STŘEDOŠKOLSKÁ ODBORNÁ ČINNOST

Obor 01 - Matematika a statistika

Kvantový algoritmus pro problém bezčtvercového čísla

Quantum Algorithm For Square-Free Integer Problem

Autor: Jan Bíma

Škola: Gymnázium prof. Jana Patočky
Jindřišská 36, Praha 1

Praha 2016

Prohlášení

Prohlašuji, že jsem svou práci SOČ vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v seznamu vloženém v práci SOČ.
Prohlašuji, že tištěná verze a elektronická verze soutěžní práce SOČ jsou shodné.
Nemám závažný důvod proti zpřístupňování této práce v souladu se zákonem č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) v platném znění.

V Praze dne 14. března 2016 _____

Anotace

Práce představuje komplexní, avšak snáze přístupný vhled do problematiky kvantových výpočetních systémů. Po krátkém představení základních charakteristik kvantových počítačů se autor intezivně zabývá možností, jak by bylo při využití efektivity diskutovaných systémů možné vyřešit problém bezčtvercovosti čísla. Práce prezentuje návrh takového algoritmu, který řeší problém efektivně a exaktně (řadí se tedy do složitostní třídyEQP) a který po teoretické stránce vychází ze specifických vlastností Gaussových sum. Spolu s tím autor demonstruje možnou implementaci algoritmu v jazyce *Quantum Computation Language*, kdy za tímto účelem konstruuje kvantové obvody pro výpočet největšího společného dělitele (GCD) a Jacobiho symbolu, dvou elementárních funkcí z oblasti teorie čísel.

Klíčová slova:
Kvantové počítače; Problém bezčtvercovosti; Gaussovy sumy; Simulace kvantových výpočtů; QCL

Anotation

The paper represents a complex and comprehensible insight into the emerging field of quantum computation. Along with a nutshell introduction to the particularities of quantum computers, the author pursues a way how the square-free integer problem could be solved using the capabalities of discussed computational systems. The work presents both effective and exact (EQP) algorithm designed for this purpose, which is theoretically based on the properties of quadratic Gauss sums. The author also offers an implementation of the algorithm in the *Quantum Computation Language* together with new optimalised quantum circuits for computing the Greatest Common Divisor and the Jacobi symbol, both fundamental algorithms of the number theory.

Keywords:
Quantum Computation; The Square-Free Integer Problem; Gauss Sums; Quantum Computation Simulation; QCL

Obsah

Úvod

Nastínění problematiky

Legendární Mooreův zákon říká, že každých osmnáct měsíců se zdvojnásobí počet tranzistorů, které mohou být umístěny na integrovaném obvodu. Jak se v průběhu času ukázalo, Gordon Moor se ve své predikci nemýlil. Avšak postupné zvyšování počtu elementárních součástek klade před výrobce mikročipů nejen stále obtížněji splnitelné technologické nároky, ale i problémy o poznání fundamentálnějšího rázu: Pokud by měla Moorova exponenciála zůstat i nadále v platnosti, znamenalo by to, že při postupném zvyšování hustoty interagujících součástek na čipu bude zapotřebí vzít v potaz specifické vlastnosti mikrosvěta – zákony kvantové fyziky. Dnešní architektura integrovaných obvodů však s kvantovými jevy ani zdaleka nepočítá, ba naopak, jejich potlačení se při konstrukci jeví být prvořadým úkolem.

Před vědci tak nutně vyvstává otázka, zda by nebylo možné fyzikálních vlastností mikrosvěta využít v oblastech, kde výpočetní kapacity současných počítačů již nedostačují. Na mysl nám nejspíše vytanou klasické těžké, NP problémy, kdy možnost jejich řešení v polynomiálním čase bývá označována jako jeden z „problémů tisíciletí“. Ve světle intenzivního studia kvantových systémů se však můžeme domnívat, že celé slavné $P = NP$? lze v mnohých případech elegantně „obejít“ právě skrze konstrukci kvantových počítačů, které teoreticky umožnují kvadratické až exponenciální zrychlení při řešení mnohých problémů. V této souvislosti pak bývá skloňováno zejména potenciální prolomení v současnosti asi nejrozšířenějšího šifrovacího protokolu RSA.

I přes nastíněný význam kvantových počítačů, jejichž realizace by znamenala revoluci na poli informatiky, však porozumění jejich principům zůstává i stranou odborné veřejnosti nanejvýš ojedinělou záležitostí. Je tomu tak zejména proto, že teorie kvantových počítačů v sobě spojuje hned tři vědní obory: Fyziku, z jejíchž zákonů potenciál kvantových systémů bytostně vyplývá, informatiku, která nabízí cennou inspiraci (nejen) v podobě NP pro-

blémů, především pak ale matematiku, jež je zcela nezbytným aparátem sloužícím k popisu kvantových systémů a jednotlivých algoritmů. Při studiu problematiky je nadto nutné čelit – s trochou nadsázky – až exponenciálnímu růstu potřebných znalostí, které podmiňují pochopení diskutované látky. Jakkoli se v následující práci snažíme problematiku kvantových počítačů co nejsrozumitelnějším způsobem ozřejmit, na informativní stránku neklademe takový důraz jako spíše na prezentovaný vlastní výzkum. Při bližším pohledu se totiž ukazuje, že většina českých i cizojazyčných odborných prací na toto téma se zpravidla omezuje na pouhý výklad a shrnutí dosavadních poznatků, svědky hlubšího přínosu v podobě originálního pohledu na věc se tak stáváme jen zřídkakdy.

Podobně jako je tomu v případě klasických počítačů, i pro kvantové systémy jsou projektovány algoritmy, které však budou v našem případě založeny na vlastnostech částic vyplývajících z prostředí mikrosvěta: Slovy kvantové fyziky se jedná o paralelismus, entaglement a interferenci. Tvůrci podobných algoritmů – spíše matematici než informatici – však při jejich navrhování musí čelit specifickým požadavkům, jež s sebou operování na poli mikrosvěta přináší, a to zejména v podobě nutnosti reversibility všech výpočtů, což v důsledku veškerou konstrukci do značné míry ztěžuje.

Na poli kvantových počítačů pak dominují zejména dva algoritmy, a sice Shorův a Groverův. Oba algoritmy bývají označovány jako fundamentální, neboť se stávají součástí naprosté většiny ostatních výpočetních postupů. V této souvislosti však nesmíme opomenout zmínit skutečnost, že samotný počet kvantových algoritmů zůstává poněkud omezeným, jak na to ve svém článku upozornil i Peter Shor [1]. Pro soupis naprosté většiny doposud známých kvantových algoritmů si dovolíme odkázat na stránku Quantum Algorithm Zoo.

Lze se domnívat, že nových algoritmů se nám nedostává ze dvou hlavních příčin: Kvůli snaze o nalezení algoritmu, jehož přínos by byl srovnatelný s prací Shora či Grovera, přestože brzké nalezení podobně významného algoritmu se jeví být ve skutečnosti poněkud nepravděpodobné, stejně tak můžeme hovořit o nárocích na odbornost a hlubokou znalost „vysoké“ matematiky, kdy se samotný rozbor již objeveného kvantového algoritmu mnohdy stává náplní celých vědeckých prací.

Stanovení vlastních cílů

I přesto se snažíme dokázat, že vlastní, originální výzkum na poli kvantových počítačů není jen doménou nejerudovanějších akademiků. Jak již název práce napovídá, za cíl našeho bádání jsme si zvolili problém bezčtver-

cového čísla, respektive možnost faktorizace násobku mocniny na prvočísla za využití nástrojů vyšší matematiky. Jakkoli se podobný problém může jevit triviálním, z pohledu teorie složitosti se neřadí jinam než do „obávané“ třídy NP. Rozklad čtvercového čísla je dále považován za problém svou obtížností srovnatelný s případem klasické faktorizace, na niž lze jeho řešení převést. Ačkoliv se tak nabízí využít možností shora zmiňovaného Shorova faktorizačního oalgoritmu, v následující práci se pokusíme nastínit poněkud odlišný přístup k nastíněné problematice: Vyjdeme z teorie Gaussových sum, které lze v případě kvantového počítače efektivně vyčíslit, a ukážeme, jak je možné jejich specifických vlastností využit právě pro řešení problému bezčtvercovosti. Přestože na možnost podobného řešení před námi poukázali již Jun Li, Dieter Suter et. al. v práci [2], výsledky jejich bádání představují spíše pouhé nastínění možného postupu, které je jakékoliv hlubší analýze vzdáleno.

V následující práci si tak klademe za cíl studovanou problematiku zachytit v její komplexnosti: V úvodu práce krátce představíme kvantové výpočetní systémy, a to jak po stránce příslušných matematických modelů, tak z hlediska jejich kvantově-mechanické povahy. Dále definujeme samotný problém bezčtvercovosti a osvětlíme nejen teoretická východiska, bez nichž by nebylo možné ke konstrukci algoritmu dospět, ale spolu s tím se i pokusíme algoritmus implementovat ve speciálním jazyce určeném pro simulaci kvantových systémů, a sice *Quantum Computation Language* [3]. Za tímto účelem pak vypracujeme návrh optimalizovaných kvantových obvodů pro výpočet největšího společného dělitele (dále jen GCD) a Jacobiho symbolu, na nichž je výpočet Gaussových sum bytostně založen.

Část I

Úvod do problematiky

Kapitola 1

Matematický model

Pro hlubší studium práce počítače se jako zcela nezbytné ukazuje zavést určitý abstraktní model, který bude možné zpětně vztáhnout na obecně každý výpočetní systém. Od první poloviny 20. století se tak na poli teoretické informatiky hovoří o tzv. Turingově stroji, který představuje univerzální, zjednodušenou matematickou představu chodu běžného počítače, uplatnitelnou při popisu libovolného algoritmu. V případě kvantových počítačů, které se oproti běžným výpočetním systémům vyznačují svým pravděpodobnostním charakterem, však nezbyde než spolu s deterministickým Turingovým strojem zavést i jeho pravděpodobnostní a kvantovou obdobu.

1.1 Deterministický Turingův stroj

Na klasický Turingův stroj je možné nahlédnout jako na zařízení sestávající ze tří částí: Konečného automatu, který je tvořen konečným počtem stavů, do nichž přechází na základě stavů předchozích (jedná se tudíž o deterministickou „procesorovou“ jednotku), nekonečné „pásky“, posloupnosti tvořené množinou symbolů, které DTS z pásky čte nebo je na ni zapisuje, a programu, čili sekvence příkazů, které tvoří vlastní přechodovou funkci. Formálně se DTS definuje jako šestice $M = (Q, \Sigma, \Gamma, \delta, q_0, F)$, přičemž

Q je konečná množina stavů,

Γ, $\Delta \in \Gamma$ je konečná množina páskových symbolů, kdy Δ je prázdný symbol,

$\Sigma \subseteq \Gamma \setminus \{\Delta\}, \Sigma \neq \emptyset$ je konečná množina vstupních symbolů,

$\delta : (Q \setminus F) \times \Gamma \to Q \times \Gamma \times \{L, P\}$ je přechodová funkce, kdy L znamená posun čtecí a zapisovací „hlavy“ DTS doleva, P doprava,

$q_0 \in Q$ je počáteční stav,

$F \subseteq Q$ je množina koncových stavů.

1.2 Pravděpodobnostní Turingův stroj

Koncový stav PTS je oproti tomu charakterizován svou stochastickou povahou, neboť k jednotlivým posunům „hlavy“ dochází na základě náhodného jevu. Deterministická přechodová funkce KTS je nahrazena dvěma funkcemi, μ_0, μ_1. Formálně

$\forall q \in Q$, $a \in \Gamma$ platí, že $\delta(q, a) \in \{\mu_0, \mu_1\}$, kdy $\mu_0, \mu_1 \in Q \times \Gamma \times \{L, P\}$. Přitom pravděpodobnost $P(\delta(q, a) = \mu_0) = P(\delta(q, a) = \mu_1) = \frac{1}{2}$.

V každém kroku je tak s 50% pravděpodobností zvolena jedna z přechodových funkcí, což průběh programu větví do podoby „binárního stromu“ o 2^n větvích, kde n je počtem volání přechodové funkce. Realizace PTS nalézá uplatnění zejména tam, kde existuje určitá možnost, že nedeterministický algoritmus dospěje ke správnému řešení v kratším čase než jeho klasická varianta, či tehdy, kdy postačuje zjistit výsledek jen s určitou pravděpodobností – klasickým příkladem je pak test prvočíselnosti.

1.3 Kvantový Turingův stroj

Kvantový Turingův stroj, jenž se na následujících stranách stane předmětem našeho studia, si lze nejlépe představit jako specifickou variantu diskutovaného pravděpodobnostního stroje, založenou na vlastnostech, které částice nabývají v prostředí mikrosvěta.

Hlavní odlišnosti od PTS lze shrnout do následujících bodů:

- Přechodové funkce jsou na rozdíl od čistě náhodných jevů ovlivněny kvantově-mechanickými interakcemi,
- kvantový stroj nepracuje pouze s čistými stavy „0“ a „1“, jak je známe z klasických počítačů, ale se superpozicemi těchto hodnot (nemusí se však pokaždé jednat o dvoustavový systém, jak později zmíníme),
- kvantový paralelismus (tj. superpozice) umožnuje oproti DTS i PTS pracovat s vícero stavy současně, chod stroje se tudíž exponenciálně větví při zachování lineárního času,
- samotný průběh algoritmu je deterministické povahy, avšak vzhledem k procesu měření, spjatým s kolapsem vlnové funkce, nabývá KTS pravděpodobnostního charakteru.

Po formální stránce se KTS od předchozích dvou diskutovaných Turingových strojů zároveň zcela odlišuje povahou své přechodové funkce, neboť ta je v tomto případě spíše než posunutím „hlavy“ (ono $\{L,\ P\}$) transformací vektoru v rámci komplexního Hilbertova prostoru $\mathcal{H}$ (později více ozřejmíme). Lance Fortnow [4] z tohoto důvodu zavedl pro popis obecného Turingova stroje nový formalismus, kde δ je přechodovou maticí aplikovanou na konkrétní konfiguraci Turingova stroje. Konfiguraci je možné zapsat jako uspořádanou trojici $C = (q,\ T,\ i) \in Q \times \Gamma \times \mathbb{Z}$, přičemž q je aktuálním stavem, T obsahem pomyslné „pásky“ a i pořadím její načtené části. $(q,\ T,\ i)$ následně udává globální stav celého Turingova stroje.

Pro zápis konfigurace se v případě kvantových počítačů využívá Diracovy notace: $|C\rangle = |q\rangle\,|T\rangle\,|i\rangle$. Prvotní stav KTS pak můžeme vyjádřit jako $|\psi(0)\rangle = |q_0\rangle\,|x\rangle\,|1\rangle$, kde $x \in \Sigma$ je počátečním stavem na „pásce“. Mezi stavy je možné přecházet aplikací operátoru $\hat{U}$, respektive přechodové matice U. Zatímco v případě DTS nabývají jednotlivé prvky přechodové matice hodnot z množiny $\{0,\ 1\}$, u PTS z $\mathbb{R}$, v rámci kvantového Turingova stroje operujeme na množině komplexních čísel $\mathbb{C}$. Jak v následujících kapitolách vysvětlíme, elementárním požadavkem na matici U je u KTS její unitarita, neboť musí být splněna podmínka reversibility všech operací prováděných na kvantovém jádře:

$$|\psi(n+1)\rangle = \hat{U}\,|\psi(n)\rangle \tag{1.1}$$

$$|\psi(n)\rangle = \hat{U}^{-1}\,|\psi(n+1)\rangle \tag{1.2}$$

Spolu s tím bylo dokázáno [5], že kvantový Turingův stroj je univerzální, tj. jeden kvantový počítač může být simulován druhým. Neboť zároveň principy KTS zůstávají v souladu se slabou Church-Turingovou tezí [5], je možné kvantový počítač – byť neefektivně – simulovat za pomoci klasického, deterministického Turingova stroje, jak to v rámci této práce později sami učiníme.

1.4 Kvantové složitostní třídy

Zároveň zmiňme složitostní třídy, s nimiž se u kvantových počítačů setkáváme.

BQP (bounded error quantum polynomial time) zahrnuje takové rozhodovací problémy, které jsou v polynomiálním čase řešitelné s pravdě-

podobností $P \geq \frac{2}{3}$. Jedná se tudíž o obdobu třídy BPP u pravděpodobnostního Turingova stroje.

EQP (exact quantum polynomial time) označuje takové rozhodovací problémy, které jsou v polynomiálním čase řešitelné s jistotou, tedy $P = 1$, hovořit lze o analogii s třídou P.

