VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

ÚSTAV MATEMATIKY

INSTITUTE OF MATHEMATICS

KRYPTOANALÝZA SYMETRICKÉ KRYPTOGRAFIE POMOCÍ KVANTOVÝCH POČÍTAČŮ

CRYPTANALYSIS OF SYMMETRIC CRYPTOGRAPHY USING QUANTUM COMPUTERS

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

MAREK GOTTWALD

VEDOUCÍ PRÁCE

SUPERVISOR

Mgr. JIŘÍ PAVLŮ

BRNO 2023

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA STROJNÍHO INŽENÝRSTVÍ

Zadání bakalářské práce

Ústav:	Ústav matematiky
Student:	**Marek Gottwald**
Studijní program:	Matematické inženýrství
Studijní obor:	bez specializace
Vedoucí práce:	**Pavlů Jiří, Mgr.**
Akademický rok:	2022/23

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma bakalářské práce:

Kryptoanalýza symetrické kryptografie pomocí kvantových počítačů

Stručná charakteristika problematiky úkolu:

Popsat model útoku pomocí kvantového počítače. Pochopit Simonův algoritmus, jeho srovnání s Groverovým algoritmem. V modelu, kdy útočník nemá k dispozici orákulum, kterého se může dotazovat v superpozici, implementovat útok na jednoduchou šifru na simulátoru kvantových počítačů, využívající právě Simonův algoritmus.

Cíle bakalářské práce:

– Popis modelů útoku pomocí kvantového počítače na symetrické kryptosystémy
– Implementace útoku na jednoduchou symetrickou šifru v modelu, kdy útočník nemá k dispozici orákulum, kterého se může dotazovat v superpozici

Seznam doporučené literatury:

BONNETAIN, Xavier, et al. Quantum attacks without superposition queries: the offline Simon's algorithm. In: International Conference on the Theory and Application of Cryptology and Information Security. Springer, Cham, 2019. p. 552-583.

GROVER, Lov K. A fast quantum mechanical algorithm for database search. In: Proceedings of the twenty-eighth annual ACM symposium on Theory of computing. 1996. p. 212-219.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2022/23

V Brně, dne

L. S.

prof. RNDr. Josef Šlapal, CSc. ředitel ústavu	doc. Ing. Jaroslav Katolický, Ph.D. děkan fakulty

Abstrakt
Tato bakalářská práce se primárně zabývá popisem a implementací útoku na Even-Mansourovo schéma v modelu, kdy útočník nemá k dispozici orákulum, kterého se může dotazovat v superpozici. Práce nejprve popisuje úvod do kvantového počítání a matematický aparát nutný k pochopení dané problematiky. Dále se text zaměřuje na kvantové algoritmy, konkrétně Simonův, Groverův a offline Simonův algoritmus. Mimoto popisuje modely kvantových útoků.

Summary
This thesis is primarily concerned with the description and implementation of an attack on the Even-Mansour scheme in a model where the attacker does not have an oracle to query in superposition. The thesis first describes an introduction to quantum computation and the mathematical apparatus necessary to understand the problem. Next, the text focuses on quantum algorithms, specifically Simon's, Grover's, and offline Simon's algorithms. In addition, it describes models of quantum attacks.

Klíčová slova
Even-Mansourovo schéma, Simonův algoritmus, Groverův algoritmus, kvantové počítání

Keywords
Even-Mansour scheme, Simon's algorithm, Grover's algorithm, quantum computing

GOTTWALD, M. *Kryptoanalýza symetrické kryptografie pomocí kvantových počítačů*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2023. 32 s. Vedoucí Mgr. Jiří Pavlů

Prohlašuji, že jsem svou bakalářskou práci na téma *Kryptoanalýza symetrické kryptografie pomocí kvantových počítačů* zpracoval samostatně pod vedením Mgr. Jiřího Pavlů s použitím materiálů uvedených v seznamu použité literatury.

Marek Gottwald

Rád bych poděkoval vedoucímu své bakalářské práce Mgr. Jiřímu Pavlů za ochotu, trpělivost a cenné rady. Dále bych rád poděkoval své rodině a přátelům za podporu, již se mi od nich dostalo.

Marek Gottwald

Obsah

Úvod

Informační bezpečnost je koncept, který se dennodenně dotýká každého z nás. Lidstvo si již pomalu zvyklo na útočníky a jejich oběti. S technologickým pokrokem však pro útoky vznikají nové a nové možnosti. Tato bakalářská práce si klade za cíl jednu z nich představit. Jde o útok s využitím kvantových počítačů. Fenomén kvantových počítačů v dnešní době znamená nejen vyhlídky na lepší budoucnost, ale i rizika pro naši bezpečnost.

V této práci na úvod představujeme základní principy kvantového počítání a s tím spojený matematický zápis. Dále se zaobíráme kvantovými modely útoku a algoritmy, které v nich mohou být využity v kontextu symetrické kryptografie. Především se tento text zaměřuje na model, kdy útočník nemá přístup ke kvantovému orákulu. Na závěr přicházíme s popisem a implementací útoku na jednoduchou symetrickou šifru právě v tomto modelu.

1. Diracova notace

Tato kapitola se věnuje Diracově notaci, používané v kvantové mechanice.

Pro účely zápisu vektorů se v kvantové mechanice a počítání využívá Diracova notace, též známá jako braketová. Jde o stručný a praktický způsob jak popsat kvantové stavy.

Uvažujme n-rozměrný sloupcový vektor $\mathbf{u}$. V Diracově notaci je označován symbolem $|u\rangle$ a platí

$$\mathbf{u} = |u\rangle = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix}, \text{ kde } u_i \in \mathbb{C}. \tag{1.1}$$

Tento vektor se nazývá ket-vektor.

Podobně si můžeme zavést symbol $\langle u|$, jenž v braketové notaci označuje řádkový vektor $\mathbf{u}$. Platí tedy

$$\mathbf{u} = \langle u| = \begin{bmatrix} u_1^* & u_2^* & \cdots & u_n^* \end{bmatrix}, \text{ kde } u_i^* \in \mathbb{C},\ u_i u_i^* = |u_i|^2. \tag{1.2}$$

Tento vektor, který je duální ke ket-vektoru, nazýváme bra-vektor.

Definujeme-li vektor $|\psi\rangle$ jako lineární kombinaci n vektorů $|\alpha_1\rangle, |\alpha_2\rangle, \ldots, |\alpha_n\rangle$, to znamená

$$|\psi\rangle = c_1|\alpha_1\rangle + c_2|\alpha_2\rangle + \cdots + c_n|\alpha_n\rangle, \text{ kde } c_i \in \mathbb{C}, \tag{1.3}$$

pak pro jeho duální vektor $\langle\psi|$ platí

$$\langle\psi| = c_1^*\langle\alpha_1| + c_2^*\langle\alpha_2| + \cdots + c_n^*\langle\alpha_n|, \text{ kde } c_i^* \in \mathbb{C},\ c_i c_i^* = |c_i|^2. \tag{1.4}$$

Z toho plyne, že vektor $|\psi\rangle$ je hermitovsky sdružený ke komplexnímu vektoru $\langle\psi|$ a naopak. Potom platí rovnost

$$|\psi\rangle = (\langle\psi|^*)^T = \langle\psi|^\dagger. \tag{1.5}$$

Zavedeme-li n-rozměrný vektor $|v\rangle$ pak v této symbolice $\langle u|v\rangle$ značí vnitřní součin, $|u\rangle\langle v|$ vnejší součin a $|uv\rangle$ tenzorový součin. Pro tyto operace platí

$$\langle u|v\rangle = \langle u||v\rangle = \begin{bmatrix} u_1 & u_2 & \cdots & u_n \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} = u_1v_1 + u_2v_2 + \cdots + u_nv_n \tag{1.6}$$

$$|u\rangle\langle v| = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} \cdot \begin{bmatrix} v_1 & v_2 & \cdots & v_n \end{bmatrix} = \begin{bmatrix} u_1v_1 & u_1v_2 & \cdots & u_1v_n \\ u_2v_1 & u_2v_2 & \cdots & u_2v_n \\ \vdots & \vdots & \ddots & \vdots \\ u_nv_1 & u_nv_2 & \cdots & u_nv_n \end{bmatrix} \tag{1.7}$$

$$|uv\rangle = |u|v\rangle = |u\rangle|v\rangle = |u\rangle \otimes |v\rangle = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} = \begin{bmatrix} u_1 v_1 \\ u_1 v_2 \\ \vdots \\ u_1 v_n \\ u_2 v_1 \\ u_2 v_2 \\ \vdots \\ u_2 v_n \\ \vdots \\ u_n v_1 \\ u_n v_2 \\ \vdots \\ u_n v_n \end{bmatrix} \tag{1.8}$$

Operaci $\langle uv|$ lze definovat jako duální zobrazení k $|uv\rangle$.

2. Základy kvantového počítání

Tato kapitola se zabývá elementárními pojmy nutnými k pochopení principů kvantového počítání a na základě představených konceptů srovnává kvantový počítač s klasickým.

2.1. Kvantový počítač

Kvantový počítač je výpočetní stroj, který využívá specifických vlastností kvantové mechaniky. Musíme si uvědomit, že u klasického počítače také dochází, díky malým rozměrům tranzistorů, k aplikaci kvantové mechaniky. Na rozdíl od kvantového však klasický počítač nevyužívá kvantové superpozice a dalších níže popsaných jevů (viz *Fenomény kvantové mechaniky*. Data jsou v kvantovém počítači reprezentována qubity, jež mohou být ve stavu 0, 1, nebo v superpozici těchto dvou stavů. Klasický počítač naopak používá bity, kdy bit může být buď ve stavu 0, nebo 1.

Jakýkoli výpočetní problém řešitelný klasickým počítačem lze vyřešit právě tak pomocí kvantového počítače. Naopak je toto tvrzení platné za předpokladu dostatečného množství času. Jinými slovy, kvantové počítače nejsou v rozporu s Churchovou-Turingovou tezí. To znamená, že zatímco kvantový počítač nám neposkytuje žádnou výhodu nad klasickým počítačem z hlediska vyčíslitelnosti, kvantový algoritmus pro jistý problém vyžaduje výrazně méně času než jeho klasický protějšek.

Rozšířená Churchova-Turingova teze tvrdí, že každý problém, který lze efektivně vyřešit, je možné účinně spočítat pomocí klasického počítače (tj. Turingova stroje). Kvantový počítač je však jedním z mála modelů, který může tuto tezi narušit. Předpokládá se totiž, že existují úlohy, které kvantový počítač dokáže rychle vyřešit i přes to, že pro klasický počítač jsou v praxi neřešitelné (řešení by trvalo miliony let). Tímto by nám kvantový počítač otevřel nové perspektivy a rozšířil naše výpočetní možnosti.

