 1996, č. 454. ISSN 1364-5021.

[3] IBM, IBM Q Experience. https://quantumexperience.ng.bluemix.net/qx/experience