PostBQP (postselection bounded error quantum polynomial time) je ryze hypotetickou třídou (viz [6]) sjednocující ty problémy, které jsou na kvantovém počítači s možností postselekce řešitelné v polynomiálním čase a $P \geq \frac{2}{3}$.

QMA (quantum Merlin Arthur) je kvantovou obdobou NP, vztah BQP k QMA je stejný jako P k NP.

Kapitola 2

Kvantový stav

Přestože jsme se v minulé kapitole zabývali možností matematické definice kvantového počítače, samotný abstraktní model slouží pouze k popisu jeho práce, a tudíž nám nenabízí hlubší vysvětlení principů, na nichž jsou kvantové systémy založeny. Jak je ostatně již zřejmé, na následujících stranách nahlédneme na studovanou problematiku optikou fyziky mikrosvěta – kvantové fyziky.

Při objasňování vlastností kvantových systémů musíme mít na paměti skutečnost, že podle kodaňské interpretace se fyzikální realita skládá ze dvou „vrstev“: Makrosvěta, běžné reality, v níž platí zákony klasické fyziky, a tudíž ji vnímáme jako zcela „přirozenou“. Spolu s tím je však nutné vzít v potaz i mikrosvět, kdy hovoříme o rozměrech menších než $10^{-8}\,m$, pohybujeme se tudíž na atomární či subatomární úrovni. Tato „vrstva“ pak nepodléhá zákonům klasické, ale kvantové fyziky, nicméně vzhledem k tomu, že se jedná o realitu našimi smysly („empiricky“) běžně nepostižitelnou, se mohou mnohé kvantové zákonitosti – jakkoli pro mikrosvět zcela přirozené – jevit paradoxními.

Ačkoliv bylo právě řečeno, že realita mikrosvěta nám není přímo přístupná, toto tvrzení se nevztahuje na případné fyzikální měření. Přesto je zapotřebí uvažovat skutečnost, že procesem měření nezískáme informaci, jež by odpovídala skutečnému stavu pozorovaného systému na úrovni mikrosvěta. Zatímco při měření nabývá pozorovaná veličina pouze diskrétních, „skokových“ hodnot (energetická hodnota elektronu, spin, polarizace fotonu), pokud necháme kvantový systém nerušeně vyvíjet, jednotlivé kvantové stavy budou na úrovni mikrosvěta mezi sebou volně, spojitě přecházet, což v důsledku znamená, že v čase budou jednotlivé veličiny nabývat nekonečného množství hodnot. Spojitý, deterministický časový vývoj kvantového systému

popisuje Schrödingerova rovnice, jejímž řešením je vlnová funkce tomuto systému příslušná.

Pro kvantovou fyziku by však nikdy nebyl příznačný její pravděpodobnostní charakter, kdybychom nemohli vzápětí dodat, že celkový stav kvantového systému může sestávat z většího množství vlnových funkcí, které jsou pak takzvaně v superpozici. Skutečnost, že systém může být v několika různých stavech současně, se pak zdá být v opozici k předchozímu tvrzení, že při měření nabývají pozorované veličiny pouze diskrétních hodnot. Tento rozpor lze jednoduše vysvětlit, pokud si uvědomíme, jakým způsobem proces měření na úrovni mikrosvěta probíhá: Pro extrahování určité informace je zapotřebí k měřenému kvantovému systému vyslat částici, její interakce s pozorovaným systémem však bude mít za následek narušení křehké superpozice; hovoříme o dekoherenci a kolapsu vlnové funkce, která stochasticky přejde do jednoho z možných vlastních stavů systému. Pravděpodobnost naměření konkrétní hodnoty (tj. vlastního stavu) je pak určena druhou mocninou amplitudy pravděpodobnosti, resp. hustotou pravděpodobnosti, která vlnové funkci daného vlastního stavu náleží. Neboť pak amplituda pravděpodobnosti nabývá hodnot z oboru komplexních čísel $\mathbb{C}$, je možné ji kvantově-mechanickými interakcemi ovlivňovat, a tak manipulovat s pravděpodobnostmi naměření jednotlivých stavů. Vzhledem k tomu, že každý kvantový systém se odlišuje měřitelnými veličinami (spin, polarizace fotonu), se pak pro zobecnění pozorované vlastnosti zavádí pojem pozorovatelná.

Jak jsme již naznačili v předchozí kapitole, kvantovému stavu přísluší stavový vektor v komplexním Hilbertově prostoru $\mathcal{H}$. Spolu s tím, jak se kvantový systém podle Schrödingerovy rovnice spojitě vyvíjí, dochází v čase k transformaci odpovídajícího stavového vektoru, který může nabývat nekonečného množství hodnot. Konkrétní stav kvantového systému, resp. vektor v rámci Hilbertova prostoru, lze vyjádřit jako součet vlastních stavů (bázových vektorů) násobených komplexními koeficienty (amplitudami pravděpodobnosti), které vyjadřují pravděpodobností „zastoupení“ daného vlastního stavu ve výsledné superpozici.

2.1 Qubit

V případě kvantových počítačů pak hovoříme o obecném dvoustavovém kvantovém systému, jehož dvěma vzájemně rozlišitelným (ortogonálním) vlastním stavům (bázovým vektorům) přiřadíme logické hodnoty „0“ a „1“. Z definice pozorovatelné vyplývá, že oproti klasickým počítačům, kde jsou logické hodnoty shodně určeny jako napětí, je kvantová „0“ a „1“ pouze

zjednodušeným označením hodnot, jichž pozorovatelná může nabýt.

Podobný dvoustavový kvantový systém pak nazveme jako kvantový bit, qubit. Ve shodě s výše uvedeným lze stav qubitu vyjádřit jako lineární kombinaci dvou vlastních stavů (bázových vektorů), resp. logických hodnot jim příslušných:

$$|\psi\rangle = \omega_0 |0\rangle + \omega_1 |1\rangle \tag{2.1}$$

Přitom $|\psi\rangle \in \mathcal{H}^2$. Druhá mocnina komplexního koeficientu ω_0, $\omega_1 \in \mathbb{C}$ vyjadřuje pravděpodobnost, se kterou daný vlastní stav naměříme. Vzhledem k normalizaci musí platit $|\omega_0|^2 + |\omega_1|^2 = 1$, obecně tedy

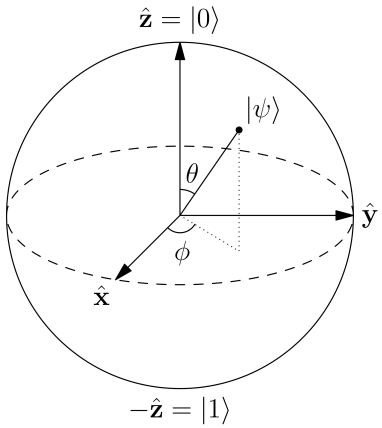
$$\sum_{i=0}^{n-1} |\omega_i|^2 = 1 \tag{2.2}$$

Diracův „ket“ $|\psi\rangle$ pro jeden qubit můžeme stejně tak zapsat jako matici, kdy jednotlivé řádky odpovídají amplitudám pravděpodobnosti daných stavů:

$$|\psi\rangle = \begin{pmatrix} \omega_0 \\ \omega_1 \end{pmatrix} \tag{2.3}$$

Celkový stav qubitu lze zobrazit jako bod na Blochově sféře, resp. na povrchu jednotkové koule $\mathbb{R}^3$. Sférické souřadnice pak udávají úhly θ, ϕ, přičemž

$$|\psi\rangle = \cos\frac{\theta}{2} |0\rangle + e^{i\phi} \sin\frac{\theta}{2} |1\rangle \tag{2.4}$$



Obrázek 2.1: Blochova sféra

Čistému stavu $|0\rangle$ odpovídá severní pól jednotkové koule, $|1\rangle$ jižní; úhel θ, který vektor svírá se svislou osou, pak vyjadřuje poměr obou vlastních stavů. Úhel ψ, o který je vektor natočen okolo svislé osy, odpovídá fázi qubitu a nabývá významu při kvantové interferenci, které se budeme věnovat později.

2.2 Kvantový registr

Samotný qubit je nicméně pouze elementární jednotkou kvantového jádra, tedy systému, na němž probíhají operace kvantového počítače. Podobně jako tomu je v případě klasických bitů, i qubity je možné uspořádat do větších celků, kdy hovoříme o takzvaných kvantových registrech. Kvantový registr o velikosti n definujme jako uspořádanou n-tici různých qubitů kvantového jádra; formálně lze registr zapsat jako direktní tenzorový součin stavových vektorů daným qubitům příslušných. Pro $n = 2$ a qubity $|\psi_A\rangle$, $|\psi_B\rangle$ pak vzhledem k rovnici 2.3 platí:

$$|\Psi_{AB}\rangle = |\psi_A\rangle \otimes |\psi_B\rangle = \begin{pmatrix} \omega_{A0} \\ \omega_{A1} \end{pmatrix} \otimes \begin{pmatrix} \omega_{B0} \\ \omega_{B1} \end{pmatrix} = \begin{pmatrix} \omega_{A0}\omega_{B0} \\ \omega_{A0}\omega_{B1} \\ \omega_{A1}\omega_{B0} \\ \omega_{A1}\omega_{B1} \end{pmatrix} = \begin{pmatrix} \omega_{00} \\ \omega_{01} \\ \omega_{10} \\ \omega_{11} \end{pmatrix} \tag{2.5}$$

Kvantové jádro se tak nachází ve stavu složeném ze dvou subsystémů (qubitů), kdy amplituda pravděpodobnosti každé ze čtyř složených hodnot odpovídá součinu komplexních koeficientů příslušných vlastních stavů výchozích qubitů. Spolu s tím je generován nový prostor $\mathbb{C}^4$, který je izomorfní prostoru Hilbertovu; bázi vzniklého prostoru lze vyjádřit jako direktní součin bázových vektorů jednotlivých qubitů. Bázové vektory a jejich zastoupení na celkovém stavu kvantového registru můžeme podobně jako v případě jediného qubitu zapsat pomocí Diracovy notace:

$$|\Psi_{AB}\rangle = \omega_{00}\,|00\rangle + \omega_{01}\,|01\rangle + \omega_{10}\,|10\rangle + \omega_{11}\,|11\rangle \tag{2.6}$$

Pokud výše uvedené vztahy zobecníme, pak pro Hilbertův prostor $\mathcal{H}$, který přísluší kvantovému registru o n qubitech s odpovídajícími prostory $\mathcal{H}_1 \ldots \mathcal{H}_n$, platí:

$$\mathcal{H} = \bigotimes_{i=1}^{n} \mathcal{H}_i \tag{2.7}$$

neboli $\mathbb{C}^{2^n}$. Samotný kvantový registr skládající se z qubitů $|\psi_1\rangle \ldots |\psi_n\rangle$ můžeme zapsat jako:

$$|\Psi\rangle = \bigotimes_{i=1}^{n} |\psi_i\rangle \tag{2.8}$$

Součet druhých mocnin amplitud pravděpodobnosti složených stavů musí být roven 1, což je v souladu s podmínkou 2.2.

Jak vyplývá z povahy obecného dvoustavového kvantového systému, kvantový registr je při zachování lineárního množství subsystémů (qubitů) schopen paralelně reprezentovat exponenciální množství informace, respektive až 2^n různých hodnot. Pokud uvážíme, že kvantově-mechanickými interakcemi je možné provádět operace nad všemi stavy složenými stavy současně, lze právě kvantový registr považovat za největší potenciál kvantových výpočetních systémů. Přesto musíme vzít v potaz skutečnost, že informace na registru uložená nám není přímo přístupná, neboť proces měření způsobí ryze pravděpodobnostní kolaps vlnových funkcí qubitům příslušných; později nicméně ukážeme, že pomocí jevu kvantové interference lze některé hodnoty „zvýraznit“, a tak podstatně zvýšit pravděpodobnost jejich naměření.

2.3 Časový vývoj systému

Jak jsme nastínili již v úvodu této kapitoly, kvantové počítače představují systémy, které jsou svou fyzikální podstatou nestacionární, kdy právě ze skutečnosti jejich časového vývoje vyplývají jistá omezení, která bude při pozdějším návrhu algoritmů zapotřebí zohlednit.

Za účelem popisu časového vývoje zavádí fyzika Schrödingerův, Heisenbergův a Diracův (reakční) obraz, nicméně vzhledem k tomu, že se jedná o ekvivalentní reprezentace, zmíníme v naší práci pouze Schrödingerův obraz, kterému mu je v literatuře věnována největší pozornost.

Spojitou časovou deterministickou evoluci uzavřeného kvantového systému popisuje nestacionární Schrödingerova vlnová rovnice ve tvaru

$$i\,\hbar\,\frac{\partial\,\Psi\,(t)}{\partial\,t} = \left(-\frac{\hbar^2}{2m}\triangle + V\,(t)\right)\Psi\,(t) \tag{2.9}$$

Pro naše účely je však vhodné provést substituci

$$|\Psi(t)\rangle = \Psi(t) \tag{2.10}$$

$$\hat{H}(t) = \left(-\frac{\hbar^2}{2m} \triangle + V\,(t) \right), \tag{2.11}$$

tak, abychom mohli rovnici zapsat jako

$$i\,\hbar\,\frac{\partial\,|\Psi(t)\rangle}{\partial\,t} = \hat{H}(t)\;|\Psi(t)\rangle\,, \tag{2.12}$$

přičemž $\hbar$ označuje Planckovu konstantu, $|\Psi(t)\rangle$ stavový vektor v čase t, kdy $|\Psi(t)\rangle \in \mathcal{H}$, a $\hat{H}$ je hamiltonián, neboli Hamiltonův hermitovský lineární operátor, který podle rovnice 2.11 odpovídá celkové energii příslušné danému kvantovému systému. Hamiltonián tak představuje informace o vlastních stavech systému, především ale reprezentuje všechny transformace, kterými kvantové jádro v čase prochází.

Předpokládejme, že hamiltonián zůstává v průběhu vývoje systému neměnným; poté lze stav kvantového jádra v čase t vyjádřit jako unitární transformaci počátečního stavového vektoru, tedy

$$|\Psi(t)\rangle = e^{-iHt/\hbar}\,|\Psi(0)\rangle \tag{2.13}$$

Pod výrazem $e^{-iHt/\hbar}$ se skrývá časově závislý unitární evoluční operátor (někdy též „propagátor“), který se zpravidla zapisuje jako $\hat{U}(t)$. Poté lze předchozí rovnici vyjádřit ve tvaru

$$|\Psi(t)\rangle = \hat{U}(t)\,|\Psi(0)\rangle\,, \tag{2.14}$$

což nás odkazuje na již dříve uvedené rovnosti 1.1 a 1.2.

Vývoj uzavřeného, izolovaného kvantového systému s časově nezávislým hamiltoniánem je tak možné popsat jako posloupnost lineárních operátorů $\hat{U}_{(t_0,t_1)} \ldots \hat{U}_{(t_{n-1},t_n)}$, kdy $\hat{U}_{(t_1,t_2)}$ představuje transformaci stavového vektoru v čase t_1 na stav v čase t_2, přičemž

$$\hat{U}_{(t_1,t_2)}\hat{U}_{(t_2,t_3)} = \hat{U}_{(t_1,t_3)} \tag{2.15}$$

$$\hat{U}_{(t_1,t_1)} = \hat{I} \tag{2.16}$$

$$\hat{U}_{(t_1,t_2)} = \hat{U}^{-1}_{(t_2,t_1)} \tag{2.17}$$

Jak vyplývá z rovnice 2.17, evoluční operátor musí splňovat podmínku unitarity, tj. pro matici operátoru příslušnou platí $UU^\dagger = I$, kde $U^\dagger$ značí matici hermitovsky sdruženou k U, I pak matici identity. Diskutovaný požadavek vyplývá z deterministické povahy Schrödingerovy rovnice, resp. nutnosti reversibility všech operací prováděných na kvantovém jádře. Tato skutečnost má své zřejmé fyzikální opodstatnění, pokud si uvědomíme, že reversibilní jsou právě takové operace, z jejichž výsledného stavu lze zpětně zkonstruovat stav počáteční: Neboť podle Landauerova principu je smazání jednoho bitu informace spojeno s vyzářením jednoho bitu entropie, tak i ztráta informace o předchozím vývoji by znamenala uvolnění energie z kvantového systému, což je ve zjevném nesouladu se základním předpokladem jeho izolovanosti.

2.4 Měření

Neméně důležitá je i problematika samotného měření, procesu, při kterém je možné z uzavřeného kvantového systému extrahovat informaci.