2.2. Qubit

Qubit neboli kvantový bit je základní jednotkou kvantové informace. Jedná se o kvantový objekt, jehož stav je prvkem dvourozměrného Hilbertova prostoru $\mathcal{H}$ s ortonormální bází $\{|0\rangle, |1\rangle\}$. Z toho plyne, že jeho stav lze popsat jako lineární kombinaci, kdy pro prvek $|\psi\rangle \in \mathcal{H}$ platí

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \begin{bmatrix}\alpha\\ \beta\end{bmatrix}, \tag{2.1}$$

kde α, β jsou komplexní souřadnice vektoru v bázi $\{|0\rangle, |1\rangle\}$. Z výše uvedené rovnice je patrné, že bázové vektory $|0\rangle, |1\rangle$ jsou vektory ve tvaru

$$|0\rangle = \begin{bmatrix}1\\ 0\end{bmatrix}, |1\rangle = \begin{bmatrix}0\\ 1\end{bmatrix}. \tag{2.2}$$

Zároveň musí být splněna normalizační podmínka

$$|\alpha|^2 + |\beta|^2 = 1. \tag{2.3}$$

Koeficienty $|\alpha|^2$, $|\beta|^2$ nazývané pravděpodobnostní amplitudy je možné interpretovat jako pravděpodobnosti výskytu příslušného bázového stavu.

Vícequbitový systém je kvantový systém složený z $n > 1$ qubitů, kde $n \in \mathbb{N}$. Stav tohoto systému je prvkem Hilbertova prostoru $\mathcal{H}_n$, jenž vznikne jako tenzorový součin n Hilbertových prostorů $\mathcal{H}$ pro jednoqubitové systémy. Platí tedy

$$\mathcal{H}_n = \underbrace{\mathcal{H} \otimes \mathcal{H} \otimes \ldots \otimes \mathcal{H}}_{n} = \bigotimes_{i=1}^{n} \mathcal{H}. \tag{2.4}$$

Dimenze takového systému je rovna $\dim \mathcal{H}_n = 2^n$. Ortonormální báze vícequbitového systému je množina, která vznikne tenzorovým součinem bázových vektorů jednoqubitových systémů. Platí

$$|a_1 a_2 \ldots a_n\rangle = |a_1\rangle \otimes |a_2\rangle \otimes \ldots \otimes |a_n\rangle, \quad \forall i \in \{1, \ldots, n\}, a_i \in \{0, 1\}. \tag{2.5}$$

Tato báze se též nazývá výpočetní báze.

Zde je potřeba zdůraznit rozdíl mezi kvantovým a klasickým počítačem. Nejprve si uvědomme, že n bity můžeme reprezentovat 2^n možných stavů. Pro kvantový systém o n qubitech platí totéž, ale navíc jsme schopni se všemi 2^n stavy pracovat zároveň (viz *Kvantová superpozice*). Díky tomuto faktu může kvantový počítač získat výhodu nad klasickým počítačem při řešení určitých typů problémů.

2.3. Fenomény kvantové mechaniky

Níže jsou stručně popsány 4 jevy kvantové mechaniky, jež jsou pro kvantový počítač naprosto klíčové.

Kvantové měření představuje jediný způsob, jak získat informaci o stavu qubitu nebo systému více kvantových bitů. V kvantové fyzice platí, že kvantový objekt neexistuje ve zcela určitém stavu, dokud nedojde k interakci s jiným fyzikálním systémem. Budeme-li tedy chtít daný qubit pozorovat, narušíme stav tohoto kvantového objektu. Přesněji řečeno dojde ke kolapsu superpozice všech možných stavů qubitu na jeden z nich. To má za následek ztrátu části informace o tomto kvantovém objektu.

Kvantová superpozice je schopnost kvantového systému, kdy qubit je schopen až do svého změření existovat ve více stavech současně (v superpozici stavů). Příkladem superpozice je kvantový bit ve stavu $|\psi\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}}$. Pokud bychom chtěli stav tohoto kvantového bitu změřit, tak by došlo k redukci stavu superpožičního na jeden z bázových.

Superpozice umožňuje kvantovému počítači paralelně zpracovat více hodnot najednou, což jej odlišuje od počítače klasického. To může mít za následek exponenciální zrychlení určitých algoritmů. Tato vlastnost se nazývá kvantový paralelismus.

Kvantová provázanost je fyzikální jev, kdy jsou dva nebo více qubitů propojeny takovým způsobem, že změření jednoho ovlivní možný výsledek měření druhého. Dále pro kvantově provázaný stav platí

$$|\psi\rangle \text{ je kvantově provázaný stav} \iff |\psi\rangle \neq |\psi_1\rangle \otimes |\psi_2\rangle, \text{ kde } |\psi\rangle \in \mathcal{H}, |\psi_1\rangle \in \mathcal{H}_1, |\psi_2\rangle \in \mathcal{H}_2. \tag{2.6}$$

Z toho plyne, že v tomto případě nejsme schopni popsat stav jednoho qubitu nezávisle na ostatních (i kdybychom je rozdělili velkou vzdáleností). Jako příklad provázanosti lze uvést Bellovy stavy, což jsou nejvýše provázané stavy dvou kvantových bitů. Jeden z těchto stavů je kupříkladu ve tvaru $|\Phi^+\rangle = \frac{|00\rangle+|11\rangle}{\sqrt{2}}$. Pokud bychom u prvního qubitu naměřili stav 0 (resp. 1), tak s jistotou víme, že druhý qubit by byl rovněž ve stavu 0 (resp. 1).

Právě i tato vlastnost je důvodem, proč přidáním dalších kvantových bitů do systému může výpočetní výkon kvantového počítače narůst exponenciálně. Pro ucelenější pohled je třeba podotknout, že v případě klasických počítačů je třeba pro zdvojnásobení výpočetní síly zdvojnásobit i počet bitů, zatímco pro kvantové počítače stačí přidat jeden kvantový bit.

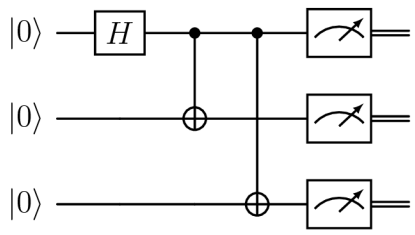
Kvantová interference je vlastnost, kdy dochází k zesílení (konstruktivní interferenci) nebo k vyrušení (destruktivní interferenci) pravděpodobnostní amplitudy kvantového systému. Tento neintuitivní jev si můžeme lépe představit, když si uvědomíme, že každý kvantový objekt lze popsat nejen částicí, ale i vlnou.

Využití interference je dalším přínosem kvantového počítače. Vhodným navržením kvantového algoritmu totiž dokážeme zvýšit pravděpodobnost požadovaných výsledků a eliminovat ty nežádoucí, protože pravděpodobnostní amplituda libovolného výstupu nemusí být nutně kladné číslo. Je důležité zmínit, že neobezřetným sestavením algoritmu by se naopak mohla zvýšit náchylnost k chybám. U klasických pravděpodobnostních počítačů však tato úvaha neplatí, jelikož každý možný výsledek klasického algoritmu má jistou nezápornou pravděpodobnost.

2.4. Kvantový obvod

Kvantový obvod je model pro kvantový výpočet, v němž výpočet představuje posloupnost kvantových operací (zvaných kvantové brány neboli hradla), měření, inicializací qubitů na dané hodnoty a případně dalších akcí aplikovaných na sadu qubitů. V kvantových obvodech probíhá vývoj času zleva doprava. Jednoduché horizontální hrany (dráty) představují qubity a dvojité horizontální hrany (dráty) představují klasické bity. Kvantový algoritmus se standardně realizuje a popisuje pomocí kvantového obvodu.

Zjednodušeně by se kvantový obvod dal popsat takto: Na začátku máme n qubitů začínajících ve stavu $|0\rangle$. Použitím posloupnosti kvantových bran dochází k modifikaci jednotlivých stavů, případně superpozice těchto stavů díky konstruktivním a destruktivním interferencím. Nakonec tento systém změříme a obdržíme jeden z vektorů výpočetní báze. Je důležité podotknout, že díky principu odloženého měření lze každé měření, k němuž dojde uvnitř obvodu, odložit až na konec kvantového výpočtu.



Obrázek 2.1: Příklad kvantového obvodu

Analogií ke kvantovému obvodu je u klasického počítače logický obvod tvořený logickými členy (hradly).

2.5. Kvantové brány

Kvantové brány jsou hlavní složkou kvantových obvodů, jelikož provádějí operace s qubity (tzn. mění jejich stav) k dosažení požadovaného výsledku. V závěru podkapitoly jsou stručně popsány typy zapojení a vybrané kvantové brány (operace) vzhledem k výpočetní bázi.

Obecně lze n-qubitovou operaci reprezentovat maticí o rozměrech $2^n \times 2^n$.

Každé kvantové hradlo je unitárním operátorem U Hilbertova prostoru $\mathcal{H}$, tedy platí

$$U^\dagger U = UU^\dagger = I, \tag{2.7}$$

kde I je operátor identity.

Z principů kvantové mechaniky plyne, že každá kvantová brána U je reverzibilní. Toto samozřejmě platí i pro sérii kvantových hradel. Použijeme-li na kvantový systém operátor U, tak můžeme obecně říci, že aplikací hermitovsky sdruženého operátoru $U^\dagger$ jsme schopni získat vstupní stav tohoto kvantového systému.

Hlavní rozdíl mezi kvantovými a logickými hradly je ten, že logické členy obecně reverzibilní nejsou.

Obecně lze jakýkoli klasický výpočet učinit reverzibilním (a tudíž implementovatelným jako kvantový obvod) za předpokladu, že se použije dostatečný počet pomocných qubitů. Pomocné kvantové bity jsou qubity, které začínají ve stavu $|0\rangle$ a po provedení výpočtu jsou odpočítáním (tzn. uplatněním stejných operací v opačném sledu) navráceny zpět do původního stavu. Odpočítání unitárního operátoru U znamená použití adjungovaného operátoru $U^\dagger$. Kvůli kvantovému provázání nemůžeme pomocné bity jednoduše ignorovat jako je tomu u klasických počítačů a pokud bychom je změřili, došlo by k narušení algoritmu.

Pro klasické výpočty je logický člen NAND univerzální operací, což znamená, že pomocí něj lze implementovat libovolný klasický výpočet. Pro kvantové počítání je možné implementovat NAND pomocí Toffoliho hradla. Tento poznatek je užitečný pro zajištění reverzibility logického obvodu při jeho převodu na kvantový obvod.

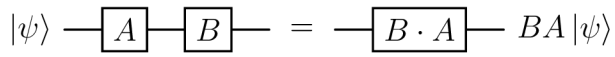
Je důležité připomenout, že kvantové měření je nevratné, a proto nemůže být kvantovou bránou.

Obrázek 2.2: Kvantový obvod obecné operace U působící na jeden qubit $|\psi\rangle$

Sériové zapojení dvou hradel A, B vyskytujících se v obvodu v tomto pořadí můžeme nahradit jedinou bránou C. Pro toto hradlo platí

$$C = B \cdot A, \tag{2.8}$$

kde $\cdot$ je násobení matic.