Uvažujme množinu P lineárních hermitovských projekčních operátorů $\hat{P}_1 \dots \hat{P}_m$, jejichž vlastní hodnoty odpovídají jednomu z možných vlastních stavů $r_1 \dots r_m$ pozorovatelné. Pro pravděpodobnost $p(r_m)$ naměření hodnoty r_m lze psát

$$p\left(r_m\right) = \langle \Psi | \, \hat{P}_m \, | \Psi \rangle \tag{2.18}$$

Po provedení měření jsou na sebe vlastní stavy $r_1 \dots r_m$ ortogonální. Vzhledem k normalizaci musí pro pravděpodobnosti naměření příslušné daným projekčním operátorům platit

$$\sum_{n \in \mu} p(r_n) = 1, \tag{2.19}$$

kdy množina μ je množinou všech naměřitelných vlastních stavů pozorovatelné.

Pokud si připomeneme, že Diracův „bra“ je „řádkou“ vlastních stavů, „ket“ „sloupcem“, což odpovídá skaláru, resp. součtu druhých mocnin amplitud pravděpodobnosti každého ze stavů

$$\langle \Psi | \Psi \rangle = \begin{pmatrix} \omega_1 & \omega_2 & \omega_3 & \omega_4 \end{pmatrix} \begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{pmatrix} = \omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2 = 1, \tag{2.20}$$

je pro globální stavový vektor kvantového systému možné napsat:

$$\langle \Psi | \Psi \rangle = 1 \tag{2.21}$$

V této souvislosti je nicméně zapotřebí zmínit, že naměřená vlastní hodnota pozorovatelné může odpovídat až n různým vlastním stavům kvantového jádra; pak hovoříme o takzvané n-krát degenerované vlastní hodnotě. Jakkoli je proces měření spjat s kolapsem vlnové funkce, tedy narušením superpozice různých hodnot, v případě naměření degenerované vlastní hodnoty zůstává systém i nadále v superpozici n vlastních stavů. Tímto způsobem lze pak „vyselektovat“ určité vlastní stavy kvantového jádra, čehož se hojně využívá při navrhování kvantových algoritmů. Fyzikální opodstatnění diskutovaného jevu spočívá ve skutečnosti, že z mikrosvěta unikla pouze taková informace, která bližší stav ostatních hodnot nespecifikuje.

2.5 Specifika kvantových počítačů

Potenciál kvantových výpočetních systémů je plně dosažitelný jen tehdy, pokud v algoritmech vhodně využijeme tří hlavních jevů, které nám fyzika mikrosvěta nabízí, a to sice možnosti superpozice částic, jejich entaglementu a interference.

2.5.1 Paralelismus

Již v sekci věnované kvantovému registru jsme nastínili, že kvantové jádro může díky vlastnostem své elementární jednotky, qubitu, reprezentovat až 2^n různých hodnot současně. Jedná se o důsledek již tolikrát diskutovaného fenoménu kvantové superpozice; z pohledu informatiky pak hovoříme o možnosti masivní paralelizace. Stejně jako je tomu v případě bitů, i nad qubity lze při použití příslušných unitárních operátorů provádět triviální operace (negace aj.) v lineárním čase $\mathcal{O}(n)$, kde n značí délku daného registru. Jak nicméně vyplývá z fyzikální podstaty kvantových systémů, každá tranformace qubitu nutně manipuluje s oběma jeho vlastními stavy zároveň, a je tudíž zřejmé, že při $\mathcal{O}(n)$ jsou ve skutečnosti prováděny operace nad všemi

2^n dílčími hodnotami zároveň ($\mathcal{O}(\log_2 2^n) = \mathcal{O}(n)$). V porovnání s klasickými počítači, které by pro provedení stejného počtu operací potřebovaly čas ($\mathcal{O}(2^n)$), se tak jedná o exponenciální zrychlení. Stejné tvrzení se pochopitelně vztahuje i na paměťovou složitost.

Je tedy zřejmé, že kvantový výpočetní systém umožňuje procházet všemi „větvemi“ programu současně, což nachází své uplatnění zejména tehdy, kdy pro řešení určitého problému neexistuje časově efektivní algoritmus či je jako v případě klasického vyhledávání zapotřebí postupně vyhodnotit vysoký počet různých stavů. Pokud budeme vycházet z předpokladu, že určitý problém (třeba právě vyhledávání) je možné převést do podoby booleánské funkce (tj. logickou hodnotou 1 označí výsledek a naopak), pak lze tuto funkci při zachování lineárního času vyčíslit na exponenciálním počtu proměnných, neboť jak bylo řečeno, jediné „zavolání“ příslušného operátoru operuje nad všemi stavy kvantového registru zároveň. Uvažujme funkci f a odpovídající unitární operátor $\hat{U}_f$, poté můžeme pro všechna x reprezentovaná kvantovým registrem psát:

$$\hat{U}_f : \ |x, 0\rangle \ \rightarrow \ |x, f(x)\rangle \tag{2.22}$$

2.5.2 Entanglement

Pojmem entanglement se rozumí propletení částic neboli taková situace, kdy jsou stavy jednotlivých subsystémů kvantového jádra vzájemně korelovány. Diskutovaný jev úzce souvisí s procesem měření, neboť naměření určitého stavu jedné propletené částice přímo ovlivní výsledný stav částice druhé. Na význam entanglementu pro kvantové výpočetní systémy snadno nahlédneme, pokud si uvědomíme, že umožňuje „sdílení“ informace mezi několika kvantovými registry; pokud budeme vycházet ze shora uvedené rovnosti 2.22, pak právě propletení částic způsobí, že při naměření určitého x zkolabuje druhý registr do odpovídající hodnoty $f(x)$.

Entanglement částic můžeme následně osvětlit na příkladu určitých stavů kvantových registrů. Mějme kvantový registr o délce $n = 2$, který není propletený. Pak lze psát

$$|\Psi\rangle = 0.5\,|00\rangle + 0.5\,|01\rangle + 0.5\,|10\rangle + 0.5\,|11\rangle \tag{2.23}$$

Je zřejmé, že nedochází ke vzájemné interakci mezi subsystémy, neboť ať již

na prvním qubitu naměříme hodnotu 0 nebo 1, obě dvě hodnoty můžeme se stejnou, tj. nezměněnou pravděpodobností naměřit i na druhé částici.

Dva entanglované qubity je oproti tomu možné zapsat jako

$$|\Psi\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle \tag{2.24}$$

nebo

$$|\Psi\rangle = \frac{1}{\sqrt{2}}|01\rangle + \frac{1}{\sqrt{2}}|10\rangle \tag{2.25}$$

Lze dovodit, že pokud připravíme jeden z výše uvedených registrů propletených qubitů, pak již po změření první částice s pravděpodobností $P = 1$ víme, v jakém stavu se nachází i druhý qubit.

Stav kvantového registru, který není entanglovaný, tj. 2.23, pak označme jako produktový, popřípadě separabilní, neboť odpovídající prostor $\mathcal{H}$ je možné rozlišit na n vzájemně nezávislých podprostorů příslušných každému ze subsystémů. Produktový stav je tak možné vyjádřit jako direktní tenzorový součin více qubitů; pro 2.23

$$|\Psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), \tag{2.26}$$

což o 2.24 ani 2.25 zjevně neplatí.

2.5.3 Interference

Bylo řečeno, že jev kvantové superpozice umožňuje vyčíslit určitou funkci na vysokém počtu proměnných při jediném „zavolání“ příslušného operátoru, doposud jsme si však nepoložili otázku, jak docílit toho, aby vlnová funkce zkolabovala ze všech možných 2^n stavů právě do hledané hodnoty, neboť je zřejmé, že v opačném případě by se celý model kvantového počítače zredukoval na pravděpodobnostní Turingův stroj.

Východisko z nastíněného problému představuje jev označovaný jako kvantová interference, kterou analogicky ke klasické interferenci rozumíme interakci mezi amplitudami pravděpodobnosti příslušnými vlnovým funkcím daného kvantového systému. Pokud se amplitudy vzájemně zesílí, hovoříme o konstruktivní interferenci; vzhledem k výše řečenému má ale největší význam desktruktivní interference, která umožňuje „shluknutí“ informace obsažené v mnoha amplitudách do jediné.

Destruktivní interference je možné dosáhnout skrze kvantovou verzi diskrétní Fourierovy transformace (což nás přímo odkazuje na obecnou povahu klasické Fourierovy transformace), popřípadě pomocí takzvané Walsh-Hadamardovy transformace, založené na specifických vlastnostech Hadamardových hradel, které představíme v následující kapitole.

Spolu s tím je však zapotřebí zmínit skutečnost, že ke zvýraznění amplitud pravděpodobnosti určitých stavů nemusí nutně vést jen nastíněné interferenční tranformace: V případě Groverova algoritmu a tzv. Groverových iterací, které slouží k podstatnému navýšení pravděpodobnosti naměření označených hodnot, se spíše než jevu interference využívá principu kvantových procházek.

Vzhledem k tomu, že předmětem našeho studia se na následujících stranách stane kvantová Fourierova transformace (dále jen QFT), si dovolíme ostatní interferenční transformace včetně Grovery iterace blíže nespecifikovat a pouze odkázat na příslušnou literaturu ([7, 8, 9]).

Kapitola 3

Kvantová hradla

Podobně jako je tomu v případě klasických počítačů, i časový vývoj kvantových výpočetních systémů představují elementární operace prováděné nad fyzickým systémem. Oproti klasickým počítačům, kdy hovoříme o integrovaných obvodech založených na manipulaci s průtokem elektrického proudu aj., nás však jejich kvantová „paralela“ staví před poněkud obtížnější problém.

Abychom mohli dosáhnout jejich potenciálu, je zapotřebí určitým způsobem kontrolovaně manipulovat s částicemi mikrosvěta, což se samo o sobě jeví být poněkud obtížným úkolem, o to složitějším, nesmí-li částice být během výpočtu vystaveny jakémukoliv vnějšímu pozorování. V souvislosti s doposud ryze experimentální realizací kvantových počítačů se pak povětšinou hovoří o kontrole za pomoci světelných záblesků (kupř. laserem), přesto však nezbývá než přihlédnout ke skutečnosti, že existuje větší množství rozlišných modelů kvantových systémů, mezi nimiž lze jmenovat adiabatický kvantový počítač či počítač založený na nukleární magnetické resonanci (Nuclear Magnetic Resonance - NMR; již v roce 2001 na něm byla provedena Shorova faktorizace čísla 15). Nastíněnou problematiku si však vzhledem k zaměření naší práce dovolíme blíže nespecifikovat a slovo ponechat spíše experimentálním fyzikům.

Vzhledem k potřebě jednotné reprezentace výpočetních postupů se stejně jako u klasických počítačů následně zavádí zjednodušená, teoretická schémata, „obvody“, které znázorňují vývoj systému v čase. V analogii s analogovými výpočetními systémy pak jednotlivé prvky obvodu nazýváme „hradly“. Kvantová hradla představují logické operace, které lze vyjádřit jako unitární matici o rozměrech $2^n \times 2^n$, kde npředstavuje počet qubitů hradlem transformovaných.

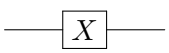
Jednoqubitová hradla

Mezi elementární jednoqubitová hradla se řadí tři Pauliho spinové matice:

Pauliho X hradlo, které otáčí stavový vektor kolem osy x, jedná se tak o transformaci stavu $|0\rangle \rightarrow |1\rangle$ a $|1\rangle \rightarrow |0\rangle$, která odpovídá klasickému NOT hradlu:

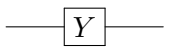
$$\hat{\sigma}_x \equiv \hat{X} \equiv NOT \equiv \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

V kvantovém obvodu lze X hradlo vyjádřit jako



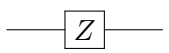
Pauliho Y hradlo, které otáčí stavový vektor kolem osy y, jedná se tak o transformaci stavu $|0\rangle \rightarrow i\,|1\rangle$ a $|1\rangle \rightarrow -i\,|0\rangle$:

$$\hat{\sigma}_y \equiv \hat{Y} \equiv \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$$



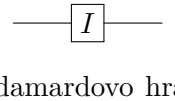
Pauliho Z hradlo, které otáčí stavový vektor kolem osy z, jedná se tak o transformaci stavu $|0\rangle \rightarrow |0\rangle$ a $|1\rangle \rightarrow -i\,|1\rangle$, která odpovídá „prohození“ fáze vektoru:

$$\hat{\sigma}_z \equiv \hat{Z} \equiv \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$



Spolu s tím definujme i matici identity pro jeden qubit:

$$\hat{I} \equiv \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

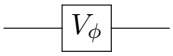


Neméně podstatné je i Hadamardovo hradlo, které umožňuje připravit vyváženou superpozici dvou vlastních stavů, tedy $|0\rangle \rightarrow \dfrac{|0\rangle + |1\rangle}{\sqrt{2}}$ a $|1\rangle \rightarrow \dfrac{|0\rangle - |1\rangle}{\sqrt{2}}$, což lze vyjádřit jako matici

$$\hat{H} \equiv \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

Dále se uvádí i obecné hradlo fázového posunu, $|0\rangle \rightarrow |0\rangle$ a $|1\rangle \rightarrow e^{i\phi}|1\rangle$, tedy

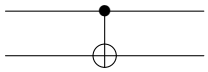
$$\hat{V}_\phi \equiv \begin{pmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{pmatrix}$$



Dvouqubitová hradla

Nejvýznamnějším dvouqubitovým hradlem je podmíněná transformace $CNOT$, která provádí operaci X na jednom qubitu, nachází-li se druhý qubit ve stavu 1. De facto je tak provedeno nedestruktivní měření, kterým je možné entanglovat dva kvantové systémy podle vztahu $|00\rangle \rightarrow |00\rangle$, $|01\rangle \rightarrow |01\rangle$, $|10\rangle \rightarrow |11\rangle$ a $|11\rangle \rightarrow |10\rangle$:

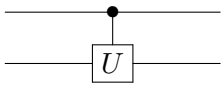
$$CNOT \equiv \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix},$$



Stojí za povšimnutí, že matici příslušnou hradlu $CNOT$ lze pak zobecnit pro každou podmíněnou dvouqubitovou transformaci jako

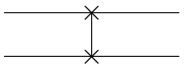
$$\hat{U}_C \equiv \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & U_{11} & U_{12} \\ 0 & 0 & U_{21} & U_{22} \end{pmatrix},$$

kde $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ je matice identity, $U_{11} \ldots U_{22}$ pak představují členy matice příslušné hradlu, které bude podmíněně aplikováno. Tedy

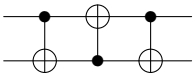


Nepodmíněnou operaci prováděnou na dvou qubitech pak představuje hradlo $SWAP$, které prohazuje pořadí bitů podle $|00\rangle \rightarrow |00\rangle$ a $|01\rangle \rightarrow |10\rangle$ a $|10\rangle \rightarrow |01\rangle$ a $|11\rangle \rightarrow |11\rangle$, vyjádřeno maticí

$$SWAP \equiv \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix},$$



což lze přepsat jako aplikaci dvou podmíněných kvantových hradel $CNOT$



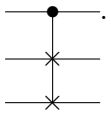
Vícequbitová hradla

Mezi tříqubitová hradla se řadí Toffoliho hradlo, někdy také $CCNOT$, které provádí operaci X na třetím qubitu, nachází-li se první dva qubitu ve stavu 1. Tedy $|000\rangle \rightarrow |000\rangle$, $|001\rangle \rightarrow |001\rangle \ldots |110\rangle \rightarrow |111\rangle$, $|111\rangle \rightarrow |110\rangle$Zapsáno maticí

$$CCNOT \equiv \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix},$$

Dále zaveďme Fredkinovo hradlo, které provádí podmíněnou $SWAP$ operaci druhého a třetího qubitu, je-li stav prvního roven 1, tudíž $|000\rangle \rightarrow |000\rangle$, $|001\rangle \rightarrow |001\rangle \ldots |110\rangle \rightarrow |101\rangle$, $|111\rangle \rightarrow |111\rangle$. Formou matice

$$CSWAP \equiv \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix},$$

 .

Za obecné n-qubitové hradlo je pak považováno i již diskutované Hadamardovo hradlo, které lze pro arbitrární velikost registru vyjádřit jako direktní tenzorový součin odpovídajícího počtu hradel

$$\hat{H}^n = \bigotimes_{i=1}^{n} \hat{H}.$$

Univerzalita kvantových hradel

Pokud v případě klasických počítačů představuje univerální hradlo logická operace $NAND$, lze ukázat, že v případě kvantových výpočetních systémů dostačuje k sestrojení libovolené funkce série Toffoliho či Fredkinových hradel, avšak vzhledem k tomu, že se jedná o operace prováděné na třech qubitech zároveň, je jejich praktická implementace pohledem současné experimentální fyziky náročná. Lze nicméně dokázat, že libovolné hradlo je v případě kvantových počítačů možné vyjádřit pouze pomocí dvouqubitového $CNOT$ a obecného jednoqubitového hradla, popřípadě ho s velkou přesností aproximovat, jak to dokazuje Solovay-Kitaevův teorém [10].

Kapitola 4

Quantum Computation Language

Jak jsme již v samotném úvodu práce nastínili, operace prováděné na kvantovém výpočetním systému budeme simulovat pomocí k tomu určeného jazyka *Quantum Computation Language* (QCL), který v rámci své magisterské práce navrhl B. Ömer z TU Wien [3]. Obsáhlou specifikaci jazyka včetně příslušného programu, který slouží jako interpret kódu a simulační prostředí zároveň, je možné najít online na adrese http://tph.tuwien.ac.at/∼oemer/qcl.html.

QCL představuje pro svou komplexnost (ačkoli se nám při práci v něm podařilo nalézt bugy, na které autora upozorníme...) jeden z nejuznávanějších simulačních jazyků na poli kvantových počítačů, z citovaných autorů ho využil kupříkladu J. Višňák pro algoritmus IPEA [9]. Svou syntaxí pak kopíruje klasické programovací jazyky, což do značné míry ulehčuje plné využití jeho potenciálu. QCL automaticky kontroluje, je-li zadaná transformace kvantového jádra unitární, dále pak plně podporuje entanglement qubitů (do fyzického výkonu příslušného klasického počítače, složitost operací přeci jenom rostě v závislosti na počtu qubitů exponenciálně) a simuluje pravděpodobnostní charakter měření za pomoci vestavěného generátoru pseudonáhodných čísel.

Pro intuitivnost QCL si dovolíme jazyk blíže nespecifikovat a pouze krátce nastínit základní strukturu kódu. Vzhledem k možnému sdílení výsledků práce s širší komunitou se budeme během programovaní držet anglického názvosloví; kompletní zdrojové kódy dále prezentovaných funkcí a procedur lze pak nalézt v příloze práce.

Demonstrace struktury jazyka QCL

```
qufunct entangle(quconst a, qureg b) // Funkce prijima dva kvantove
    registry: A typu quconst, ktere se v prubehu operace nemeni (
    obecne ho lze deklarovat jako qureg), a qureg b
{
  int i; // Deklarace promenne typu int
  for i = 0 to #b-1 { CNot(b[i], a); } // Operace CNOT podminena
      stavem qubitu A a aplikovana na qubit b[i], obecne tedy CNot(
      target, control qubit)
  // Coz lze take vyjadrit jen pomoci CNot(a, b), pouze tak
      demonstrujeme indexovani kvantoveho registru
}

procedure demonstrace()
{
  qureg a[1]; // Deklarace kvantovych registru
  qureg b[8];
  int vala; // Deklarace promenne typu int
  int valb;

  H(a); // Hadamardovo hradlo aplikovane na registr a
  X(a); // Pauliho X operace na stejnym registrem
  !X(a); // ! pred funkci znamena jeji inverzi, diky reversibilite
      operace tak dostavame stav pred pouzitim prvniho hradla X()

  entangle(a,b); // Volame funkci, ktera nam bude entanglovat registr
      a s registrem b, jde nam o pouhou demonstraci operace, kterou by
       bylo mozne jinak provest rychleji

  dump a; // Vypiseme stav registru a

  measure a,vala; measure b,valb; // Provedeme mereni na registru a i
      registru b, vysledky ulozime do prislusnych promennych typu int

  print "Namerena␣hodnota␣registru␣a:␣",vala, "␣a␣registru␣b:␣", valb;
       // vypis
}
```

Po načtení souboru se zdrojovým kódem pomocí příkazu `include "demonstrace.qcl"` a zavolání funkce `demonstrace()` dostaneme výstup, který odpovídá entanglování částic

```
: SPECTRUM a: <0>
0.5 |0>, 0.5 |1>
: Namerena hodnota registru a:  1  a registru b:  255
```

Část II

Kvantový algoritmus

Kapitola 1

Problém bezčtvercového čísla

Uvažujme číslo $N \in Z$ s prvočíselným rozkladem

$$N = p_1^{\alpha_1} p_2^{\alpha_2} \dots p_i^{\alpha_i},$$

kde $p_1 \dots p_i$ jsou prvočísla a $\alpha_1 \dots \alpha_i$ jim odpovídající exponenty. Pak každé takové N, kdy $\forall \alpha_i$, $i > 1$ platí $\alpha_i > 1$, nazvěme číslem bezčtvercovým (square-free). Důvod restrikce na $i > 1$ je zřejmý, pokud uvážíme, že každé číslo lze zapsat jako $N = a^2 b$. Problém bezčtvercového čísla tak můžeme formulovat jako ověření, zdali je a větší než 1: Pokud ano, nejedná se o bezčtvercové číslo.[1] Spolu s tím tak dodefinujme další problém, jímž bude nalezení čtvercové a bezčtvercové části čísla, tedy rozklad na již uvedený tvar

$$N = a^2 b.$$

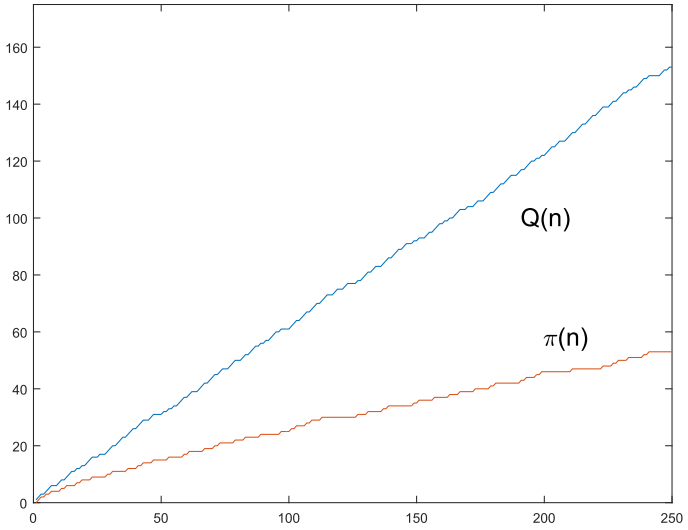
Jak je zjevné, diskutovaný problém lze převést na faktorizaci čísla, neboť známe-li prvočíselný rozklad, je samotné rozhodnutí „bezčtvercovosti“ a nalezení příslušných částí pouze triviální záležitostí. Lze se dále domnívat, že problém není o nic jednodušším než samotná faktorizace, respektive není doposud znám žádný způsob, jímž by bylo možné na deterministickém Turingově stroji v polynomiálním čase úkol vyřešit. Na tomto poznatku jsou pak založeny i některé návrhy šifrovacích protokolů. V teorii čísel pak problém nabývá významu v případě číselných sít (Number Field Sieve), respektive problém výpočtu okruhu celistvých čísel lze deterministicky polynomiálně převést právě na problém bezčtvercovosti, kdy nalezení příslušného efek-

[1] Nabízí se říci, že jde o číslo „čtvercové“, podobná formulace by však nebyla zcela přesná, protože čtvercové číslo je samotnou druhou mocninou. V anglické literatuře se nicméně můžeme setkat s výrazem „squareful“.

tivního algoritmu pro případ klasického počítače by značně zjednodušilo prolomení šifrovacího protokolu RSA.

1.1 Rozložení bezčtvercových čísel

Jak lze ukázat, rozložení bezčtvercových čísel vykazuje oproti prvočíslům větší pravidelnost; vzhledem k určité provázanosti obou případů je následně možné této vlastnosti využít pro některé důkazy ohledně prvočísel. Označme $Q(n)$ počtem bezčtvercových čísel a $\pi(n)$ počtem prvočísel menších než dané n, pak situaci pro $n < 250$ ilustruje následující graf



Obrázek 1.1: Rozložení bezčtvercových čísel

Uvažujme Möbiovu multiplikativní funkci $\mu(n)$ definovanou jako

$$\mu(n)\begin{cases}1 & \text{pro } n=1\\ 0 & \text{pokud } a^2 \mid n \text{ pro } a>1\\ (-1)^r & \text{pro } n \text{ s } r \text{ různými děliteli,}\end{cases}$$

pak lze pro $Q(n)$ psát $Q(n) = \sum_{k=1}^{n-1} |\mu(k)|$. Asymptotickou hodnotu tohoto

výrazu je možné aproximovat jako

$$Q(n) = \frac{6n}{\pi^2} + \mathcal{O}(\sqrt{n}),$$

což nás v případě asymptotické hustoty výrazu zpětně dovádí k prvočíslům, povšimneme-li si, že $\frac{6}{\pi^2} = \frac{1}{\xi(2)} \approx 0.607$, kde $\xi(2)$ není ničím jiným než hodnotou Riemannovy zeta funkce v bodě 2.[11]

1.2 Teoretické řešení problému

Jak je již z předchozího výkladu zjevné, při hledání východiska pro řešení nastíněného problému nezbyde než využít nějakého ze specifických vztahů z oboru teorie čísel.

Uvažujme kvadratický Gaussův součet $G(a, \chi)$ jako specifický konečný součet komplexních jednotek

$$G(a, \chi) = \sum_{m=0}^{N-1} e^{2\pi i a m^2/N} \tag{1.1}$$

Gaussův součet definujme na zbytkové třídě mod N pro celá a, N, kdy $N > 1$; pro případ, že N je prvočíslem splňujícím podmínku $N \nmid a$, lze psát

$$G(a, \chi_N) = \sum_{m=0}^{N-1} \chi_N(m) e^{2\pi i a m/N}. \tag{1.2}$$

χ je pak Dirichletův charakter modulo N. Z definice Dirichletova charakteru vyplývá, že pro prvočíselné N je χ rovno příslušnému Legendreho symbolu, pro N libovolné liché pak symbolu Jacobiho. Připomeňme, že Legendreho symbol je významná číselnoteoretická multiplikativní funkce,

$$\left(\frac{a}{p}\right) \begin{cases} 0 & \text{pokud } a \equiv 0 \, mod \, p \\ 1 & a \text{ je kvadratický zbytek mod } p, \text{ zároveň } a \nmid d \\ -1 & \text{není kvadratický zbytek mod } p. \end{cases}$$

Pro případ, kdy N není prvočíslem, pak zaveďme zobecněný Jacobiho symbol, který je součinem symbolů Legendreho pro jednotlivé faktory čísla N, tedy

$$\left(\frac{a}{n}\right) = \left(\frac{a}{p_1}\right)^{\alpha_1} \left(\frac{a}{p_2}\right)^{\alpha_2} \left(\frac{a}{p_3}\right)^{\alpha_3} \cdots \left(\frac{a}{p_i}\right)^{\alpha_i}.$$

Pro zjednodušení následujícího zápisu si dovolíme zavést nové označení Gaussových sum, kdy $g(a, N)$ bude odpovídat shora definované sumě $G(a, \chi_N)$ na třídě mod N. Pak uvažujme N takové, které je součinem nesoudělných čísel p, q, a celé kladné číslo a; vzhledem k multiplikativnímu charakteru Gaussových sum lze psát

$$g(a, pq) = g(pa, q)g(qa, p) \qquad (p, q) = 1. \tag{1.3}$$

Pro elaborovaný důkaz tohoto tvrzení za využití zákona kvadratické reciprocity si dovolíme pouze skromně odkázat na příslušnou literaturu [12, strana 15].

Na následujících řádcích pak formulujme čtyři věty, jejichž prostřednictvím bude možné ke shora vytyčenému problému přistoupit.

Poznámka.

Výraz 1.2 je podle definice výše ekvivalentní základnímu tvaru kvadratické Gaussovy sumy právě tehdy, kdy N je prvočíslem a zároveň $N \nmid a$. Pro účely naší práce si nicméně dovolíme definici rozšířit i pro N složené a mající s číslem a společného dělitele. Jakkoli se podobný krok může zdát poněkud pochybným, zmiňme, že H. Cohen v [13, str. 31] definuje Gaussovu sumu i pro „not necessarily primitive“ Dirichletův charakter modulo N. Po zbytek práci si tudíž dovolíme uvažovat pouze formu sumy podle 1.2; je zjevné, že multiplikativnost bude i nadále zachována.

Věta I. Pro každé bezčtvercové N a číslo a takové, že $(a, N) > 1$, je Gaussova suma rovna $G(a, \chi_N) = 0$.

Důkaz. Uvažujme N s jeho prvočíselným rozkladem $N = pq$; pokud $(a, N) > 1$, pak zjevně $p \mid a$ nebo $q \mid a$. Zároveň platí, že $(p, q) = 1$, což nás přivádí k výše uvedenému vztahu 1.3. Gaussovu sumu lze pak vzhledem k jejímu multiplikativnímu charakteru vyjádřit jako

$$g(a, pq) = g(aq, p)g(ap, q) = \left(\sum_{m=0}^{p-1} \chi_p(m) e^{2\pi i q a m/p} \right) \left(\sum_{m=0}^{q-1} \chi_q(m) e^{2\pi i p a m/q} \right),$$

Bez újmy na obecnosti pak předpokládejme, že $p \mid a$; následně lze sumu $g(aq, p)$ upravit na

$$g(aq,p) = \sum_{m=0}^{p-1} \chi_p(m) e^{2\pi i q a m/p} = \sum_{m=0}^{p-1} \chi_p(m) e^{2\pi i q m}.$$

Zjevně $e^{2\pi i q m} = 1$ pro každé $q,\ m \in N$, tedy

$$g(aq,p) = \sum_{m=0}^{p-1} \chi_p(m).$$

Dále připomeňme, že Dirichletův charakter $\chi_p(m)$ je roven symbolu Jacobiho $\left(\frac{m}{p}\right)$, resp. symbolu Legendreho vzhledem k prvočíselnosti p, a ve shodě se zákonem kvadratické reciprocity definujme následující vztah

$$\left(\frac{m}{p}\right) = -\left(\frac{p-m}{p}\right).$$

Uvedená rovnost je zřejmá: Uvažujme m takové, že $m \equiv x^2 \bmod p$, kdy podle definice Legendreho symbolu $\left(\frac{m}{p}\right) = 1$. Pak $p - m \not\equiv x^2 \bmod p$, neboli $m - p \equiv -(p-m) \equiv x^2 \bmod p$. K výsledku lze dospět i prostou úvahou, pokud si uvědomíme, že každé prvočíslo má podle rovnosti $a^2 \equiv (p-a)^2 \bmod p$ právě $\frac{p-1}{2}$ kvadratických zbytků.

Z uvedených vztahů je zjevné, že na zbytkové třídě mod p je součet Legendreho symbolů roven 0: Pro $\left(\frac{0}{p}\right) = 0$; Legendreho symbol $\left(\frac{m}{p}\right)$ se následně sečte se symbolem $\left(\frac{p-m}{p}\right)$ tak, že $\left(\frac{m}{p}\right) + \left(\frac{p-m}{p}\right) = 0$. Připomeňme výše uvedené

$$g(aq,p) = \sum_{m=0}^{p-1} \chi_p(m) = \sum_{m=0}^{p-1} \left(\frac{m}{p}\right),$$

pak tedy nutně $g(aq,p) = 0$. Tím je tvrzení dokázáno.

Věta II. Pro každé N, kterému odpovídá prvočíselný rozklad $N = pq^2$ při zachování výše uvedené podmínky $q \neq 1$, a číslo a takové, že $(a,q) = 1$, je Gaussova suma rovna $G(a,\chi_N) = 0$.

Důkaz. Z definice vyplývá $(p,q^2) = 1$, analogicky k předchozímu případu lze tak sumu vyjádřit jako

$$g(aq^2,p)g(ap,q^2)=\left(\sum_{m=0}^{p-1}\chi_p(m)e^{2\pi iq^2am/p}\right)\left(\sum_{m=0}^{q^2-1}\chi_{q^2}(m)e^{2\pi ipam/q^2}\right).$$

Dirichletův charakter $\chi_N(m)$ odpovídá Jacobiho symbolu, respektive součinu symbolů Legendreho pro jednotlivá čísla v prvočíselném rozvoji N; pak je možné psát $\chi_{q^2}(m)=\left(\frac{m}{q}\right)\left(\frac{m}{q}\right)=\left(\frac{m}{q}\right)^2$. Pokud $(m,q)>1$, platí $\chi_{q^2}(m)=0$, v opačném případě se vzhledem k právě uvedému vztahu situace zjednodušuje na $\chi_{q^2}(m)=1$.