Obrázek 2.3: Sériové zapojení dvou kvantových bran

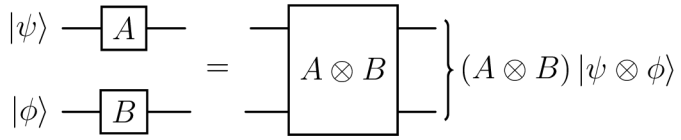
Paralelní zapojení dvou bran A, B vystupujících v obvodu v tomto pořadí (tedy shora dolů) lze nahradit jedním hradlem C. Pro tuto bránu platí

$$C = A \otimes B, \tag{2.9}$$

kde $\otimes$ je tenzorový součin. V případě matic se tento součin také někdy nazývá Kroneckerův.

Obecně pro n paralelně zapojených kvantových hradel U (tj. působících na n qubitů) platí

$$\underbrace{U \otimes U \otimes \cdots \otimes U}_{n} = \bigotimes_{1}^{n} U = U^{\otimes n} = U_n. \tag{2.10}$$

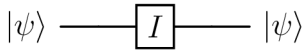


Obrázek 2.4: Paralelní zapojení dvou kvantových bran

Kvantová brána I je definována pro jeden qubit a reprezentuje ji jednotková matice, tedy

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \tag{2.11}$$

Toto hradlo nezmění stav kvantového bitu a je nezávislé na bázi Hilbertova prostoru. Má však své využití při matematickém zápisu různých kvantových algoritmů nebo pokud chceme v kvantovém obvodu zdůraznit neměnnost stavu kvantového objektu.



Obrázek 2.5: Kvantový obvod operace I působící na qubit $|\psi\rangle$

Kvantová brána NOT zvaná též Pauliho brána X je jednoqubitové hradlo reprezentované Pauliho maticí σ_x, pro kterou platí

$$\sigma_x = X = \mathrm{NOT} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}. \tag{2.12}$$

Pro její unitární a hermitovskou transformaci platí

$$X = X^\dagger = X^{-1},\ X^2 = I. \tag{2.13}$$

Použití tohoto operátoru prohodí hodnotu qubitu, to znamená

$$X|0\rangle = |1\rangle\,,\ X|1\rangle = |0\rangle\,. \tag{2.14}$$

Obrázek 2.6: Kvantový obvod operace X působící na qubit $\alpha|0\rangle + \beta|1\rangle$

Kvantová brána CNOT je dvouqubitové hradlo, kdy jeden qubit je kontrolní a druhý cílový. Maticová reprezentace tohoto operátoru je

$$\mathrm{CNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{2.15}$$

Pro jeho unitární a hermitovskou transformaci platí

$$\mathrm{CNOT} = \mathrm{CNOT}^{\dagger} = \mathrm{CNOT}^{-1}, \;\; \mathrm{CNOT}^2 = I. \tag{2.16}$$

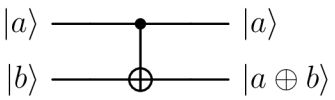
Tento operátor provede operaci NOT na cílovém kvantovém bitu pouze za předpokladu, že kontrolní kvantový bit je ve stavu $|1\rangle$, tedy

$$\mathrm{CNOT}\,|00\rangle = |00\rangle\,, \;\; \mathrm{CNOT}\,|01\rangle = |01\rangle\,, \;\; \mathrm{CNOT}\,|10\rangle = |11\rangle\,, \;\; \mathrm{CNOT}\,|11\rangle = |10\rangle\,, \tag{2.17}$$

což se dá zapsat jako funkce

$$|a, b\rangle \to |a, a \oplus b\rangle\,. \tag{2.18}$$

Symbol $\oplus$ představuje logickou operaci XOR nebo v tomto případě i součet modulo 2. Pro zajímavost uveďme, že kvantovou bránu CNOT lze použít pro vytvoření provázaného stavu.



Obrázek 2.7: Kvantový obvod operace CNOT působící na qubity $|a\rangle$, $|b\rangle$

Kvantová brána CCNOT neboli Toffoliho hradlo představuje trojqubitový operátor, kdy dva qubity jsou kontrolní a jeden cílový. Maticové vyjádření je ve formě

$$\mathrm{CCNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \tag{2.19}$$

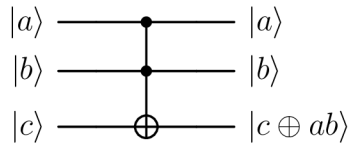
Pro jeho unitární a hermitovskou transformaci platí

$$\mathrm{CCNOT} = \mathrm{CCNOT}^{\dagger} = \mathrm{CCNOT}^{-1}, \;\; \mathrm{CCNOT}^2 = I, \tag{2.20}$$

což můžeme vyjádřit jako funkci ve tvaru

$$|a, b, c\rangle \to |a, b, c \oplus (a \wedge b)\rangle\,. \tag{2.21}$$

Symboly $\wedge$, $\oplus$ vyjadřují logický AND, respektive XOR.



Obrázek 2.8: Kvantový obvod operace CCNOT působící na qubity $|a\rangle$, $|b\rangle$

Kvantová brána H je definována pro jeden qubit a reprezentována maticí ve tvaru

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}. \tag{2.22}$$

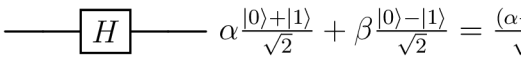
Pro její unitární a hermitovskou transformaci platí

$$H = H^\dagger = H^{-1},\ H^2 = I. \tag{2.23}$$

Tato operace vyjadřující Hadamardovu transformaci vytvoří ze stavů bázových stavy superpoziční, přesněji

$$H\,|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = |+\rangle\,,\ H\,|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = |-\rangle\,. \tag{2.24}$$

Této vlastnosti se hojně využívá v kvantových algoritmech, neboť jsme například schopni klást superpoziční dotazy.



Obrázek 2.9: Kvantový obvod operace H působící na qubit $\alpha\,|0\rangle + \beta\,|1\rangle$

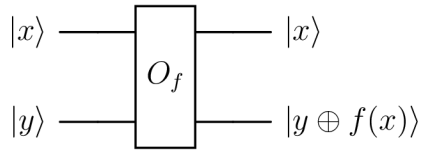
Na závěr je potřeba říci, že kombinací Hadamardova a Toffoliho hradla jsme schopni vyjádřit každou unitární operaci jako konečnou posloupnost těchto bran.

2.5.1. Kvantové orákulum

Mnoho kvantových algoritmů vyžaduje přístup ke kvantovému orákulu, což je speciální typ kvantové (reverzibilní) brány. Rozdíl oproti klasickému orákulu spočívá v tom, že dotazy jsme schopni pokládat v superpozici. V tom tkví další výhoda kvantového počítání, protože jsme díky tomu schopni paralelně zkoumat různé vstupy a tím zefektivnit naše výpočty. Orákulum si v podstatě můžeme představit jako černou skříňku, což je systém, u něhož je nám znám jeho vstup a výstup, ale nemáme znalosti o jeho vnitřním fungování. Kvantové orákulum lze matematicky reprezentovat pomocí funkce

$$|x, y\rangle \to |x, y \oplus f(x)\rangle\,. \tag{2.25}$$

Uvažujeme-li, že qubit $|y\rangle$ začína ve stavu $|0\rangle$, funkční zápis se zjednoduší na tvar $|x, y\rangle \to |x, f(x)\rangle$.



Obrázek 2.10: Schéma kvantového orákula O_f

2.6. Problémy dnešních kvantových počítačů

Kvantové počítače a s tím spojené počítání mají potenciál nejen zásadně změnit oblast výpočetní techniky, ale i další aspekty lidské činnosti. Proto je na závěr této kapitoly vhodné jmenovat i některé problémy kvantových počítačů, které plynou především ze současných technologických omezení. Vývoj kvantových počítačů je však na vzestupu, tudíž lze doufat, že tyto překážky jsou pouze dočasné.

Šum a dekoherence jsou jedny z největších výzev, jimž dnes výzkum čelí. Fyzické qubity jsou extrémně citlivé na okolní prostředí, kdy jakýkoli šum z okolí může vyvolat poruchu a narušit přesnost a spolehlivost výpočtu. S tím souvisí pojem dekoherence, což je interakce kvantového bitu s okolím, při němž dochází k poruše a kolapsu superpozičního stavu. V dnešní době mají kvantové systémy tedy stále tendenci chybovat. Je ale vhodné zmínit, že i klasické počítače nebyly od počátku bezchybné. První generace klasických počítačů byla poměrně nespolehlivá, protože docházelo například k častému praskání elektronek uvnitř obvodů.

Škálovatelnost je dalším problémem, neboť sestrojení kvantového počítače o velkém počtu qubitů není vůbec jednoduché. V současnosti je počet kvantových bitů, s nimiž dokážeme operovat, pořád relativně malý. I z tohoto pohledu není zatím potenciál kvantové výpočetní techniky naplněn.

Korekce chyb je metoda snažící se zajistit přesnost a spolehlivost kvantových výpočtů aplikací k tomu určených algoritmů a zabránit tak vlivu šumu a dekoherence. Metoda je to náročná, protože chyby je v porovnání s klasickým počítačem mnohem těžší odhalit a opravit. Kvantová korekce chyb vyžaduje nemalé množství prostředků, jako například část výpočetního výkonu (tzn. kvantových bitů) počítače.

Nedostatek kvantových algoritmů nás limituje v počtu problémů, jež jsme schopni řešit o mnoho efektivněji a rychleji pomocí kvantového počítače než na klasickém počítači. Návrhy nových kvantových algoritmů si žádají zcela nové přístupy a myšlenkové postupy, poněvadž kvantový a klasický výpočet nejsou totéž.

Vysoké náklady jsou další překážkou pojící se s konstrukcí a údržbou kvantového počítače, jelikož je zapotřebí specializovaná technika a vysoce kvalifikovaný personál. I z tohoto důvodu nejsou nejmodernější kvantové počítače veřejně dostupné.

Spotřeba energie představuje značnou výzvu pro provoz kvantových počítačů. To může do budoucna negativně ovlivnit vývoj kvantových počítačů, neboť s rozšiřováním kvantových počítačů o další qubity rostou také požadavky na množství energie.

3. Asymptotická složitost a vybrané kvantové algoritmy

Tato kapitola zavádí pojem asymptotická složitost a popisuje Simonův a Groverův algoritmus, jejichž znalost je pro další pochopení textu zcela zásadní.

Kvantový algoritmus je algoritmus, který oproti klasickému algoritmu využívá vlastností kvantového počítání ke zrychlení nebo jiného zvýšení efektivity řešení daného problému. Algoritmus můžeme popsat jako soubor instrukcí zadaných počítači.

V textu pro zjednodušení zápisu při popisu kvantových algoritmů zanedbáváme normalizační faktory kvantových stavů.

3.1. Asymptotická složitost

Při studiu a návrhu algoritmu se zaměřujeme na jeho časovou složitost, neboť nám to umožňuje posoudit jeho efektivitu a případně ji porovnat s jiným algoritmem. Časovou složitost algoritmu můžeme popsat jako funkci $g(n)$, jež udává počet operací, které algoritmus provede při vstupu o velikosti n. Vzhledem k často složitému chování funkce $g(n)$ a našemu zájmu o její základní charakteristiku byl zaveden pojem asymptotická složitost, kterou popisuje Landauova notace (zvaná též asymptotická nebo Bachmannova-Landauova notace notace). Asymptotická složitost slouží pouze k řádovému odhadu časové složitosti algoritmu, neboť neuvažuje multiplikativní a aditivní konstanty.