Spolu s tím připomeňme, že součet komplexních jednotek přes zbytkovou třídu mod g je vždy roven 0, tedy

$$\sum_{m=0}^{g-1}e^{2\pi im/g}=0.$$

Výraz lze následně zobecnit pro každé celočíselné d takové, že $g\nmid d$, zřejmě tak platí

$$\sum_{m=0}^{g-1}e^{2\pi imd/g}=0.$$

Zdůrazněme, že $d\in N$, $g\nmid d$. Znovu pak uveďme tvar diskutované druhé Gaussovy sumy

$$g(ap,q^2)=\sum_{m=0}^{q^2-1}\left|\chi_{q^2}(m)\right|e^{2\pi ipam/q^2}$$

Vzhledem k výše řečenému nabývá Dirichletův charakter $\chi_{q^2}(m)$, resp. ekvivalentně $\left|\chi_{q^2}(m)\right|$ hodnot 0 nebo 1; má-li být výsledná suma následně rovna $g(ap,q^2)=0$, je nutné dokázat, že případy $\chi_{q^2}(m)=0$ konečný součet nijak neovlivní. Nejprve připomeňme, že $\chi_{q^2}(m)=0$ nastává pouze tehdy, kdy $(m,q)>1$, vzhledem k prvočíselnosti q se jedná o každou q-periodu.

Dále postupujme sporem: Předpokládejme, že součet přes q-periody se na konečné sumě podílí, pak zřejmě $\sum_{m=0}^{q-1}e^{2\pi iap(mq)/q^2}\neq 0$. Nicméně platí

$$\sum_{m=0}^{q-1}e^{2\pi iap(mq)/q^2}=\sum_{m=0}^{q-1}e^{2\pi iapm/q},$$

což nás na základě shora uvedených vztahů dovádí ke kýženému sporu.

Ukazuje se proto, že $\sum_{m=0}^{q-1} e^{2\pi iap(mq)/q^2} = 0$, a diskutovaná Gaussova suma $g(ap, q^2)$ je pro $(a, q) = 1$ ekvivalentní výrazu

$$g(ap, q^2) = \sum_{m=0}^{q^2-1} e^{2\pi ipam/q^2} = 0.$$

Tím je shora uvedené tvrzení dokázáno.

Spolu s tím považujeme za vhodné objasnit dva další případy, které mohou u Gaussových sum při řešení diskutovaného problému nastat.

Věta III. Pro každé N s prvočíselným rozvojem $N = pq^2$, kde $q \neq 1$, a číslo a takové, že $(a, q) > 1$, bude pro Gaussovu sumu platit $G(a, \chi_{pq^2}) \neq 0$.

Důkaz. Postupujme obdobně jako u předchozí věty. Na základě 1.3 lze psát $g(a, pq^2) = g(aq^2, p)g(ap, q^2)$, kdy druhou sumu $g(ap, q^2)$ můžeme vyjádřit jako

$$g(ap, q^2) = \sum_{m=0}^{q^2-1} \chi_{q^2}(m) e^{2\pi ipam/q^2}.$$

Z formulace věty vyplývá $q \mid a$ a analogicky k předcházejícímu důkazu platí vztah $\chi_{q^2}(m) = \left|\chi_{q^2}(m)\right|$, tedy

$$g(ap, q^2) = \sum_{m=0}^{q^2-1} \left|\chi_{q^2}(m)\right| e^{2\pi ipm/q}.$$

Podle shora uvedených vztahů se jedná o q-násobný součet součtu q komplexních jednotek; pro každé $m > q$ lze tak psát $e^{2\pi ipm/q} = e^{2\pi ip(xq+y)/q}$, kde $m = xq + y$. x vyjadřuje v pořadí x-tou periodu, pro y pak položme rovnost $y = m \bmod q$. Pak platí

$$e^{2\pi ip(xq+y)/q} = e^{2\pi ipxq/q} e^{2\pi ipy/q},$$

kde $e^{2\pi ipxq/q} = e^{2\pi ipx} = 1$. Pro nultou periodu zřejmě také $e^0 = 1$. Na základě toho lze sumu zapsat jako

$$\sum_{m=0}^{q^2-1} \left|\chi_{q^2}(m)\right| e^{2\pi ipm/q} = \sum_{n=0}^{q-1} \left(\sum_{m=0}^{q-1} \left|\chi_{q^2}(m)\right| e^{2\pi ipm/q} \right),$$

podle očekávání tak dostáváme q-násobný součet součtů komplexních jednotek, a to opět s problematickým případem pro $\chi_{q^2}(m) = 0$. Protože $\chi_{q^2}(0) = 0$, lze sumu vzhledem k předchozím vztahům vyjádřit jako

$$\sum_{n=0}^{q-1} \left(\sum_{m=1}^{q-1} e^{2\pi ipm/q} \right).$$

Pokud v případě sumy komplexních jednotek platí $\sum_{m=0}^{g-1} e^{2\pi im/g} = 0$, pak pochopitelně $\sum_{m=1}^{q-1} e^{2\pi ipm/q} \neq 0$, resp. $\sum_{m=1}^{q-1} e^{2\pi ipm/q} = -1$.

Neboť podle definice Gaussových sum nutně platí $g(aq^2, p) \neq 0$ pro každé $(a, p) = 1$, je tvrzení dokázáno.

Věta IV. Pro každé N s prvočíselným rozvojem $N = pq^2$, kde $q \neq 1$, a číslo a takové, že $(a, p) > 1$, bude Gaussova suma $G(a, \chi_{pq^2}) = 0$.

Důkaz. Triviální. Stejně jako v předchozích případech platí $g(a, pq^2) = g(aq^2, p)g(ap, q^2)$. Neboť $(aq^2, p) > 1$, pak podle první věty $g(aq^2, p) = 0$. Tím je tvrzení dokázáno.

Jakkoli je předchozí věta zřejmá, při studiu problematiky jsme se opakovaně setkali s tvrzením, že pro N, které není bezčtvercovým, a obecné a takové, že $(a, N) > 1$, platí pro Gaussovu sumu $G(a, \chi_{pq^2}) \neq 0$. Ověřit nesprávnost tohoto tvrzení se nám nicméně podařilo dosáhnout i poněkud „experimentálním“ způsobem, a to za pomoci vyčíslení v prostředí programu MATLAB; vzniklé chyby u jiných autorů přičítáme nedbalosti, neboť podobná tvrzení byla ponejvíce uváděna bez hlubšího důkazu, popřípadě s důkazem vycházejícím z nesprávně použitých výchozích vztahů a definic.

Výše uvedené věty lze následně zobecnit do čtyř, respektive s přihlédnutím k definici Gaussových sum pěti základních vztahů, které se pro řešení nastíněného problému ukazují jako zcela zásadní:

$$N = pq \; : \qquad G(a, \chi_N) \neq 0 \qquad (a, N) = 1 \tag{1.4}$$

$$N = pq \; : \qquad G(a, \chi_N) = 0 \qquad (a, N) > 1 \tag{1.5}$$

$$N = pq \ : \qquad G(a, \chi_N) \neq 0 \qquad (a, N) = 1 \tag{1.6}$$

$$N = pq^2 \ : \qquad G(a, \chi_N) = 0 \qquad (a, N) = 1 \tag{1.7}$$

$$N = pq^2 \ : \qquad G(a, \chi_N) = 0 \qquad (a, p) > 1 \tag{1.8}$$

$$N = pq^2 \ : \qquad G(a, \chi_N) \neq 0 \qquad (a, q) > 1 \tag{1.9}$$

1.3 Kvantový algoritmus

Vzhledem k výše uvedeným vztahům se ukazuje, že Gaussovy sumy představují při vyčíslení pro všechna $a < N$ možnost, jak lze rozhodnout, zdali je číslo N bezčtvercové, v opačném případě pak vedou k zjištění faktoru jeho čtvercové části. Uvažujme tedy algoritmus, který v polynomiálním čase vyčíslí diskutované sumy na všech proměnných $a < N$, a jakési „orakulum“, které ukáže na takovou Gaussovu sumu, pro kterou platí $G(a, \chi_N) \neq 0$, pak výstup podobného algoritmu, tj. číslo a, představuje deterministické polynomiální řešení diskutovaného problému.

Hovoříme-li o možnosti efektivního určení hodnot dané funkce na celém jejím intervalu, tj. na exponenciálním počtu proměnných vzhledem k bitové velikosti vstupu, je zřejmé, že efektivní řešení podobného, na klasickém Turingově stroji neschůdného problému představují právě kvantové výpočetní systémy. Před námi tak nyní vyvstává nelehký úkol v podobě navržení takového kvantového algoritmu, který bude i přes určitá omezení plynoucí z povahy kvantových systémů nutně zachovávat nezbytnou matematickou korektnost, a tak povede k zamýšlenému řešení problému bezčtvercového čísla.

Pro komplexnost studovaného algoritmu si dovolíme celý výpočet rozlišit na tři části, kde smysl jednotlivých kroků ozřejmíme nejen z matematického, ale i kvantově-mechanického hlediska.

Část I. Na cestě k Jacobiho symbolu Na vstupu algoritmu předpokládejme celé liché číslo N; pak alokujme dva nulové kvantové registry A, B o délce $n = \lceil \log_2 N \rceil$. Stav složeného systému označme jako $|\Psi\rangle$, pak lze psát

$$|\Psi\rangle_0 = |0\rangle_A^{\otimes n} \otimes |0\rangle_B^{\otimes n} \, . \tag{1.10}$$

Následně uvedeme registr A do vyvážené superpozice stavů $m < N$, neboli[2]

$$|\Psi\rangle_1 \to \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} |m\rangle_A |0\rangle_B . \tag{1.11}$$

Dále je zapotřebí vyselektovat takové stavy, pro které platí $(m, N) > 1$; doposud samostatné registry tudíž pomocí binárního algoritmu pro výpočet GCD entanglujeme tak, aby každému m odpovídající stav registru B obsahoval (m, N), tedy

$$|\Psi\rangle_2 \to \hat{U}_{GCD} |\Psi\rangle_1 = \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} |m\rangle_A |(m, N)\rangle_B . \tag{1.12}$$

Spolu s tím připomeňme Eulerovu funkci $\varphi(n)$, která udává počet čísel menších než n takových, že jsou s n nesoudělná. Podle definice platí

$$\varphi(n) = n \prod_{p|n} \left(1 - \frac{1}{p}\right).$$

Stejně jako tomu bylo v případě Gaussovy sumy, i Eulerova funkce je funkcí multiplikativní, tedy $\varphi(mn) = \varphi(m)\varphi(n)$ pro $(m, n) = 1$. Pro případ $\varphi(m^k)$, kde m je prvočíslo, můžeme následně psát $\varphi(m^k) = m^k - m^{k-1}$.

Eulerovy funkce lze v našem případě využít k odhadu pravděpodobnosti, s jakou se kvantový registr B nachází ve stavu 1, resp. $(m, N) = 1$. Uvažujme bezčtvercové N s prvočíselným rozvojem $N = pq$, pak zřejmě platí $\varphi(pq) = \varphi(p)\varphi(q)$. Buď P pravděpodobností, že $(m, N) = 1$ pro $m < N$; pro prvočísla p a q můžeme P vyjádřit jako

$$P_0 = \frac{\varphi(N)}{N} = \frac{\varphi(p)\varphi(q)}{pq} = \frac{(p-1)(q-1)}{pq}.$$

Pokud N není bezčtvercovým, tedy $N = pq^2$, lze podle shora uvedených vztahů psát

$$P_1 = \frac{\varphi(N)}{N} = \frac{\varphi(p)\varphi(q^2)}{pq} = \frac{(p-1)(q^2-q^1)}{pq^2}.$$

[2] Jak později ozřejmíme, transformace lze dosáhnout pomocí N-dimenzionální kvantové Fouerieovy transformace, která provede příslušnou konstruktivní interferenci. V případě implementace však aplikujeme nejprve n dimenzionální Hadamardovo hradlo, kterým registr uvedeme do superpozice všech 2^n stavů, jež následně vyselektujeme. V případě ryze teoretického popisu si však dovolíme popsat ideální stav, ze kterého následně vyplynou příslušné charakteristiky algoritmu.

Jakkoli faktorizaci čísla N v tuto chvíli neznáme, a tak nemůžeme zastoupení nesoudělných m určit, v obou případech platí

$$\lim_{N\to\infty}(P_0) = \lim_{N\to\infty}(P_1) = 1.$$

Pravděpodobnost případu $(m, N) = 1$ se tudíž s rostoucím N zvyšuje, čehož využijeme v dalším kroku diskutovaného algoritmu.

Proveďme tak měření na kvantovém registru B, kdy vzhledem k právě řečenému a již dříve nastíněné povaze kolapsu vlnové funkce přejde B s největší pravděpodobností do stavu $|1\rangle_B$. Vlastní hodnota příslušná $|1\rangle_B$ je nicméně degenerovaná; respektive byla naměřena pouze informace $(m, N) = 1$, což znamená, že stejně jako v případě předchozího měření zůstává kvantový systém ve vyvážené superpozici všech m, která jsou s N nesoudělná. Tudíž

$$|\Psi\rangle_2 \to \frac{1}{\sqrt{\varphi(N)}} \sum_{(m,N)=1} |m\rangle_A |1\rangle_B . \tag{1.13}$$

S ohledem na další kroky algoritmu je vhodné jednoqubitovým Pauliho-X hradlem vynulovat registr B, tedy

$$|\Psi\rangle_3 \to \frac{1}{\sqrt{\varphi(N)}} \sum_{(m,N)=1} |m\rangle_A |0\rangle_B . \tag{1.14}$$

V případě, že registr B do stavu $|1\rangle_B$ nezkolabuje, pak bude naměřená hodnota odpovídat $(m, N) > 1$; je-li $(m, N) \neq N$, dostáváme netriviální faktor N. Pokud na základě znalosti společného dělitele nelze problém bezčtvercovosti stále vyřešit, je po příslušné redukci N zapotřebí celý algoritmus opakovat.

Část II. Výpočet Jacobiho symbolu

Uvažujme kvantový operátor $\hat{U}_J$ takový, který odpovídá booleánské funkci s předpisem

$$f_J(m) \begin{cases} 0 & m \text{ je kvadratickým zbytkem modulo } N \\ 1 & m \text{ není kvadratickým zbytkem modulo } N \end{cases}$$

Po aplikaci operátoru lze stav systému vyjádřit jako

$$|\Psi\rangle_4 \to \hat{U}_J |\Psi\rangle_3 = \frac{1}{\sqrt{\varphi(N)}} \sum_{(m,N)=1} |m\rangle_A |f_J(m)\rangle_B . \tag{1.15}$$

Kvantový systém v tuto chvíli obsahuje dostatek logických informací na to, abychom mohli určit Jacobiho symbol pro každé naměřené m, vzhledem k dalším výpočtům je však zapotřebí informaci z registru B převést do A tak, že koeficient před každým m bude odpovídat příslušné numerické hodnotě $(\frac{m}{N})$. Proveďme tedy fázový posun na registru B o $\phi = \pi$ za pomoci jednoqubitového Pauliho-Z hradla; pak lze psát

$$\hat{Z}\,|f_j(m)\rangle_B = (-1)^{f_J(m)}\,|f_j(m)\rangle_B\,. \tag{1.16}$$

Stav kvantového systému můžeme následně vyjádřit ve tvaru

$$|\Psi\rangle_5 \to \frac{1}{\sqrt{\varphi(N)}} \sum_{(m,N)=1} (-1)^{f_J(m)}\,|m\rangle_A\,|f_J(m)\rangle_B\,, \tag{1.17}$$

registr B posléze dealokujme,[3] tedy

$$|\Psi\rangle_5 = \frac{1}{\sqrt{\varphi(N)}} \sum_{(m,N)=1} (-1)^{f_J(m)}\,|m\rangle_A\,. \tag{1.18}$$

Amplituda pravděpodobnosti před každým m nyní odpovídá příslušnému Jacobiho symbolu: V zápisu uvedeném výše platí $(m, N) = 1$, pak tedy $(\frac{m}{N}) = \pm 1$, nicméně registr latentně obsahuje informaci také o stavech, kdy $(m, N) > 1$, tedy $(\frac{m}{N}) = 0$. Díky předchozí redukci superpozice je v těchto případech amplituda pravděpodobnosti nulová, z čehož vyplývá možnost reformulace zápisu na

$$|\Psi\rangle_5 = \frac{1}{\sqrt{\varphi(N)}} \sum_{m=0}^{N-1} \left(\frac{m}{N}\right) |m\rangle_A\,. \tag{1.19}$$

Tím je Jacobiho symbol pro všechna $m < N$ vyčíslen.