Definice 3.1. Nechť $O(f(x))$ je množina funkcí a $f(x)$, $g(x)$ jsou dvě funkce definované na nějaké podmnožině reálných čísel $\mathbb{R}$, kdy $x \to \infty$. Potom lze říci, že $g(x)$ je asymptoticky menší nebo rovna $f(x)$, což zapíšeme jako $g(x) \in O(f(x))$, právě když existují kladné konstanty c a x_0 takové, že pro všechna x větší než x_0 je $g(x)$ menší než $c \cdot f(x)$. Tedy platí

$$g(x) \in O(f(x)) \iff \exists x_0 > 0, c > 0, \forall x > x_0 : g(x) < c \cdot f(x). \tag{3.1}$$

To znamená, že funkce $g(x)$ roste v nekonečnu stejně rychle nebo pomaleji než $f(x)$. Jinými slovy, je-li časová složitost nějakého algoritmu $g(x) \in O(f(x))$, tak daný algoritmus probíhá asymptoticky stejně rychle nebo rychleji než $f(x)$.

Definice 3.2. Nechť $\Omega(f(x))$ je množina funkcí a $f(x)$, $g(x)$ jsou dvě funkce definované na nějaké podmnožině reálných čísel $\mathbb{R}$, kdy $x \to \infty$. Potom lze říci, že $g(x)$ je asymptoticky větší nebo rovna $f(x)$, což zapíšeme jako $g(x) \in \Omega(f(x))$, právě když existují kladné konstanty c a x_0 takové, že pro všechna x větší než x_0 je $g(x)$ větší než $c \cdot f(x)$. Tedy platí

$$g(x) \in \Omega(f(x)) \iff \exists x_0 > 0, c > 0, \forall x > x_0 : g(x) > c \cdot f(x). \tag{3.2}$$

To znamená, že funkce $g(x)$ roste v nekonečnu stejně rychle nebo rychleji než $f(x)$. Jinými slovy, je-li časová složitost nějakého algoritmu $g(x) \in \Omega(f(x))$, tak daný algoritmus probíhá asymptoticky stejně rychle nebo pomaleji než $f(x)$.

Definice 3.3. Nechť $\Theta(f(x))$ je množina funkcí a $f(x)$, $g(x)$ jsou dvě funkce definované na nějaké podmnožině reálných čísel $\mathbb{R}$, kdy $x \to \infty$. Potom lze říci, že $g(x)$ je asymptoticky

stejná jako $f(x)$, což zapíšeme jako $g(x) \in \Theta(f(x))$, právě když existují kladné konstanty c_1, c_2 a x_0 takové, že pro všechna x větší než x_0 platí $c_1 \cdot f(x) < g(x) < c_2 \cdot f(x)$. Tedy platí

$$g(x) \in \Theta(f(x)) \iff \exists x_0 > 0, c_1 > 0, c_2 > 0, \forall x > x_0 : c_1 \cdot f(x) < g(x) < c_2 \cdot f(x). \quad (3.3)$$

To znamená, že funkce $g(x)$ roste v nekonečnu stejně rychle jako $f(x)$ (tzn. chová se řádově jako $f(x)$). Jinými slovy, je-li časová složitost nějakého algoritmu $g(x) \in \Theta(f(x))$, tak daný algoritmus probíhá asymptoticky stejně rychle jako $f(x)$.

Mezi množinami $O(f(x))$, $\Omega(f(x))$ a $\Theta(f(x))$ platí vztah

$$\Theta(g(x)) = O(g(x)) \cap \Omega(g(x)). \quad (3.4)$$

Poznámka 3.4. V odborné literatuře se někdy vyskytuje zápis $g(x) \in \tilde{O}(f(x))$, pro který platí

$$g(x) \in \tilde{O}(f(x)) \iff \exists k : g(x) \in O(f(x) \log^k(f(x))). \quad (3.5)$$

Časová složitost algoritmu je určena funkcí $f(x)$ vystupující v zápisu Landauovy notace. Například časová složitost $O(n^2)$ představuje kvadratickou časovou složitost. To znamená, že doba běhu algoritmu se kvadraticky zvyšuje s velikostí vstupních dat n. Pokud je ale časová složitost $O(1)$, tak na velikosti vstupních dat n nezáleží. Dále poznamenejme, že polynomiální časovou složitost $O(n^k)$, kde $k > 1$, lze taktéž značit jako poly(n).

Algoritmus považujeme za efektivní, jestliže je jeho časová (a prostorová) složitost $O(n^k)$, kde $k < \infty$. Obecně platí, že algoritmus A je efektivnější než algoritmus B, pokud má menší časovou složitost než B pro stejný problém nebo stejnou velikost vstupních dat n.

Na závěr je potřeba zmínit, že Landauova notace se uplatňuje i při popisu asymptotické prostorové složitosti. V tomto případě ale uvažujeme závislost mezi prostorovými (paměťovými) nároky algoritmu a velikostí vstupních dat n.

3.2. Simonův algoritmus

Simonův algoritmus je kvantový algoritmus, jenž nám poskytuje exponenciální zrychlení níže popsaného problému.

Simonův problém. Nechť je dána funkce $f : \{0,1\}^n \to \{0,1\}^n$, která je buď injektivní, nebo existuje $s \in \{0,1\}^n$ takové, že pro funkci f platí

$$\forall x, y : f(x) = f(y) \iff y = x \vee y = x \oplus s, \quad (3.6)$$

kde symbol $\oplus$ značí bitový operátor XOR. Pak nalezněme s nebo určeme, že je funkce f injektivní.

Jinými slovy, pokud takové s existuje, funkce f má skrytou booleovskou periodu. Tento problém lze dále rozšířit na hledání skrytého booleovského posunu, kdy hledáme $s \in \{0,1\}^n$ takové, že pro dvě funkce $f, g : \{0,1\}^n \to \{0,1\}^n$ platí

$$\forall x : f = g \iff f(x) = g(x) \vee f(x) = g(x \oplus s). \quad (3.7)$$

Obecně mohou mít funkce f a g jako obor hodnot libovolnou množinu X za předpokladu, že ji lze dobře popsat. V našem případě se ale omezíme na funkce, jejichž obor hodnot jsou posloupnosti n binárních hodnot, tedy n-bitové řetězce. Tyto řetězce můžeme také interpretovat jako (binární) vektory n-dimenzionálního vektorového prostoru nad konečným tělesem $\mathbb{Z}_2$.

Máme-li přístup ke klasickému orákulu, tak vyřešení tohoto problému vyžaduje $\Omega(2^{n/2})$ dotazů, kdy hledáme periodu s funkce f (pokud vůbec existuje). V případě Simonova algoritmu, kdy máme přístup ke kvantovému orákulu O_f, si však vystačíme pouze s $O(n)$ superpozičními dotazy.

Simonův algoritmus realizujeme pomocí cn iterací Algoritmu 1 (viz níže), kdy zvolením malé celočíselné konstanty $c \geq 1$ zajistíme dostatečnou pravděpodobnost úspěšného nalezení skryté periody s (samozřejmě za předpokladu že existuje).

Věta 3.5. Předpokládejme, že funkce $f : \{0,1\}^n \to \{0,1\}^n$ má periodu $s \neq 0^n$, tedy $\forall x \in \{0,1\}^n : f(x \oplus s) = f(x)$, a platí

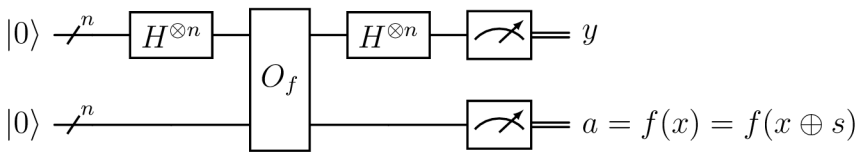
$$\max_{t \neq \{s,0^n\}} \Pr_x [f(x \oplus t) = f(x)] \leq \frac{1}{2}. \tag{3.8}$$

Aplikací Simonova algoritmu na funkci f získáme s s pravděpodobností alespoň $1 - 2^n \cdot (3/4)^{cn}$.

Výsledkem Simonova algoritmu může být následující:

- cn náhodných binárních vektorů y (funkce f není periodická),
- cn binárních vektorů y generujících polorovinu $y \cdot s = 0$, tedy vektorů kolmých na s (funkce f je periodická).

Dále je již snadné určit, zdali perioda s existuje či nikoliv. Toho docílíme řešením soustavy rovnic $y \cdot s = 0 \pmod{n}$. Operace $\cdot$ je násobení v tělese $\mathbb{Z}_2$, případně ji můžeme interpretovat jako bitový operátor AND. Potřebujeme $n - 1$ lineárně nezávislých vektorů, protože řešením soustavy je jednodimenzionální podprostor. Tento podprostor je tvořen nulovým vektorem, což je triviální řešení soustavy, a námi hledaným vektorem s. Neznámou s lze nalézt například pomocí Gaussovy eliminační metodoy v čase $O(n^3)$.



Obrázek 3.1: Schéma obvodu pro Simonův algoritmus

Algorithm 1 Subrutina Simonova algoritmu

Vstup: $n, O_f : |x\rangle |0\rangle \to |x\rangle |f(x)\rangle$, kde $f : \{0,1\}^n \to \{0,1\}^n$ je buď injektivní, nebo periodická

Výstup: y

1: Začneme v nulovém stavu $|0\rangle |0\rangle$, kde první a druhý registr obsahují n qubitů.

2: Aplikací Hadamardových bran $H^{\otimes n}$ na první registr obdržíme stav

$$\sum_{x\in\{0,1\}^n} |x\rangle |0\rangle .$$

3: Dotazem na kvantové orákulum O_f (tj. použitím kvantového hradla O_f) obdržíme stav

$$\sum_{x\in\{0,1\}^n} |x\rangle |f(x)\rangle = \sum_{a\in\{0,1\}^n} \left(\sum_{x\in\{0,1\}^n | f(x)=a} |x\rangle \right) |a\rangle .$$

4: Změřením registru a (popř. můžeme toto měření odložit) získáme náhodnou hodnotu $a \in \{0,1\}^n$ a stav

$$\sum_{x\in\{0,1\}^n | f(x)=a} |x\rangle ,$$

kde jsme druhý registr zanedbali, protože byl změřen.

5: Opětovným použitím Hadamardových hradel $H^{\otimes n}$ obdržíme stav:

$$\sum_{y\in\{0,1\}^n} \left(\sum_{x\in\{0,1\}^n | f(x)=a} (-1)^{x\cdot y} \right) |y\rangle .$$

6: Nyní změříme registr y. Mohou nastat dva případy:

- Funkce f nemá žádnou periodu s, v tomto případě dostaneme náhodné y.
- Funkce f má periodu s, v tomto případě pro pravděpodobnostní amplitudu stavu $|y\rangle$ platí

$$\sum_{x\in\{0,1\}^n | f(x)=a} (-1)^{x\cdot y} = (-1)^{x_0\cdot y} + (-1)^{(x_0\oplus s)\cdot y}.$$

Tato amplituda je nulová, pokud $y \cdot s = 1$. V opačném případě je nenulová.

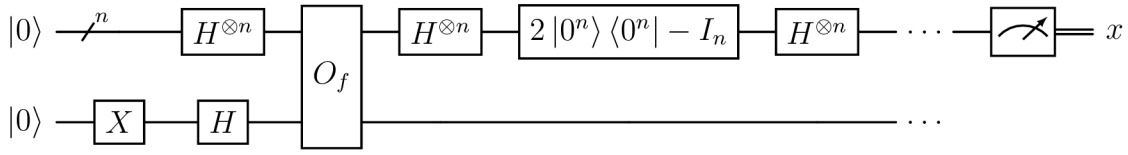
– V takové situaci měřením získáme náhodné y takové, že $y \cdot s = 0$.

3.3. Groverův algoritmus

Groverův algoritmus je kvantový algoritmus, jenž nám poskytuje (ve srovnání s klasickým případem) kvadratické zrychlení hledání v seznamu prvků. Přesněji řečeno, algoritmus dokáže vyřešit níže popsaný problém.

Groverův problém. Uvažujme množinu X (tzv. vyhledávací prostor), jejíž prvky jsou reprezentovány na $\lceil \log_2(|X|) \rceil$ qubitech tak, že rovnoměrná superpozice $\sum_{x\in X} |x\rangle$ je spočitatelná v čase $O(1)$. Předpokládejme, že máme kvantové orákulum (tzv. testovací orákulum) s přístupem k funkci $f : X \to \{0,1\}$. Potom hledejme $x \in X$ takové, že $f(x) = 1$.

Uvažujme klasický případ, kdy máme přístup pouze ke klasickému počítači (a tedy klasickému orákulu). Jestliže k prvku 1 existuje 2^t jeho předobrazů, tak předpokládáme, že jeden tento předobraz (tj. prvek $x \in X$) nalezneme s $O(|X|/2^t)$ klasickými dotazy v čase $O(|X|/2^t)$. Pokud ale máme přístup ke kvantovému počítači (a tedy kvantovému orákulu), tak jsme schopni pomocí Groverova algoritmu najít jeden předobraz s $\tilde{O}(\sqrt{|X|/2^t})$ superpozičními dotazy v čase $\tilde{O}(\sqrt{|X|/2^t})$. Jestliže kvantové orákulum využívá a pomocných qubitů, tak Groverův algoritmus potřebuje pouze $a + \lceil \log_2(|X|) \rceil$ kvantových bitů.



Obrázek 3.2: Schéma obvodu pro Groverův algoritmus

Algorithm 2 Groverův algoritmus

Vstup: $n = |X|, O_f : |x\rangle\,|0\rangle \to |x\rangle\,|f(x)\rangle$, kde $f : \{0,1\}^n \to \{0,1\}^n$

Výstup: x

1: Začneme ve stavu $|0\rangle\,|0\rangle$, kde první registr obsahuje n qubitů. Druhý registr představuje pomocný kvantový bit.

2: Použitím hradla X a Hadamardovy brány H (tzn. operátoru $H \cdot X$) na druhý registr dostaneme stav

$$|0\rangle\,|-\rangle$$

3: Aplikací Hadamardových bran $H^{\otimes n}$ na první registr obdržíme stav

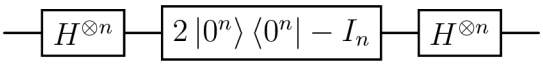
$$\sum_{x \in X} |x\rangle\,|-\rangle\,.$$

4: Provedeme k-krát následující následující dva kroky, kde k je $\tilde{O}(\sqrt{|X|/2^t})$.

- Aplikujeme testovací operátor O_f, kdy pro nějaké x platí

$$\begin{aligned} O_f(|x\rangle \otimes |-\rangle) &= \frac{1}{\sqrt{2}}(O_f\,|0\rangle\,|0\rangle - O_f\,|0\rangle\,|1\rangle) \\ &= \frac{1}{\sqrt{2}}(|x\rangle\,|f(x)\rangle - |x\rangle\,|1\rangle \oplus |f(x)\rangle) \\ &= \begin{cases} \frac{1}{\sqrt{2}}(-|x\rangle\,|0\rangle + |x\rangle\,|1\rangle) & \text{jestliže } f(x) = 1, \\ \frac{1}{\sqrt{2}}(|x\rangle\,|0\rangle - |x\rangle\,|1\rangle) & \text{jestliže } f(x) = 0 \end{cases} \end{aligned}$$

- Aplikujeme difuzní operátor. Schéma difuzního operátoru lze schématicky popsat jako



5: Změříme výsledný kvantový stav ve výpočetní bázi.

Pro ucelenější pohled můžeme vysvětlit funkci použitých operátorů. Aplikací testovacího operátoru O_f se pravděpodobnostní amplituda hledaného prvku x stahe zápornou,

což můžeme interpretovat jako jeho označení. Použitím difuzního operátoru dojde (ve srovnání s původní hodnotou příslušného normalizačního faktoru v kroku číslo 2) k zesílení pravděpodobnostní amplitudy hledaného prvku x. To má za následek zmenšení pravděpodobnostní amplitudy ostatních prvků. Díky tomu jsme po daném počtu iterací schopni naměřit hledaný prvek x s vysokou pravděpodobností.

4. Útok na EM schéma v modelu Q1

Tato kapitola představuje popis útoku na Even-Mansourovo schéma v modelu Q1 pomocí zdokonaleného Simonova algoritmu.

4.1. Even-Mansourovo schéma

Definice 4.1. Nechť je dána veřejně známá permutace $P : \{0,1\}^n \to \{0,1\}^n$ a dva n-bitové klíče $k_1, k_2 \in \{0,1\}^n$. Pak pro Even-Mansourovu šifru $E_{k_1,k_2} : \{0,1\}^n \to \{0,1\}^n$ platí

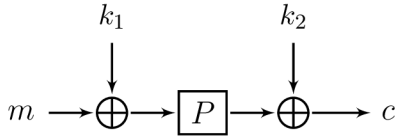
$$E_{k_1,k_2}(m) = P(k_1 \oplus m) \oplus k_2 = c, \tag{4.1}$$

kde $\oplus$ značí binární operaci XOR a $m, c \in \{0,1\}^n$ jsou prostý a šifrový text. Pro dešifrovací funkci $D_{k_1,k_2} : \{0,1\}^n \to \{0,1\}^n$ tedy platí

$$D_{k_1,k_2}(c) = P^{-1}(k_2 \oplus c) \oplus k_1 = m. \tag{4.2}$$

Even-Mansourovo schéma (zkráceně EM schéma) je symetrická (jednobloková) šifra, což je šifrovací algoritmus využívající k šifrování i dešifrování jeden klíč. V tomto kontextu je naším klíčem $k = k_1||k_2$, kde symbol $||$ představuje operaci zřetězení. Jde v podstatě o schéma konceptuálně nejjednodušší šifry, která ještě může být bezpečná. Šifra je bezpečná i v případě identických klíčů, neboť struktura šifry nebude narušena. Pokud bychom ale jeden z klíčů k_1, k_2 odstranili, tak šifra již bezpečná nebude.

Uvažujeme-li, že nemáme přístup ke kvantovému počítači, tak tato konstrukce je přibližně bezpečná až do $O(2^{n/2})$ dotazů na orákulum D a výpočtů T (tj. počtu operací). Pro složitost klasického útoku platí $D \cdot T = 2^n$, kdy nejlepšího výsledku dosáhneme při $T = D = 2^{n/2}$. Pro upřesnění poznamenejme, že obecně $T = O(\cdot)$ a $D = O(\cdot)$.



Obrázek 4.1: Even-Mansourovo schéma

4.2. Modely kvantových útoků

Kvantové útoky můžeme rozdělit do dvou kategorií (modelů), které se liší v prostředcích, jež má útočník k dispozici.

Model Q1 představuje typ útoku, kdy útočník může provádět libovolné offline výpočty pomocí kvantového počítače, ale online dotazy na orákulum jsou realizovány klasickým způsobem. Vzhledem k současným technologiím je tento typ útoku ve srovnání s modelem Q2 proveditelnější.

Tato kapitola se zaměřuje na útok právě v tomto modelu. Útok realizujeme offline verzí Simonova algoritmu. Pro složitost tohoto útoku platí $D \cdot T^2 = 2^n$, kdy nejlepšího výsledku docílíme při $T = D = 2^{n/3}$.

Model Q2 popisuje útok, ve kterém je útočník schopen nejen offline kvantových výpočtů, ale také online superpozičního dotazování na kvantové orákulum. Tento model útoku je pro útočníka samozřejmě výhodnější, neboť přístup ke kvantovému orákulu dokáže jeho útoky učinit efektivnějšími. V praxi ale zatím není takový typ útoku moc reálný, protože oběť by musela nechat běžet šifru na kvantovém počítači s kvantovým přístupem. Není důvod, proč by to měla dělat. Stálo by ji to nemalé prostředky a vystavovala by se riziku možných útoků v tomto modelu.

Útok na EM schéma v modelu Q2 lze realizovat Simonovým algoritmem, kde klíč k_1 je skrytou periodou funkce $F(x) = E_{k_1,k_2}(x) \oplus P(x)$. Jinými slovy, využijeme algebraické struktury E_{k_1,k_2}. Určení hodnoty klíče k_2 je již otázkou klasického postprocessingu. Pro složitost tohoto útoku platí $T = D = n$. Oproti klasickému útoku tedy dojde k exponenciálnímu zrychlení. Je třeba zdůraznit, že pokud známe jeden z klíčů, tak šifra už není bezpečná, a tedy druhý klíč již určíme snadno.

4.3. Popis útoku

Náš útok na EM schéma v modelu Q1 lze interpretovat jako řešení problému asymetrického hledaní periody, což je kombinace Simonova a Groverova problému.

Problém asymetrického hledaní periody Nechť jsou dány dvě funkce $F : \{0,1\}^m \times \{0,1\}^n \to \{0,1\}^l$ a $g : \{0,1\}^n \to \{0,1\}^l$. Funkci F uvažujeme jako množinu i funkcí a píšeme $F(i,\cdot) = f_i(\cdot)$, kde $i \in \{0,1\}^m$. Předpokládáme, že máme kvantové orákulum s přístupem k funkci F a klasické orákulum s přístupem k funkci g.