Část III. Vyčíslení Gaussovy sumy

Pokud jsme se již v sekci věnované kvantové interferenci zmínili o diskrétní Fourierově transformaci coby fundamentální součásti většiny kvantových algoritmů, na následujících řádkách ukážeme, že nejinak tomu bude i v našem případě. Nejprve připomeňme tvar Gaussovy sumy

[3] De facto je nutné hodnotu $f_J(m)$ „odpočítat“, čehož lze dosáhnout další aplikací příslušného operátoru $\hat{U}_J$, neboť $|f_J(m) \otimes f_J(m)\rangle = |0\rangle$. Koeficient před m zůstane i nadále zachován.

$$G(a, \chi_N) = \sum_{m=0}^{N-1} \chi_N(m) e^{2\pi i a m/N},$$

a spolu s tím zaveďme kvantovou Fourierovu transformaci (dále jen QFT) jako kvantovou obdobu diskrétní Fourierovy transformace (DFT). Zatímco DFT je transformací prováděnou nad sekvencí komplexních čísel, QFT pak představuje lineární transformaci stavových vektorů kvantového systému. Na povahu Fouerieovy transformace lze nahlédnout jako na mapování časové domény určité funkce na její frekvenční spektrum, což v případě kvantových algoritmů nachází uplatnění pokaždé, kdy je vzhledem k možnosti efektivního vyčíslení funkcí na rozsáhlém intervalu zapotřebí vyšetřit jejich periodicitu. Právě pro hledání příslušné periody se QFT využívá kupříkladu v algoritmu hledání řádu, resp. Shorově algoritmu, spolu s tím pak můžeme zmínit i algoritmus odhadu fáze [9]. Především je pak na kvantovém počítači možné QFT provést efektivně, a to v čase $\mathcal{O}(n^2)$ oproti $\mathcal{O}(n\, 2^n)$ v případě klasického počítače.[4] Pro bližší osvětlení významu QFT v kvantových algoritmech si dovolíme odkázat na příslušnou literaturu [8, 7] a konečně uvést samotný předpis transformace, tedy

$$\hat{U}_{QFT} \left|x\right\rangle = \frac{1}{\sqrt{2^n}} \sum_{a=0}^{2^n-1} \left(e^{2\pi i a x/2^n}\right) \left|a\right\rangle . \tag{1.20}$$

Jakkoli jsme se několik stránek nazpět intenzivně zabývali teorií Gaussových sum, opomněli jsme zmínit nadmíru zajímavou skutečnost, a sice že Gaussovy sumy úzce souvisí s diskrétní (kvantovou) Fourierovou transformací, konkrétně jsou pak stopou matice DFT, v určitém smyslu i její vlastní funkcí, kdy pro bližší ozřejmění těchto tvrzení odkážeme na matematicky nadmíru zajímavý článek dostupný online na adrese The Sign of The Gauss Sum. Podobnost QFT a Gaussových sum je nicméně zřejmá již z porovnání obou předpisů.

Kvantový systém jsme zanechali ve tvaru

$$\left|\Psi\right\rangle_5 = \frac{1}{\sqrt{\varphi(N)}} \sum_{m=0}^{N-1} \left(\frac{m}{N}\right) \left|m\right\rangle_A , \tag{1.21}$$

a jak je již z předchozích řádků již zřejmé, v dalším kroku algoritmu provedeme nad registrem A právě kvantovou Fourierovu transformaci. Přestože

[4] Nejrychlejší algoritmus pro výpočet DFT představuje v současnosti rychlá Fouerierova transformace (FFT) s časovou složitostí $\mathcal{O}(n \log N)$. Pokud má být nicméně tranformace provedena nad všemi 2^N stavy, jako je tomu v případě kvantových algoritmů, stává se doposud polynomiální složitost rázem exponenciální.

hovoříme o výpočtu Gaussových sum, ve své podstatě se jedná o destruktivní interferenci, čili shluknutí většího množství amplitud pravděpodobnosti. Předpokládejme operátor $\hat{U}_{QFT}$ odpovídající N-dimenzionální kvantové Fourierově transformaci, pak lze psát

$$|\Psi\rangle_6 \to \hat{U}_{QFT}\,|\Psi\rangle_5 = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} \left(\sum_{a=0}^{N-1} \left(e^{2\pi i a x/N} \right) |a\rangle_A \right), \tag{1.22}$$

což lze pouhým přeskupením přepsat do tvaru

$$|\Psi\rangle_6 = \frac{1}{\sqrt{N}} \sum_{a=0}^{N-1} \left(\sum_{x=0}^{N-1} e^{2\pi i a x/N} \right) |a\rangle_A \,, \tag{1.23}$$

čímž podle očekávání dostáváme superpozici Gaussových sum pro všechna $a < N$, tedy

$$|\Psi\rangle_6 = \frac{1}{\sqrt{N}} \sum_{a=0}^{N-1} G(a, \chi_N)\,|a\rangle_A \,. \tag{1.24}$$

Tímto způsobem se nám podařilo hodnotu sumy převést do koeficientu před a; pokud pro Gaussovu sumu platí $G(a, \chi_N) = 0$, je amplituda pravděpodobnosti naměření vlastního stavu, který odpovídá danému a, nutně nulovou. Následným procesem měření na registru A tak získáme pouze takový stav, kdy $G(a, \chi_N) \neq 0$, což podle dříve uvedených vět vede k řešení problému bezčtvercovosti: Pokud je naměřené a takové, že $(a, N) = 1$, prohlasíme číslo N za bezčtvercové, v opačném případě pak dostaneme číslo, které sdílí společný faktor se čtvercovou částí N. Za pomoci Eukleidova algoritmu lze N efektivně redukovat; v případě, že ani poté nezískáme jeho úplný prvočíselný rozklad, resp. bude stále zapotřebí ověřit, zdali je takto získané číslo bezčtvercovým, je možné celý výpočet pro novou hodnotu zopakovat.

Analýza algoritmu

Vzhledem k stochastickému charakteru kvantového měření před námi vyvstává otázka, s jakou pravděpodobností bude možné k předpokládaným „výstupům“ algoritmu dospět.

V případě prvního měření přejde algoritmus buďto do stavu $(m, N) = 1$, nebo $(m, N) > 1$, což implikuje přinejmenším částečné vyřešení problému. Pokud však N obsahuje více čtvercových částí, může nastat situace, že ani ze znalosti společného faktoru není možné bezčtvercovost stále rozhodnout;

pak bude nutné celý algoritmus pro redukované N zopakovat. Pro uvažované $N = pq^2$ i $N = pq$, kde p, q jsou prvočísla, k rozkladu čísla nicméně dospějeme, proto lze psát $p = 1$.

Po závěrečné aplikaci kvantové Fourierovy transformace bude možné naměřit jen stavy s nenulovou amplitudou pravděpodobnosti, respektive $G(a, \chi_N) \neq 0$ podle dříve uvedených vět. Po měření mohou nastat dva případy: $a$nebude mít s N žádného společného dělitele, pak je N nutně bezčtvercové, v případě $(a, N) > 1$ pak podle věty III získáme faktor čtvercové části čísla, čímž je celý problém ihned vyřešen. Jakkoli jsme větu III formulovali pro prvočísla, zjevně platí $N = p(qu)^2$, pak tedy

$$g(a, pq^2u^2) = g(aq^2u^2, p)g(apu^2, q^2)g(apq^2, u^2).$$

Situace $g(a, pq^2u^2) \neq 0$ nastane jen tehdy, kdy $qu \mid a$, což zabrání „vynulování“ dvou posledních sum.

Spolu s tím se nabízí otázka, s jakou pravděpodobností naměříme jaké a, respektive m; je zřejmé, že amplituda pravděpodobnosti bude dosáhne maxima tehdy, kdy bude a rovno čtvercové části, neboť podle důkazů výše platí

$$g(aq^2, p)g(ap, q^2) = \left(\sum_{m=0}^{p-1} \chi_p(m) e^{2\pi i q^2 am/p}\right) \left(\sum_{m=0}^{q^2-1} \chi_{q^2}(m) e^{2\pi ipam/q^2}\right),$$

pokud tedy $q^2 \mid a$, pak

$$g(ap, q^2) = \left(\sum_{m=0}^{q^2-1} \chi_{q^2}(m) e^{2\pi ipm}\right),$$

$\chi_{q^2}(m)$ je rovno 1 pro všechna $(m, N) = 1$ a $e^{2\pi ipm} = 1$, tudíž

$$g(ap, q^2) = \sum_{(m,N)=1} 1.$$

Pro všechna ostatní a soudělná s čtvercovou částí bude pravděpodobnost stejná, neboť sumace vyjadřují skládání různě „natočených“, avšak vždy stejně velkých, tj. jednotkových vektorů.

Jak se ještě ukáže, pravděpodobnostní charakter algoritmu nutně závisí na možnosti sestrojení N-dimenzionální QFT; pokud je možné takového operátoru využít, pak nutně dospějeme ke správnému výsledku s pravděpodobností $p = 1$, a algoritmus lze tak zařadit do třídy *Exact Quantum Polynomial* (*EQP*).

... polynomial. Jak vzápětí ukážeme, algoritmus je skutečně polynomiální. Uvažujme vstup N a předpokládejme, že kvantové obvody pro elementární operace (sčítání aj.) mají časovou komplexitu právě $\mathcal{O}(\log N)$. Dále prezentovaný binární algoritmus pro výpočet GCD i Jacobiho symbolu proběhne v přibližně $\mathcal{O}(\log N)$ iteracích, ve výsledku tak získáváme $\mathcal{O}(\log^2 N)$. Kvantová Fourierova transformace má komplexitu $\mathcal{O}(n^2)$, kde n počet qubitů, resp. $n = \log_2 N$, a tedy $\mathcal{O}(\log^2 N)$. Neboť $\mathcal{O}(\log^2 N + \log^2 N) = \mathcal{O}(\log^2 N)$, je časová složitost algoritmu polynomiální, konkrétně polylogaritmická, a algoritmus je tudíž efektivní.

Kapitola 2

Implementace v QCL

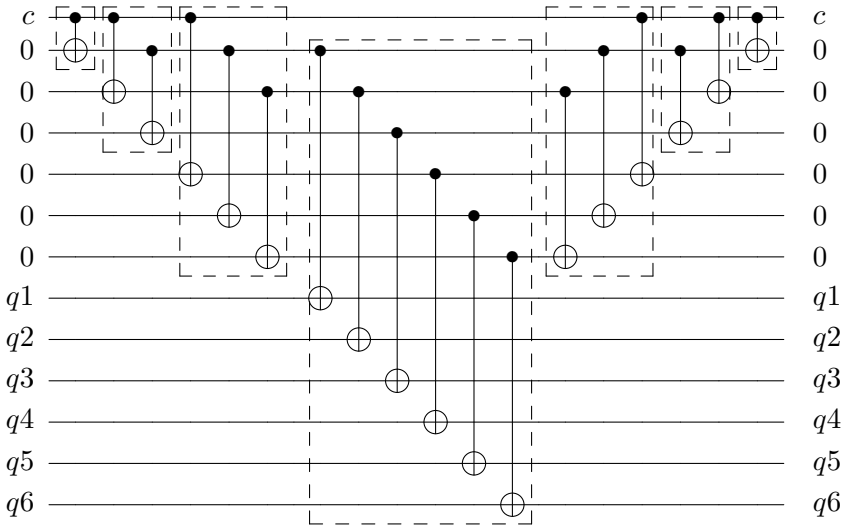
Na následujících stránkách se pokusíme výše popsaný algoritmus implementovat v jazyce QCL. Postupovat budeme od dílčích funkcí, na jejichž základě vystavíme kvantové obvody pro elementární matematické operace, čehož následně využijeme při konstrukci algoritmů na výpočet GCD a Jacobiho symbolu. Vzhledem k náročnosti operací prováděných nad kvantovým jádrem před námi vyvstává požadavek v podobě nezbytné optimalizace obou procedur, neboť právě jejich komplexita je s ohledem na výše řečené určující pro výslednou časovou i paměťovou náročnost diskutovaného algoritmu. Při snaze o dosažení nezbytné efektivity se dovolíme inspirovat některými z postupů vedoucích k paralelizaci, tj. urychlení výpočtů tak, jak byly tyto možnosti nastíněny v [14]; naše práce jednotlivé návrhy rozpracuje, algoritmizuje a především jich poprvé využije pro simulaci kvantových výpočtů. Kompletní zdrojové kódy všech níže zmíňovaných funkcí lze pak okomentované nalézt v příloze práce.

2.1 Elementární operace

Operace rozvětvení stavu qubitů (fan-out)

Při programování kvantových funkcí se nutně setkáváme s případy, kdy je prováděné operace zapotřebí podmínit stavem určitého qubitu. Předpokládejme qubit, který kontroluje hradla aplikovaná na dalších n bitech; vzhledem k povaze kvantových výpočtů je pak nutné jednotlivá hradla aplikovat v posloupnosti, což však vyústí v lineární, tj. $\mathcal{O}(n)$ složitost. Tuto složitost lze však jednoduše redukovat na logaritmickou: Předpokládejme dalších $n-1$ pomocných qubitů (dále jen ancillae), do nichž stav výchozího qubitu zkopírujeme, čehož je možné dosáhnout právě v čase $\log_2 n$. Proceduru ilustruje

následující schéma

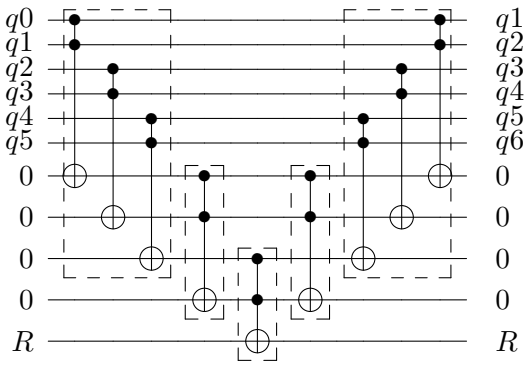


Je zřejmé, že každá ohraničená skupina nekolidujících kvantových hradel zabere právě jeden časový úsek; nejprve je stav qubitu překopírován do prázdných ancillae, v jediném kroce je provedena podmíněná operace[1] a hodnoty uložené v ancillae jsou pomocí inverzní operace „odpočítány“, respektive navráceny do počátečního stavu podle $|1 \otimes 1\rangle = |0\rangle$. Jakkoli jsme dosáhli optimalizace časové náročnosti, učinili jsme tak za využití dalších qubitů, což v případě reálné implementace zjevně ovlivní riziko kvantové dekoherence. Kvůli kaskádové konstrukci může nastat situace, že jediná chyba vzniklá špatnou aplikací hradla bude následně rozšířena do dalších qubitů, přesto se však domníváme, že zatímco chyby plynoucí z nedokonalé implementace lze do jisté míry potlačit, právě čas představuje jeden z nejvýznamnějších faktorů podílejících se na jevu dekoherence.

Operace fan-in

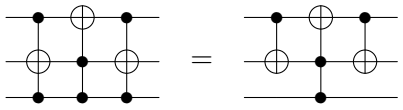
Dále zaveďme operaci fan-in coby logickou funkci AND nad všemi stavy q_0, $q_1 \ldots q_n$ kvantového registru, která výsledek uloží do předpřipraveného qubitu R. „Shluknutí“ logických hodnot lze analogicky k předchozímu případu provést za pomoci Toffoliho hradel v logaritmickém čase při využití lineárního počtu qubitů, pro $n = 6$ situaci znázorňuje

[1] Zde CNOT, obecně se může jednat o jakékoliv podmíněné (ono C-) hradlo.



Podmíněný SWAP

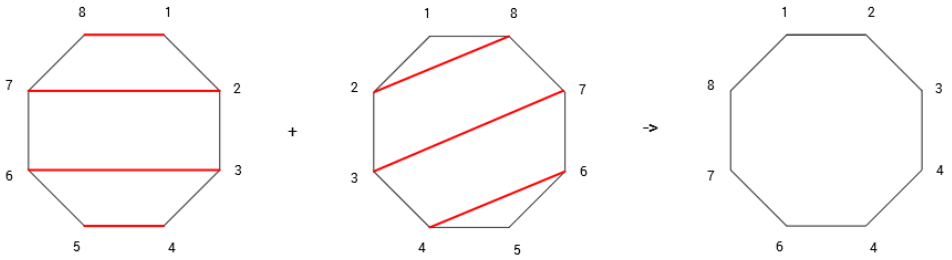
Spolu s tím definujme i optimalizovanou verzi Fredkinova (CSWAP) hradla. V základní verzi lze podmíněné prohození stavů dvou qubitů provést pomocí tří Toffoliho hradel (CCNOT), což je možné nicméně zredukovat na dvě CNOT hradla a jedno hradlo Toffoliho, tudíž



Podmíněný SWAP, čili podmíněné prohození stavů dvou registrů, lze pak vzhledem ke shora uvedenému operátoru „rozvětvení“ provést v logaritmickém čase, respektive $2\log_2 n+1$, což zahrnuje zkopírování hodnot kontrolního qubitu, paralelní aplikaci Fredkinových hradel v jednom kroce a následné odpočítání hodnot jednotlivých ancillae qubitů.