Pak předpokládejme, že existuje právě jedno $i_0 \in \{0,1\}^m$ takové, že funkce $f_{i_0} \oplus g$ má skrytou periodu s, tedy pro nějaké s platí

$$\forall x \in \{0,1\}^n, \exists! i_0 \in \{0,1\}^m : f_{i_0}(x) \oplus g(x) = f_{i_0}(x \oplus s) \oplus g(x \oplus s). \tag{4.3}$$

Dále pro zjednodušení problému předpokládejme, že platí

$$\max_{\substack{i \in \{0,1\}^m \setminus \{i_0\} \\ t \in \{0,1\}^n \setminus \{0^n\}}} \Pr_{x \leftarrow \{0,1\}^n} [(f_i \oplus g)(x \oplus t) = (f_i \oplus g)(x)] \leq \frac{1}{2}. \tag{4.4}$$

Pak nalezněme i_0 a s.

Uvažujeme, že jedno vyčíslení funkce (např. blokové šifry) lze pro jednoduchost provést v čase $O(1)$ a předpokládáme, že délky klíčů n-bitové blokové šifry jsou v $O(n)$.

Pro realizaci našeho útoku rozdělíme klíč k_1 na dvě části – $k_1^{(1)}$ o u bitech a $k_1^{(2)}$ o $n-u$ bitech. Potom platí, že $k_1 = k_1^{(1)} || k_1^{(2)}$.

Dále nechť pro celé číslo u platí $0 \leq u \leq n$. Definujme funkci $F : \{0,1\}^{n-u} \times \{0,1\}^u \to \{0,1\}^n$ jako množinu funkcí f, kdy platí

$$F(i,x) = f_i(x) = P(x||i), \tag{4.5}$$

a funkci $g : \{0,1\}^u \to \{0,1\}^n$, pro kterou platí

$$g(x) = E_{k_1,k_2}(x||0^{n-u}). \tag{4.6}$$

Potom platí, že funkce $F(k_1^{(2)}, x) \oplus g(x)$ má periodu $k_1^{(1)}$, neboť $F(k_1^{(2)}, x) \oplus g(x) = P(x||k_1^{(2)}) \oplus P((x \oplus k_1^{(1)})||k_1^{(2)}) \oplus k_2$. Této vlastnosti v našem útoku záměrně využijeme. Je-li P náhodná permutace, lze předpokládat, že funkce $f_i \oplus g = P(\cdot||i) \oplus E_{k_1,k_2}$ není periodická pro všechna $i \neq k_1^{(2)}$ a že předpoklad (4.4) platí.

Útok realizovaný offline verzí Simonova algoritmu by se dal shrnout do třech kroků:

1. Nalezení $k_1^{(2)}$ pro výše uvedené funkce F a g pomocí algoritmu Alg-ExpQ1.
2. Nalezení $k_1^{(1)}$ pro funkce $f_{k_1^{(2)}}$ a g pomocí algoritmu SimQ1.
3. Určení k_2, kde $k_2 = E_{k_1,k_2}(0^n) \oplus P(k_1)$.

Z Věty 4.3 vyplývá, že algoritmus Alg-ExpQ1 v kroku 1 s vysokou pravděpodobností určí hledané $k_1^{(2)}$ pomocí $O(2^u)$ (online) klasických dotazů na g. Offline výpočet algoritmu proběhne za čas $O(n^3 2^{(n-u)/2})$. Připomeňme, že jedno vyčíslení funkce g (resp. F) lze provést $O(1)$ vyčísleními funkce E_{k_1,k_2} (resp. P).

Dále z Věty 4.4 plyne, algoritmus SimQ1 v kroku 2 s vysokou pravděpodobností nalezne $k_1^{(2)}$ pomocí $O(2^u)$ (online) klasických dotazů na g. Offline výpočet algoritmu proběhne za čas $O(n^3)$.

Určení k_2 v kroku 3 vyžaduje $O(1)$ dotazů na E_{k_1,k_2} a $O(1)$ offline výpočtů.

Zvolením dostatečně velké celočíselné konstanty $c \geq 1$ zajistíme dostatečnou pravděpodobnost úspěšného nalezení $k_1^{(1)}$, $k_1^{(2)}$ (viz Věta 4.5).

Celkem vzato náš útok nalezne s vysokou pravděpodobností $k_1^{(1)}$, $k_1^{(2)}$ pomocí $D = O(2^u)$ klasických dotazů na E_{k_1,k_2} a $T = O(n^3 2^{(n-u)/2})$ offline výpočtů. Pro nalezení kompromisu mezi T a D určíme rovnost $T = D = \tilde{O}(2^{n/3})$. Kombinace algoritmů Alg-ExpQ1 a SimQ1 využívá poly(n) qubitů a poly(n) klasické paměti počítače.

4.4. Algoritmus pro vytvoření g-databáze

Tento algoritmus je prvním krokem algoritmů Alg-ExpQ1 a SimQ1. Umožňuje nám klasickými dotazy na funkci g připřavit g-databázi $|\psi_g\rangle$, což je v podstatě kvantový stav obsahující hodnoty $x \in \{0,1\}^n$ a k nim příslušné funkční hodnoty $g(x)$. Tuto databázi nám stačí vytvořit pouze jednou pro oba algoritmy.

Sestavení této databáze vyžaduje přibližně 2^n času a klasických dotazů.

Algorithm 3 Vytvoření g-databáze $|\psi_g\rangle$

Vstup: g
Výstup: g-databáze

$$|\psi_g\rangle = \bigotimes^{cn} \left(\sum_{x\in\{0,1\}^n} |x\rangle\,|g(x)\rangle \right)$$

1: Začneme v nulovém stavu

$$\bigotimes^{cn} |0\rangle\,|0\rangle\,.$$

2: Aplikací Hadamardových bran H na první registr obdržíme stav

$$\bigotimes^{cn} \sum_{x\in\{0,1\}^u} |x\rangle\,|0\rangle\,.$$

3: Pro každé $x \in \{0,1\}^n$ se klasicky zeptáme na $g(x)$. Potom použijeme unitární operátor, který zapíše hodnoty $g(x)$ do druhého registru, jestliže první registr obsahuje hodnotu x.

4.5. Algoritmus Alg-ExpQ1

Algoritmus Alg-ExpQ1 je v zásadě kombinací Simonova a Groverova algoritmu, kdy testovací orákulum Grovera algoritmu je Simonův algoritmus.

Tento algoritmus by se dal slovně shrnout do dvou kroků:

1. Online dotazy: Uskutečnění 2^n klasických dotazů na g a vytvoření kvantového stavu $|\psi_g\rangle$
2. Offline výpočty: Spuštění Groverova algoritmu pro vyhledávací prostor $i \in \{0,1\}^m$, který zahrnuje provedení procedury test pro každou hodnotu i. Tato procedura pomocí stavu $|\psi_g\rangle$ a kvantových dotazů na f_i otestuje, zdali pro dané i platí, že funkce $f_i \oplus g$ je periodická. Následně odpočítáním dojde k navrácení do původního stavu $|\psi_g\rangle$.

Nalezení periody. Výše popsaný algoritmus Alg-ExpQ1 sice nalezne i_0 takové, že funkce $f_{i_0} \oplus g$ je periodická, ale již neurčí příslušnou periodu. Pokud bychom chtěli danou periodu určit, tak musíme na tuto funkci použít Simonův algoritmus (popř. znovu použít g-databázi a lehce upravený operátor test). Proto jsme zavedli algoritmus SimQ1.

Poznámka 4.2. Výskyt chyby v každé iteraci Algoritmu 5 je ohraničen maximem z $p^{(i)} := \Pr[\dim(\mathrm{Span}(u_1,\ldots,u_{cn})) < n]$, kde vektory $u_1,\ldots,u_{cn}$ jsou generovány Simonovým algoritmem. Hodnotu $p^{(i)}$ můžeme interpretovat jako pravděpodobnost, že Simonův algoritmus označí funkci $f_i \oplus g$ jako periodickou, i když ve skutečnosti není. Na základě předpokladu (4.4) lze ukázat, že platí nerovnost $p^{(i)} \leq 2^{(n+1)/2}((1+\frac{1}{2})/2)^{cn/2}$.

Algorithm 4 Alg-ExpQ1

Vstup: f, g

Výstup: i

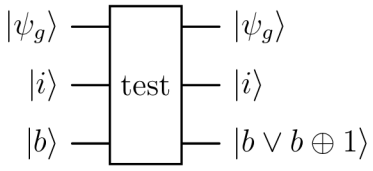
1: Pomocí 2^n klasických dotazů na g vytvoříme kvantový stav $|\psi_g\rangle$ (viz Algoritmus 2).

Kvantový obvod nyní obsahuje $|\psi_g\rangle$ (g-databázi) a další registry, na kterých můžeme Groverovo vyhledávání. Všimněme si, že stav $|\psi_g\rangle$ obsahuje cn nezávislých neprovázaných registrů.

2: Vytvoříme rovnoměrný superpoziční stav přes $i \in \{0,1\}^m$, tedy

$$|\psi_g\rangle \otimes \sum_{i \in \{0,1\}^m} |i\rangle\,.$$

3: Aplikujeme Groverův algoritmus pro vyhledávací prostor $i \in \{0,1\}^m$. Testovacím orákulem je zde unitární operátor test. Jeho vstupy jsou stavy (registry) $|\psi_g\rangle$, $|i\rangle$ a $|b\rangle$, který je nulový. Toto kvantové orákulum testuje, zdali má funkce $f_i \oplus g$ skrytou periodu. Pokud ano, tak $|b\rangle$ změní na stav $|b \oplus 1\rangle$. V opačném případě ke změně nedojde. Operátor test pro fixní i podrobněji popisuje Algoritmus 5. Tento operátor lze schématicky popsat jako



4: Po provedení $O(2^{m/2})$ Groverových iterací změříme registr $|i\rangle$.

Věta 4.3. Předpokládejme, že m je v $O(n)$. Nechť c je dostatečně velká konstanta. Dále uvažujme zadání problému asymptotického hledání periody, tedy nechť pro $i_0 \in \{0,1\}^m$ je funkce $f_{i_0} \oplus g$ periodická a předpoklad (4.4) platí. Potom algoritmus Alg-ExpQ1 určí hledané i_0 s pravděpodobností $\Theta(1)$ provedením $O(2^n)$ klasických dotazů na g a $O(n2^{m/2})$ kvantových (superpozičních) dotazů na F. Offline výpočet algoritmu (neuvažujeme přípravu kvantového stavu $|\psi_g\rangle$) proběhne za čas $O((n^3 + nT_f)2^{m/2})$, kde T_f je čas potřebný na jedno vyčíslení funkce F.

Algorithm 5 Procedura test

Vstup: $|\psi_g\rangle$, $|i\rangle$, $|b\rangle$
Výstup: $|\psi_g\rangle$, $|i\rangle$, $|b \vee b \oplus 1\rangle$

1: Začneme g-databází, to znamená stavem $|\psi_g\rangle$ ve tvaru

$$\bigotimes^{cn} \left(\sum_{x \in \{0,1\}^n} |x\rangle \, |g(x)\rangle \right)$$

2: Pomocí cn superpozičních dotazů na f obdržíme kvantový stav $|\psi_{f \oplus g}\rangle$ ve tvaru

$$\bigotimes^{cn} \left(\sum_{x \in \{0,1\}^n} |x\rangle \, |g(x) \oplus f(x)\rangle \right).$$

Nyní reverzibilním způsobem použijeme Simonův algoritmus, abychom zjistili, zda funkce $g \oplus f$ má skrytou periodu či nikoliv (tj. zdali funkce f a g mají skrytý posun).

3: Aplikací operátoru $(H^{\otimes n} \otimes I_m)^{cn} \otimes I_1$ na $|\psi_{f \oplus g}\rangle \otimes |b\rangle$ obdržíme stav

$$\left(\sum_{u_1, x_1 \in \{0,1\}^n} (-1)^{u_1 \cdot x_1} \, |u_1\rangle \, |(f \oplus g)(x_1)\rangle \right) \otimes \cdots$$
$$\cdots \otimes \left(\sum_{u_{cn}, x_{cn} \in \{0,1\}^n} (-1)^{u_{cn} \cdot x_{cn}} \, |u_{cn}\rangle \, |(f \oplus g)(x_{cn})\rangle \right) \otimes |b\rangle .$$

4: Spočítáme dimenzi lineární obalu vektorového prostoru generovaného vektory $u_1, \ldots, u_{cn}$, tedy $d \coloneqq \dim(\mathrm{Span}(u_1, \ldots, u_{cn}))$. Pokud $d = n$, tak $r \coloneqq 0$. Jestliže $d \leq n$, tak $r \coloneqq 1$. Výslednou hodnotu r přičteme k b pomocí bitového operátoru XOR. Poté odpočítáme d, r a obdržíme stav

$$\sum_{\substack{u_1, \ldots, u_{cn} \\ x_1, \ldots, x_{cn}}} (-1)^{u_1 \cdot x_1} \, |u_1\rangle \, |(f \oplus g)(x_1)\rangle \otimes \cdots$$
$$\cdots \otimes (-1)^{u_{cn} \cdot x_{cn}} \, |u_{cn}\rangle \, |(f \oplus g)(x_{cn})\rangle \otimes |b \oplus r\rangle .$$

Uvědomme si, že r závisí na $u_1, \ldots, u_{cn}$ a že poslední registr $|b \oplus r\rangle$ může být provázaný s registry obsajující $u_1, \ldots, u_{cn}$.

5: Odpočítáme operátor $(H^{\otimes n} \otimes I_m)^{cn} \otimes I_1$.

6: Pomocí dalších cn superpozičních dotazů na f navrátíme $|\psi_{f \oplus g}\rangle$ zpět do stavu $|\psi_g\rangle$.
Mohou nastat dva případy:

- Funkce $f \oplus g$ má skrytou periodu, v tomto případě vždy $r = 1$. Výstupní registr $|b\rangle$ je tedy ve stavu $|1\rangle$.
- Funkce $f \oplus g$ nemá skrytou periodu, v tomto případě s vysokou pravděpodobností $r = 0$. Výstupní registr $|b\rangle$ je tedy ve stavu $|0\rangle$.

4.6. Algoritmus SimQ1

Algoritmus SimQ1 je v podstatě modifikovaný Simonův algoritmus, neboť hledá skrytou periodu funkce $f_{i_0} \oplus g$.

Algorithm 6 Algoritmus SimQ1

Vstup: f_{i_0}, g
Výstup: v

1: Pomocí 2^n klasických dotazů na g vytvoříme kvantový stav $|\psi_g\rangle$ (viz Algoritmus 2).
2: Pomocí cn superpozičních dotazů na f_{i_0} obdržíme kvantový stav $\left|\psi_{f_{i_0} \oplus g}\right\rangle$ ve tvaru

$$\bigotimes^{cn} \left(\sum_x |x\rangle \, |f_{i_0}(x) \oplus g(x)\rangle \right).$$

3: Aplikací Hadamardových bran $H^{\otimes n}$ na každý registr $|x\rangle$ obdržíme stav

$$\bigotimes^{cn} \left(\sum_{x,u} (-1)^{x \cdot u} |u\rangle \, |f_{i_0}(x) \oplus g(x)\rangle \right).$$

4: Změřením všech registrů $|u\rangle$ obdržíme cn vektorů $u_1, \ldots u_{cn}$.
5: Spočítáme dimenzi d vektorového prostoru V generovaného vektory $u_1, \ldots u_{cn}$. Pokud $d \neq n-1$, tak výsledek algoritmu je neplatný. Jestliže $d = n-1$, určíme vektor $v \neq 0^n \in \{0,1\}^n$ kolmý na V.

Věta 4.4. Předpokládejme, že funkce $f_{i_0} \oplus g$ má periodu $s \neq 0^n$ a platí

$$\max_{t \neq \{s, 0^n\}} \Pr_x \left[(f_{i_0} \oplus g)(x \oplus t) = (f_{i_0} \oplus g)(x) \right] \leq \frac{1}{2}. \tag{4.7}$$

Potom Algoritmus SimQ1 vrátí s pravděpodobností alespoň $1 - 2^n \cdot (3/4)^{cn}$ námi hledané s provedením $O(2^n)$ klasických dotazů na g a cn kvantových (superpozičních) dotazů na f_{i_0}. Offline výpočet algoritmu (neuvažujeme přípravu kvantového stavu $|\psi_g\rangle$) proběhne za čas $O(n^3 + nT_f)$, kde T_f je čas potřebný na jedno vyčíslení funkce f_{i_0}.

Věta 4.5 (Odhad konstanty c)**.** V praxi je pro platnost Věty 4.3 konstanta $c \simeq m/(n \log_2(4/3))$ dostatečná, kde symbol $\simeq$ představuje asymptotickou rovnost.

Důkaz. Chceme, aby platila nerovnost $4\lfloor \pi/(4 \arcsin(2^{-m/2})) \rfloor 2^{(n+1)/2} (3/4)^{cn/2} < 1/2$. Lze tvrdit, že $\arcsin x \simeq x$ a že zaokrouhlování má zanedbatelný vliv na určení hodnoty c. Proto nerovnost upravíme na tvar $m/2 + (n+1)/2 + \log_2(\pi) + log_2(3/4)cn/2 < -1$, z níž plyne $c > \log_2(4/3)^{-1}(m + 3 + 2log_2(\pi))/n \simeq m/(n \log_2(4/3))$. □

Poznámka 4.6. Jestliže $m = n$, tak $c \simeq 2{,}5$. Pokud $m = 2n$, tak $c \simeq 5$. Tyto odhady platí pro velké hodnoty m, n.

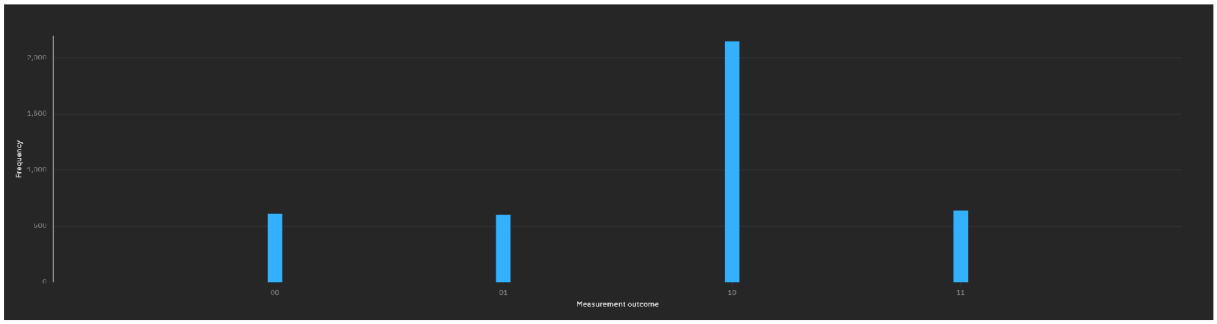
4.7. Implementace útoku

Výstupem této práce je pokus o realizaci výše popsaného útoku na Even-Mansourovo schéma, což má demonstrovat porozumění dané problematice a ukázat, že offline verzi Simonova algoritmu lze implementovat.

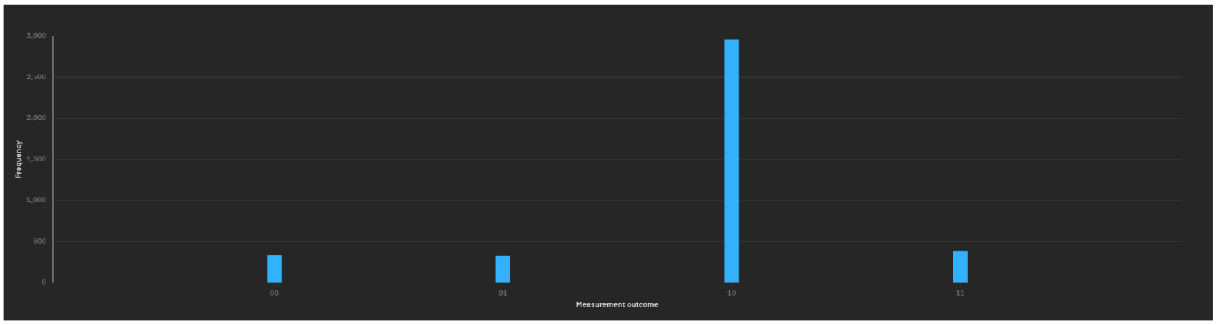
Útok provedeme konkrétně na šifru s 3-bitovými klíči k_1, k_2, kdy klíč k_1 rozdělíme na $k_1^{(1)} \in \{0,1\}$ a $k_1^{(2)} \in \{0,1\}^2$. To znamená, že $n = 3$, $u = 1$ a $m = 2$. Dále jsme experimentálně ověřili, že pro náš útok je $c \in \{2,3\}$ postačující, neboť jak bylo zmíněco dříve, odhad konstanty c (Věta 4.5) platí pro velké hodnoty m, n.

Dalším zkoumáním této instance EM schématu jsme zjistili, že zrealizovat jsme schopni pouze první krok offline verze Simonova algoritmu, tedy algoritmus Alg-ExpQ1. Jeho výstupem je $k_1^{(2)}$, což jsou 2/3 klíče k_1. Druhý krok (tzn. algoritmus SimQ1) je v tomto případě neuskutečnitelný z toho důvodu, že vektorový prostor V generovaný (jednorozměrnými) vektory $u_1, \dots u_{cn}$ je $d = 0$. Tím pádem nejsme schopni jednoznačně určit nenulový vektor v, který je na prostor V kolmý. Může nás tedy napadnout se pokusit $k_1^{(1)}$ uhodnout a na základě naší domněnky určit pomocí třetího kroku klíč k_2. Výsledkem by pak byly dvě možné varianty klíčů k_1, k_2. S pomocí těchto klíčů bychom pak mohli vytvořit dvě nové databáze funkce g, jež bychom porovnali s původní g-databází. Pokud by v jednom z případů došlo ke shodě, mohli bychom prohlásit, že jsme úspěšně nalezli klíče pro dané EM schéma. Bohužel i zde narazíme na problém, kdy nebudeme schopni určit, která dvojice klíčů je ta správná. To je způsobeno tím, že máme malou šifru, a tedy málo dat k odhadnutí $k_1^{(1)}$ a k_2.