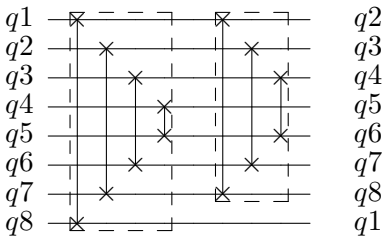
Cyklická permutace registru

Při konstrukci sofistikovanějších algoritmů se setkáváme s nutností implementace aritmetických operací, mezi něž se na základní úrovni řadí i redukce faktorů dvojky. Vzhledem k binární reprezentaci čísel lze podobnou operaci převést na posun bitů „doprava“, neboli cyklickou permutaci registru, vzhledem k vzájemné provázanosti hodnot však není možné provést posun v jediném kroce. Lineární časovou složitost lze nicméně jednoduše nahradit složitostí konstantní, povšimneme-li si, že každá permutace cyklu sestává ze dvou vzájemně nezávislých transpozic; kupříkladu pro $n = 8$ platí



Obrázek 2.1: Cyklická permutace pro $n = 8$

Příslušný kvantový obvod lze znázornit jako



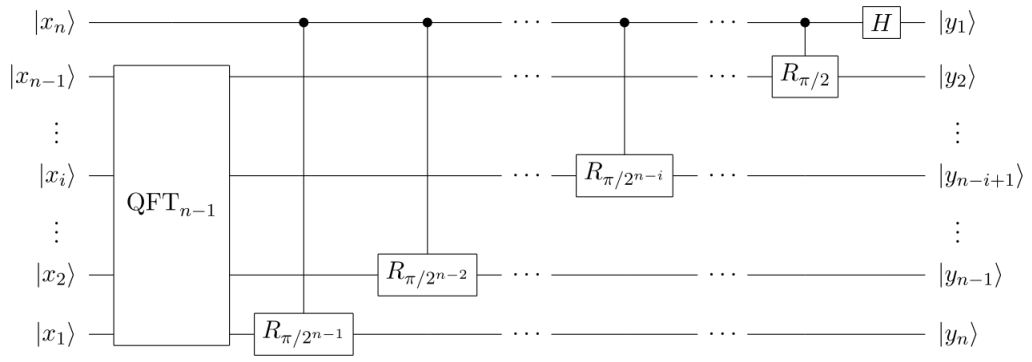
kde ohraničené skupiny hradel lze aplikovat v jediném kroce. Ačkoliv schémata ukazují pouze posun „doprava“, analogicky je možné vyjádřit libovolnou cyklickou permutaci, čehož v kontrastu se zachyceným dělením využijeme i pro násobení mocninou dvojky. Každá permutace pak zabere celkem 6 kroků, respektive dvakrát tři kroky shora uvedeného Fredkinova hradla.

Popsanou operaci lze zobecnit i pro situace, kdy je zapotřebí provedení permutace podmínit stavem určitého qubitu: V tom případě stačí využít operátoru rozvětvení, což výslednou časovou složitost zvyšuje na $2 \log_2 n + 6$.

Kvantová Fourierova transformace pro 2^n

Ačkoliv Fourierova transformace představuje aritmeticky náročnou operaci, vzhledem k jejímu využití coby základního prvku v mnohých algoritmech si dovolíme obvod pro výpočet 2^n-dimenzionální QFT zařadit mezi elementární operátory. Operátor sestává ze série Hadamardových hradel a fázových posunů stavů qubitů, viz schéma níže

Výsledná časová složitost obvodu je kvadratická vzhledem k počtu transformovaných qubitů, tedy $\mathcal{O}(n^2)$.



Obrázek 2.2: Obvod pro výpočet QFT, přebírám z [15]

2.2 Elementární matematické operace

Jakkoli se následujících řádkách budeme věnovat operacím, jež jsme nazvali matematickými *elementárními*, při implementaci se právě tato část algoritmu ukázala být nejvíce problematickou. Hovoříme pak zejména o kvantovém obvodu pro sčítání, resp. odčítání a porovnání dvou aritmetických hodnot, kdy jednotlivé funkce lze mezi sebou do jisté míry mapovat.

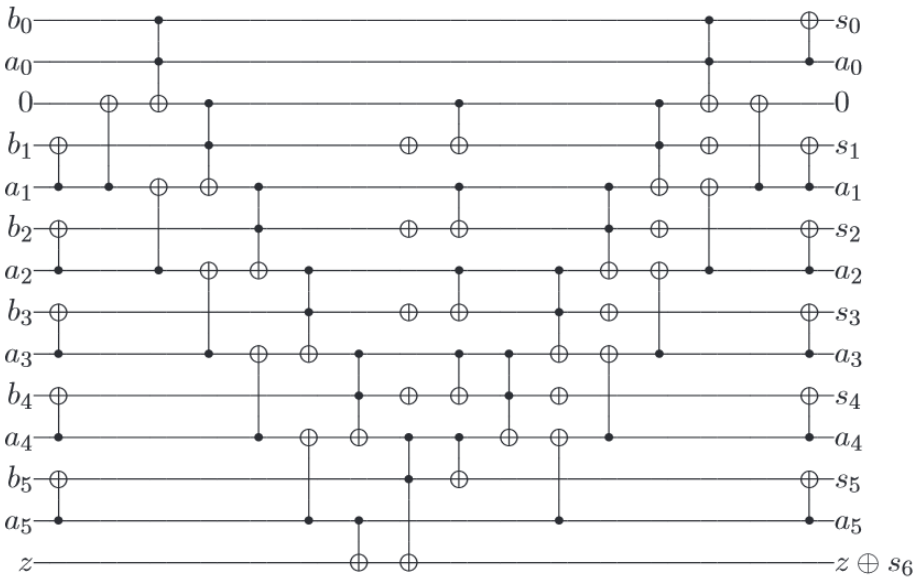
Pokud jsme v sekci věnované složitostní analýze algoritmu hovořili o možnosti provedení podobných operací v čase logaritmickém, bylo tomu tak proto, že některé články věnující se konstrukci kvantových sčítaček právě k této časové komplexitě[2] dospěly. Konkrétně pak máme na mysli práci *A logarithmic-depth quantum carry-lookahead adder* [16], která prezentuje efektivní algoritmy pro sčítání, odčítání, porovnávání aj. v čase $\mathcal{O}(\log n)$ za využití $\mathcal{O}(n)$ pomocných qubitů (ancillae). Jedná se o obvody nadmíru sofistikované, nicméně stejně tak i poměrně nejednoduše implementovatelné, jakkoli se nám i přes mnohé nejednoznačnosti v diskutované práci podařilo kompletní algoritmizace dosáhnout.

Jistě bychom implementaci sčítačky neuváděli podobnými slovy, pokud bychom nemuseli vzápětí dodat, že ačkoliv pro určité hodnoty vracely algoritmy správné výsledky, v případě větších čísel docházelo k opakovaným chybám. Vzhledem ke skutečnosti, že autoři v práci prezentují schémata pro případy některých velikostí kvantových registrů, si jsme na základě porovnání chodu algoritmu jisti správnou interpretací uvedených obvodů, přesto však v budoucnu zamýšlíme celou situaci znovu podrobně přezkoumat.

[2] Neboli též hloubce (depth), v každém případě se jedná o počet časových kroků nutných k provedení dané operace.

S ohledem na korektnost matematických operací je tak nutné od zamýšlené časové složitosti ustoupit a implementovat takový obvod, který bude o něco robustnějším: Přestože mnohé práce věnující se problematice kvantových algoritmů stále využívají sčítačky V. Vedrala, A. Barenca et al. [17], rozhodli jsme se držet postupu navrženého v práci *A new quantum ripple-carry addition circuit* [18]. Obvod oproti shora diskutované nestabilní sčítačce sice charakterizuje lineární časová složitost, pro výpočet je však zapotřebí pouze jediného pomocného qubitu (ancillae).

Uvažujme dva kvantové registry A, B o délce $n = 6$ s příslušnými qubity a_0, a_1 ... a_5, b_0, b_1 ... b_5 a qubit z, do kterého se uloží nejvyšší bit součtu; pak situaci ilustruje schéma níže



Obrázek 2.3: Obvod kvantové sčítačky pro $n = 6$. Schéma přejímám z [18].

Vzhledem k požadavku na reverzibilitu prováděných operací je zřejmé, že při invertování pořadí jednotlivých hradel lze obvodu využít i pro odčítání; pokud $b_1 = b_0 + a$, pak invertováním příslušného operátoru dostaneme $b_0 = b_1 - a$, s ohledem na další konstrukci algoritmu pak bude vždy platit $b_1 > a$, což situaci zjednodušuje. V opačném případě je nutné výslednou hodnotu b komplementovat.

Obvod můžeme jednoduchou úpravou rozšířit i pro případ porovnání dvou hodnot; uvažujme čísla a, b, pak je nejprve zapotřebí sérií paralelních

Pauliho-X hradel komplementovat a, aplikovat obvod do fáze, kdy se infromace o nejvyšším bitu uloží do z, a následně všechny předchozí operace kromě samotného zápisu do $z$invertovat. Pokud platí $a < b$, pak nutně $z = 1$.

Implementaci posledně zmiňovaného obvodu v jazyce QCL lze spolu s předchozí, nestabilní sčítačkou jakož i dílčími modifikacemi obou funkcí nalézt v příloze této práce.

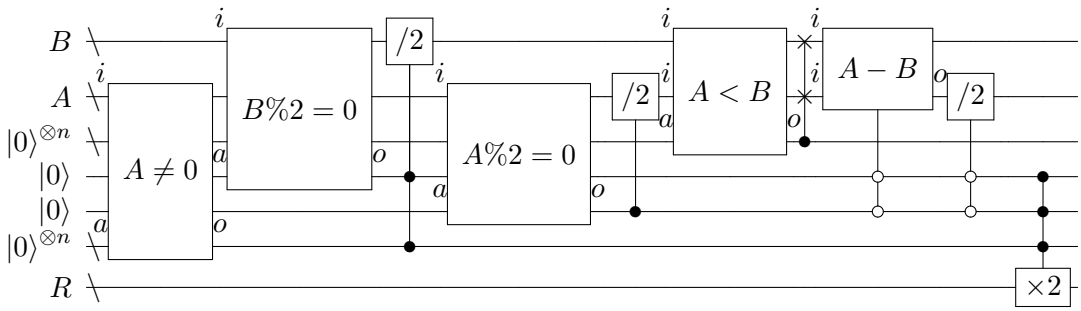
2.3 Obvod pro výpočet GCD

Konečně přistupme k návrhu obvodu pro výpočet největšího společného dělitele. Jakkoli je již několik tisíciletí lidstvu znám Eukleidův algoritmus, který představuje možnost efektivního výpočtu GCD, s rozvojem výpočetní techniky se v druhé polovině minulého století ukázalo nezbytné nalézt takový algoritmus, jenž by byl optimalizovaný pro využití v procesorech operujících na bázi dvojkové soustavy. Řešení tohoto problému nabízí Steinův, neboli též binární GCD algoritmus, z něhož při návrhu kvantového obvodu vyjdeme. Nejprve však připomeňme, že binární GCD algoritmus je algoritmem rekurzivním, kdy jednotlivé výpočetní cesty zachycuje předpis

$$GCD(A,B)\begin{cases} 2GCD(A/2,B/2) & \text{Pokud } A\%2=0 \text{ a } B\%2=0 \\ GCD(A/2,B) & \text{Pokud } A\%2=0 \text{ a } B\%2=1 \\ GCD(A,B/2) & \text{Pokud } A\%2=1 \text{ a } B\%2=0 \\ GCD(\frac{A-B}{2},B) & \text{Pokud } A\%2=1 \text{ a } B\%2=1, \text{ zároveň } A \geq B \end{cases}$$

Zatímco v případě klasických počítačů se průběh výpočtu na konci každé iterace větví, při konstrukci kvantové obdoby obvodu musíme mít na paměti, že algoritmus bude nutně zpracovávat celou superpozici čísel: Při každé iteraci algoritmus projde „všemi“ možnými výpočetními cestami, resp. jim příslušnými funkcemi, jejichž vykonání bude ad hoc podmíněno hodnotami kvantových registrů. Dílčí iteraci nazvěme výpočetním blokem; podobný blok lze pak sestrojit jako posloupnost operací prováděných v každé z větví právě nastíněného binárního algoritmu.

Problematikou výpočtu GCD na kvantovém počítači se před námi zabýval již I. Markov a A. Saeedi v práci [19], kde podobně jako v našem případě prezentovali obvod založený na Steinově binárním algoritmu; jejich návrhu pak odpovídá schéma níže



Uvažujme stejně velké vstupní registry A, B s příslušnými číselnými hodnotami, čtyři ancillae registry X_1, X_2, X_3, X_4 po jednom bitu a pomocný registr R o velikosti shodné s registry A, B; pak každý výpočetní blok sestává ze čtyř částí podle definice výše:

I. část Proběhne ověření, zdali pro A na vstupu platí $A \neq 0$; pokud ano, uloží se do X_1 logická hodnota 1.

II. část Redukce faktoru dvojky čísla B. Nejprve je nutné zjistit, zdali $B\%2 = 0$, čehož lze díky binární reprezentaci čísel dosáhnout jedinou, stavem posledního bitu B podmíněnou CNOT operací na qubitu X_2.[3] Následně proběhne podle shora diskutovaných operací cyklická permutace registru B podmíněná stavem X_2, v případě výše zmiňovaného návrhu resp. i X_1, což dále ještě ozřejmíme.

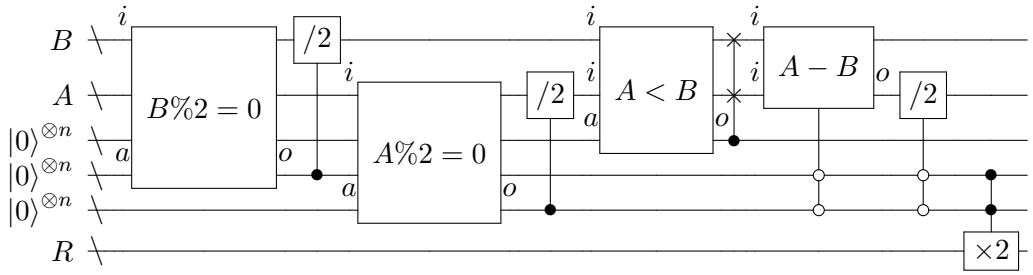
III. část Analogicky k předchozí části, tentokrát však budou operace provedeny nad registrem A a qubitem X_3. Pokud platí $X_2 = X_3 = 1$, resp. $A\%2 = B\%2 = 0$, podle shora uvedené definice je závěru algoritmu zapotřebí získanou hodnotu GCD vynásobit dvojkou; podmíněně tak permutujme stav pomocného registru R, resp. vynásobme příslušnou hodnotu R dvojkou.

IV. část Závěrečná část obvodu. Nejprve ověříme, zdali neplatí $A < B$, a logickou hodnotu uložíme do X_4: je-li $X_4 = 1$, s ohledem na další kroky je nutné hodnoty v registrech A i B pomocí CSWAP operace prohodit. V další fázi proběhne podmíněné odečtení obou hodnot; provedeme tudíž reverzní

[3] Poslední bit, resp. qubit zápisu je před provedením operace i po ní nutné invertovat, neboť logická hodnota 1 odpovídá situaci, kdy $B\%2 = 0$.

operaci sčítání, kdy hradla budou podmíněna qubity X_2, X_3. Transformace proběhne právě tehdy, pokud na počátku iterace platilo $A\%2 = B\%2 = 1$. Rozdíl dvou lichých čísel je však sudé číslo, proto je zapotřebí z výsledku uloženého v registru A analogicky podmíněnou permutací redukovat faktor dvojky.