Algoritmus Alg-ExpQ1 pro $c \in \{2,3\}$ je implementován v jazyce Python, konkrétně ve výjovovém prostředí Qiskit od společnosti IBM. Qiskit je software, který nám poskytuje sadu nástrojů a knihoven k provádění kvantových výpočtů a jejich simulací. Kvantové výpočty probíhají na kvantovém simulátoru, což je klasický počítač modelující chování ideálního kvantového počítače. Simulátor provedl v obou případech celkem 4000 měření registru $|i\rangle$, jejichž hodnoty a četnost výskytu zaznamenávají níže uvedené histogramy. Tyto grafy zaznamenávají konkrétně výsledky hledání $k_1^{(2)}$ pro $k_1 = 101$ a $k_2 = 010$. Příslušné zdrojové kódy s popisem a schémata kvantových obvodů jsou součástí přílohy.



Obrázek 4.2: Výsledky měření algoritmu Alg-ExpQ1 pro $c = 2$



Obrázek 4.3: Výsledky měření algoritmu Alg-ExpQ1 pro $c = 3$

Algoritmus Alg-ExpQ1 je v tomto případě správně implementován, jestliže hodnota $k_1^{(2)} = 01$ bude výrazně převyšovat četnost ostatních výsledků, což platí pro oba histogramy (čísla na vodorovné ose čteme zprava doleva). Tímto jsme ukázali, že algoritmus je pro $k_1 = 101$ a $k_2 = 010$ funkční. Máme experimentálně ověřeno, že implementace jsou funkční pro jakékoli kombinace k_1, k_2. Srovnáním výsledků pro obě hodnoty c si můžeme všimnout, že algoritmus pro $c = 3$ je mnohem přesnější.

Zmíníme, že oproti teoretickému popisu algoritmu došlo k malé modifikaci. Registr $|b\rangle$ byl sloučen s pomocným registrem $|-\rangle$. Tímto krokem jsme nenarušili funkčnost algoritmu. Naopak se velikost obvodu snížila o 1 kvantový bit. Kvantový obvod pro $c = 2$ tedy vyžaduje 11 qubitů a obvod pro $c = 3$ potřebuje 15 qubitů.

Závěr

Tento odborný text se čtenáři pokusil přiblížit fungování kvantového počítače a nastínil typy útoků využívajících kvantové algoritmy. Byly představeny dva kvantové algoritmy a následně algoritmus, jenž je jejich kombinací – offline verze Simonova algoritmu. Tímto algoritmem jsme se pokusili zaútočit na jednoduchý model Even-Mansourova schématu.

Během snahy o realizaci tohoto útoku jsme ale narazili na omezení, která s sebou tato jednoduhá symetrická šifra přinesla. Provedli jsme tedy alespoň dvě varianty částečného útoku, jejichž správnost jsme ověřili. Nakonec jsme jejich výsledky mezi sebou porovnali. Za úspěch lze považovat drobnou modifikaci algoritmu oproti jeho teoretickému popisu, což by nebylo možné bez hlubšího pochopení dané problematiky.

Literatura

[1] Qubit, An Intuition #2 — Inner Product, Outer Product, and Tensor Product in Bra-ket Notation | by Andi Sama | Medium. Andi Sama – Medium [online; cit. 2023-05-25]. Dostupné z: https://andisama.medium.com/qubit-an-intuition-2-inner-product-outer-product-and-tensor-product-in-bra-ket-notation-9d598cbd6bc

[2] UNLV Physics & Astronomy [online; cit. 2023-05-25]. Dostupné z: https://www.physics.unlv.edu/ bernard/phy721_99/tex_notes/node6.html

[3] Diracova notace – Wikipedie. [online; cit. 2023-05-25]. Dostupné z: https://cs.wikipedia.org/wiki/Diracova_notace

[4] Introduction to Quantum Computing | IntechOpen. IntechOpen - Open Science Open Minds | IntechOpen [online]. Copyright © 2020 The Author [cit. 2023-05-26]. Dostupné z: https://www.intechopen.com/chapters/73811

[5] Qubit – Wikipedie. [online; cit. 2023-05-25]. Dostupné z: https://cs.wikipedia.org/wiki/Qubit

[6] Qubit - Wikipedia. [online; cit. 2023-05-25]. Dostupné z: https://en.wikipedia.org/wiki/Qubit

[7] Introduction to quantum computing - GeeksforGeeks. GeeksforGeeks | A computer science portal for geeks [online; cit. 2023-05-25]. Dostupné z: https://www.geeksforgeeks.org/introduction-quantum-computing/

[8] Kvantový počítač – Wikipedie. [online; cit. 2023-05-25]. Dostupné z: https://cs.wikipedia.org/wiki/Kvantový_počítač

[9] Quantum computing - Wikipedia. [online; cit. 2023-05-25]. Dostupné z: https://en.wikipedia.org/wiki/Quantum_computing

[10] Decoherence: Quantum Computer's Greatest Obstacle | by Tanisha Bassan | Medium. Tanisha Bassan – Medium [online; cit. 2023-05-25]. Dostupné z: https://tanishabassan.medium.com/decoherence-quantum-computers-greatest-obstacle-67c74ae962b6

[11] Quantum circuit - Wikipedia. [online; cit. 2023-05-25]. Dostupné z: https://en.wikipedia.org/wiki/Quantum_circuit

[12] Vacuum-tube computer - Wikipedia. [online; cit. 2023-05-25]. Dostupné z: https://en.wikipedia.org/wiki/Vacuum-tube_computer

[13] Yes, quantum supremacy disproves the extended church-turing thesis (Bernstein-Va... | Hacker News. Hacker News [online; cit. 2023-05-25]. Dostupné z: https://news.ycombinator.com/item?id=21045616

[14] Quantum logic gate - Wikipedia. [online; cit. 2023-05-25]. Dostupné z: https://en.wikipedia.org/wiki/Quantum_logic_gate

[15] Matematická sekce | Matematicko-fyzikální fakulta Univerzita Karlova [online; cit. 2023-05-25]. Dostupné z: https://www2.karlin.mff.cuni.cz/ holub/soubory/qc/node18.html

[16] A Simple Proof that Toffoli and Hadamard are Quantum Universal - arXiv Cornell University. [online; cit. 2023-05-25] Dostupné z: https://arxiv.org/pdf/quant-ph/0301040.pdf

[17] Základy kvantového počítání - Katedra fyziky Fakulty jaderné a fyzikálně inženýrské ČVUT v Praze [online]. Copyright © [cit. 2023-05-26]. Dostupné z: https://physics.fjfi.cvut.cz/files/predmety/02UKT/Prezentace/12_-_zaklady_kvantoveho_pocitani_Deutschuv_algoritmus.pdf

[18] Church–Turing thesis - Wikipedia. [online; cit. 2023-05-25]. Dostupné z: https://en.wikipedia.org/wiki/Church–Turing_thesis

[19] Oracle machine - Wikipedia. [online; cit. 2023-05-25]. Dostupné z: https://en.wikipedia.org/wiki/Oracle_machine

[20] Quantum algorithm - Wikipedia. [online; cit. 2023-05-25]. Dostupné z: https://en.wikipedia.org/wiki/Quantum_algorithm

[21] Asymptotická složitost – Wikipedie. [online; cit. 2023-05-25]. Dostupné z: https://cs.wikipedia.org/wiki/Asymptotická_složitost

[22] Time complexity - Wikipedia.[online; cit. 2023-05-25]. Dostupné z: https://en.wikipedia.org/wiki/Time_complexity

[23] ŠKODA, Petr. Kvantové algoritmy [online]. Praha, 2007 [cit. 2023-05-25]. Dostupné z: https://www-ucjf.troja.mff.cuni.cz/cejnar/Skoda.pdf. Bakalářská práce. Univerzita Karlova v Praze, Matematicko-fyzikální fakulta. Doc. RNDr. Pavel Cejnar, Dr.

[24] Asymptotická složitost. Algoritmus [online]. Copyright © 2015 [cit. 2023-05-26]. Dostupné z: https://www.algoritmy.net/article/102/Asymptoticka-slozitost

[25] Big O notation - Wikipedia. [online; cit. 2023-05-25]. Dostupné z: https://en.wikipedia.org/wiki/Big_O_notation

[26] Landauova notace – Wikipedie. [online; cit. 2023-05-25]. Dostupné z: https://cs.wikipedia.org/wiki/Landauova_notace

[27] Asymptotická složitost | Pro vývojáře. Pro vývojáře [online; cit. 2023-05-25]. Dostupné z: https://wiki.provyvojare.cz/teorie/asymptotic-notation

[28] Basics of Quantum Computing - Asia Pacific Center for Theoretical Physics [online]. Copyright © [cit. 26-05-2023]. Dostupné z: https://www.apctp.org/temp_file/Lecture%201.pdf

[29] Simon's problem - Wikipedia. [online; cit. 2023-05-25]. Dostupné z: https://en.wikipedia.org/wiki/Simon%27s_problem

[30] Tight Bounds for Simon's Algorithm - Cryptology ePrint Archive [online]. Copyright © [cit. 26-05-2023]. Dostupné z: https://eprint.iacr.org/2020/919.pdf

[31] Quantum Period Finding against Symmetric Primitives in Practice - Cryptology ePrint Archive [online]. Copyright © [cit. 2023-05-26]. Dostupné z: https://eprint.iacr.org/2020/1418.pdf

[32] Using Simon's algorithm to attack symmetric-key cryptographic primitives - arXiv Cornell University. [online; cit. 2023-05-25] Dostupné z: https://arxiv.org/pdf/1603.07856.pdf

[33] Quantum Attacks without Superposition. Queries: the Offline Simon's Algorithm - Cryptology ePrint Archive [online]. Copyright © [cit. 2023-05-26]. Dostupné z: https://eprint.iacr.org/2019/614.pdf

[34] Security on the quantum-type Even-Mansour cipher | IEEE Conference Publication | IEEE Xplore. 301 Moved Permanently [online]. Copyright © Copyright 2023 IEEE [cit. 2023-05-26]. Dostupné z: https://ieeexplore.ieee.org/document/6400943

[35] Simon's Algorithm. [online; cit. 2023-05-25]. Dostupné z: https://learn.qiskit.org/course/ch-algorithms/simons-algorithm#4.-Oracle-

[36] Grover's Algorithm. [online; cit. 2023-05-25]. Dostupné z: https://learn.qiskit.org/course/ch-algorithms/grovers-algorithm

[37] Grover's algorithm - Wikipedia. [online; cit. 2023-05-25]. Dostupné z: https://en.wikipedia.org/wiki/Grover%27s_algorithm

[38] Symetrická šifra – Wikipedie. [online; cit. 2023-05-25]. Dostupné z: https://cs.wikipedia.org/wiki/Symetrick%C3%A1_%C5%A1ifra

Seznam příloh

1. Alg-ExpQ1_c2.ipynb – implementace algoritmu Alg-ExpQ1 pro $c = 2$
2. Alg-ExpQ1_c3.ipynb – implementace algoritmu Alg-ExpQ1 pro $c = 3$