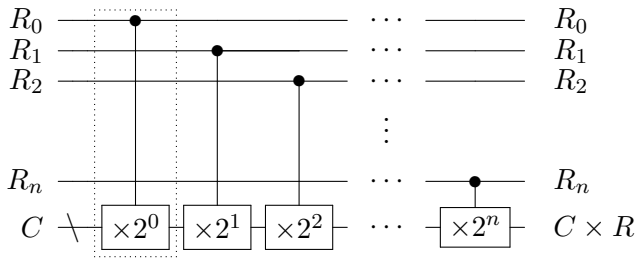
Jak se tedy ukazuje, prezentovaný obvod pro výpočet GCD představuje nanejvýš věrnou, avšak specifikům kvantových výpočtů přizpůsobenou verzi původního binárního algoritmu. Přesto se domníváme, že autoři nedosáhli při konstrukci bloku nejvyšší možné efektivity, neboť celou první část obvodu není zapotřebí provádět. Ačkoliv mělo ověření $A \neq 0$ zamezit situaci, kdy by byl po skončení reálného výpočtu redukován faktor dvojky z čísla B, není podobný krok nezbytným, ba ho lze považovat za nadbytečný, neboť další chod možnou chybu ošetřuje. Důvod je zřejmý: Pokud $A = 0$, pak muselo proběhnout záverečné odečtení hodnot $\frac{A-B}{2}$, což však nastane pouze tehdy, kdy $B\%2 = 1$. Neboť nyní $A = 0$, resp.$A\%2 = 0$, další odečtení neproběhne, a zjevně tak bude i nadále platit $B\%2 = 1$. Tím je dokázano, že pro $A = 0$ k redukci B nedojde, díky čemuž však můžeme redukovat podobu algoritmu:



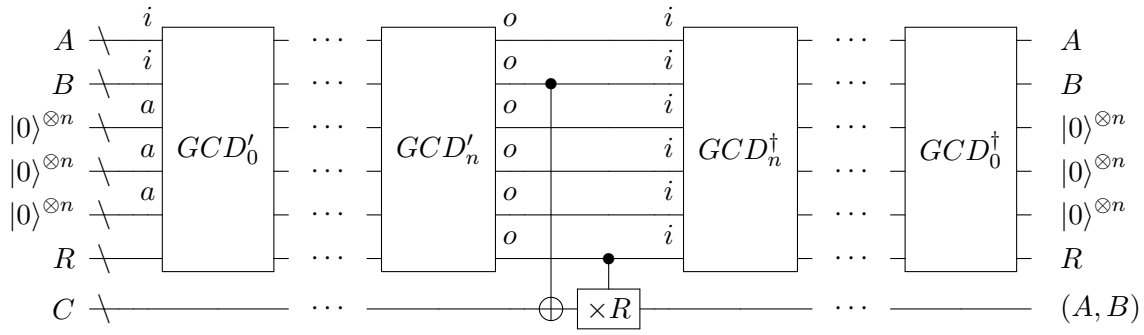
Zjednodušením výpočetního bloku jsme tak dosáhli nejen snížení časové složitosti, ale optimalizovali jsme i počet ancillae qubitů.

Po *n*iteracích bude lichá část největšího společného dělitele uložena v registru A; v jediném kroce pak stav hodnoty uložené v A překopírujme do připraveného registru C. Následně je nezbytné hodnoty registru $C \cdot R$ vynásobit tak, aby výsledný GCD obsahoval i dříve vyredukovaný faktor dvojky. Neboť R je tudíž mocninou dvojky, lze operaci násobení převést na sled cyklických permutací registru A v závislosti na stavu jednotlivých

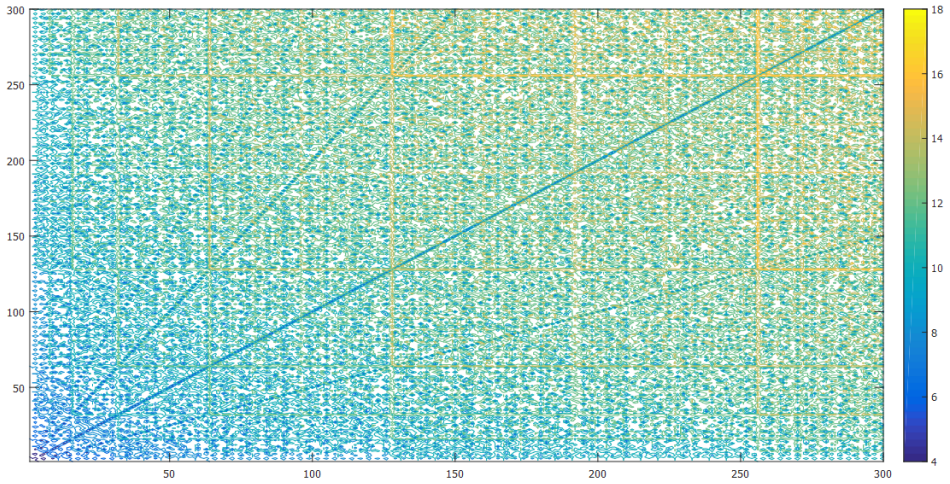
qubitů R, tedy



Je zřejmé, že hradlo v ohraničené části není nutné aplikovat, neboť pro $R_0 = 2^0 = 1$. Po vynásobení hodnot je hodnota GCD uložena v registru C; následně je možné celý algoritmus invertovat, a tudíž při odpočítání hodnot uložených v ancillae obnovit výchozí hodnoty A, B. Celkový průběh algoritmu znázorňuje schéma níže

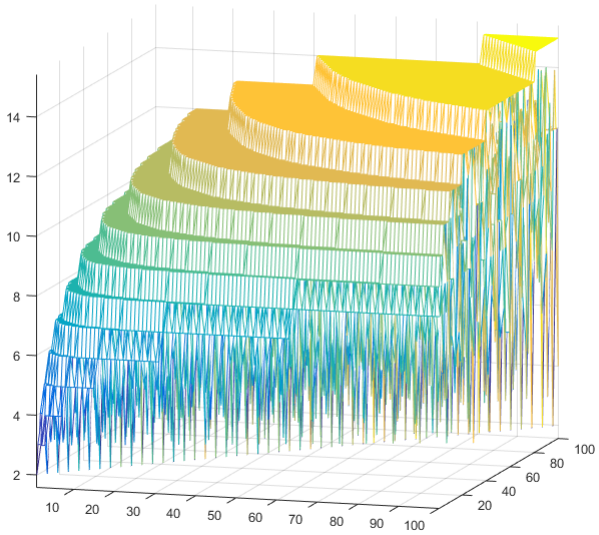


Připomeňme, že ancillae qubity je s ohledem na požadavek reverzibility všech prováděných transformací nutné ponechat v nezměněném stavu po celou dobu výpočtu. Dále uvažujme pomocný registr R o délce shodné s velikostí registrů A, B, do něhož budou postupně ukládány vyredukované faktory dvojky pro případ $A\%2 = B\%2 = 0$. Pokud jako $n$označíme počet iterací algoritmu, je paměťová složitost celého algoritmu zřejmě lineární, tedy $\mathcal{O}(n)$. Stále jsme však nezodpověděli otázku, kolikrát bude zapotřebí dílčí výpočetní bloky iterovat: Vzhledem k superpozici je nezbytné hledat právě tu „nejdelší“ možnou cestu, resp. maximální počet iterací příslušný množině daných kvantových stavů. Prostou úvahou lze dospět ke stanovení logaritmické časové závislosti, kdy se na nejdelší cestu algoritmem zřejmě vydají mocniny dvojky; situaci ilustruje následující schéma



Obrázek 2.4: Počet iterací pro odpovídající hodnoty (x, y), kdy $x \leq 300$.

Výslednou časovou složitost lze pak experimentálně aproximovat jako $\lceil \log_2[a(b+1)] \rceil + 1$, kde a, b jsou bitové velikosti čísel A, B, jak to dokazuje diagram níže



Obrázek 2.5: Porovnání počtu iterací (x, y) a hodnot funkce $\lceil \log_2[x(y+1)] \rceil + 1$, kdy $x \leq 100$.

2.4 Obvod pro výpočet Jacobiho symbolu

Podobně jako tomu bylo v předchozím případě, i při konstrukci kvantového obvodu pro výpočet Jacobiho symbolu přímo vyjdeme z příslušného binárního algoritmu. Konkrétně se pak necháme inspirovat návrhem klasického obvodu tak, jak na základě Steinova algoritmu představil J. Shallit v [20]. Vzhledem k podobnosti s binárním algoritmem pro výpočet GCD se tudíž celá implementace do jisté míry zjednodušuje; průběh výpočtu pro $R = (\frac{a}{n})$ pak definujme jako

Binární algoritmus pro výpočet Jacobiho symbolu

dokud $(a\%2 = 0)$
    $a := a/2$;
pokud $(n\%8 = 3)$ nebo $(n\%8 = 5)$
    $R := -R$;
pokud $(a < n)$
    $SWAP(a, n)$;
pokud $(a\%4 = 3)$ a $(n\%4 = 3)$
    $R := -R$;

    $a := (a - n)/2$;

pokud $(n\%8 = 3)$ nebo $(n\%8 = 5)$
    $R := -R$;

Jednotlivé kroky algoritmu jsou založeny na vztazích, které pro výpočet Jacobiho symbolu vyplývají ze zákona kvadratické reciprocity. Stejně jako v případě GCD je pak nutné volit „nejdelší“ možnou cestu, resp. obvod zkonstruovat tak, aby v každé iteraci podmíněně aplikoval operace příslušné všem krokům shora definovaného algoritmu. Samotný obvod pak charakterizuje schéma na následující stránce.

Uvažujme kvantové registry A, N o stejné velikosti, čtyři ancillae qubity X_1, X_2, X_3, X_4 a pomocný registr R taktéž o velikosti jednoho qubitu; pak lze průběh jednoho výpočetního bloku popsat následovně:

Část I. Redukce faktoru dvojky z čísla A. Podobně jako u GCD proběhne nejprve ověření, zda $A\%2 = 0$, kdy je příslušná logická hodnota uložena do X_1; následuje cyklická permutace, neboli posun posun hodnot bitů „doprava“. Podle předpisu algoritmu je nutné zjistit, zda $A\%8 = 3$ nebo

$A\%8 = 5$, čehož lze dosáhnout hradly aplikovanými na poslední 3 qubity registru A; pokud je podmínka splněna, aplikujeme hradlo NOT, resp. CNOT na registr R.

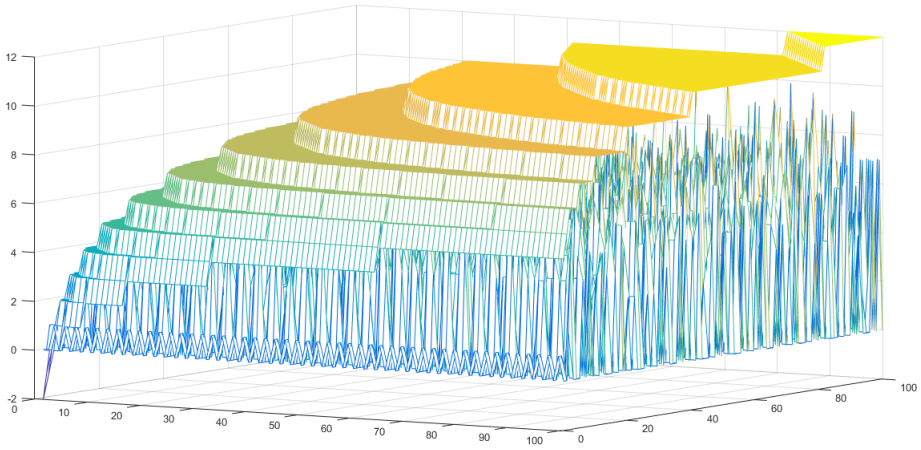
Část II. Proběhne další ověření, zda snad již platí $A\%2 = 1$.[4] Pokud ano, tj. $X_2 = 1$, a zároveň $A < N$, proběhne prohození příslušných stavů. Následně zjišťujeme, zda $A\%4 = 3$ a zároveň $N\%4 = 3$, pokud ano, pak aplikujeme NOT, resp. CNOT na registr R.

Část III. Závěrečná část. Pokud z předchozího ověření $A\%2 = 1$ platí $X_2 = 1$, proběhne odečtení $A - N$, které je následováno podmíněnou redukcí dvojky, resp. cyklickou permutací nad registrem A. Pak je ale nutné podobně jako v I. části zjistit, zda zda $A\%8 = 3$ nebo $A\%8 = 5$, pokud ano, znovu aplikujeme operaci NOT na R.

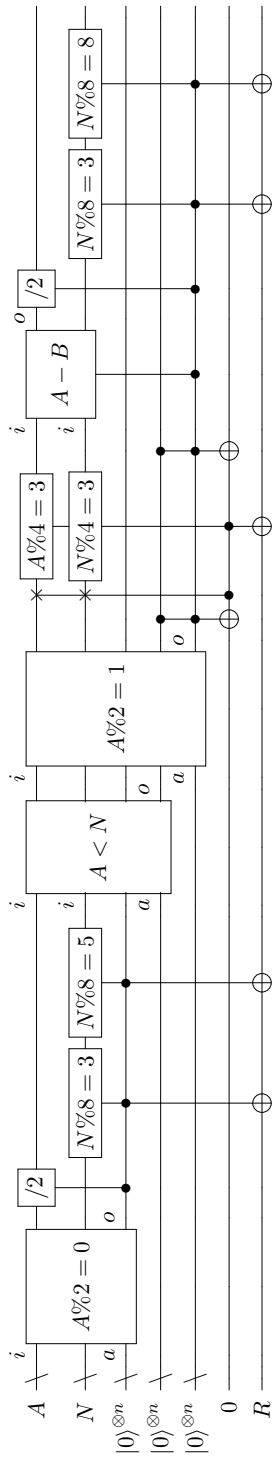
Pokud výše uvedený algoritmus provádí „násobení“ $R := -R$, v našem případě lze využít povahy NOT hradla coby logické operace XOR, tedy $|0 \otimes 0\rangle = |0\rangle$, $|0 \otimes 1\rangle = |1\rangle$, $|1 \otimes 1\rangle = |0\rangle$. Výslednou hodnotu R je po skončení výpočtu možné zkopírovat do připraveného registru a celý běh algoritmu invertovat, čímž obnovíme původní hodnoty A, N i vynulujeme stavy jednotlivých ancillae qubitů.

Pokud pak coby n označíme počet iterací, bude zapotřebí právě $3n + 1$ ancilla qubitů a jeden qubit R, což stejně jako u algoritmu GCD implikuje lineární paměťovou složitost. Maximální počet iterací je lue následně aproximovat jako $\lceil \log_2(an) \rceil - 2$, kde a, n jsou bitové velikosti čísel A, N; tedy

[4] Nabízí se otázka, proč jsme druhotného ověření faktoru dvojky nevyužili již v případě GCD obvodu: Bylo tomu tak vzhledem ke snaze o redukování počtu ancillae, kdy nešlo použít méně než čtyři pomocné qubity. Pro výpočet Jacobiho symbolu jsou v zásadě zapotřebí pouze tři ancillae, proto se nabízí jeden přebývající qubit využít pro druhotné ověření, a tak zároveň urychlit chod algoritmu.



Obrázek 2.6: Porovnání počtu iterací algoritmu pro (x, y) a hodnot funkce $\lceil \log_2(xy) \rceil - 2$, kdy $x \leq 100$.


A
N
$|0\rangle^{\otimes n}$
$|0\rangle^{\otimes n}$
$|0\rangle^{\otimes n}$
0
R
$A\%2 = 0$
/2
$N\%8 = 3$
$N\%8 = 5$
$A < N$
$A\%2 = 1$
$A\%4 = 3$
$N\%4 = 3$
$A - B$
/2
$N\%8 = 3$
$N\%8 = 8$
i
o
a

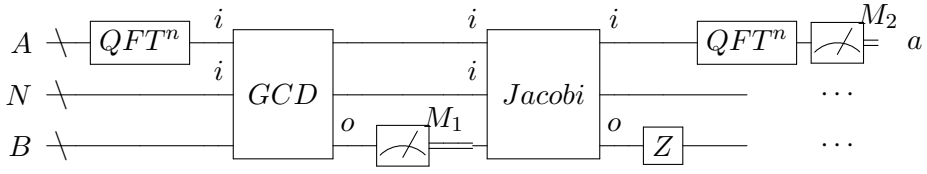
2.5 Finální algoritmus

Po sestrojení všech nezbytných kvantových procedur pak konečně přistupme k implementaci shora disktuvaného algoritmu pro řešení problému bezčtvercovosti.

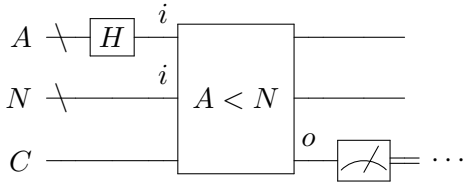
Přestože jsme v sekci věnované teoretickému popisu uvažovali možnost využití právě N-dimenzionální kvantové Fouerierovy transformace, ukazuje se, že při praktické implementaci se budeme muset spokojit s QFT 2^n dimenzionální: Jakkoli je teoreticky možné konstrukce zamýšleného operátoru dosáhnout ([21]), uvedený způsob se jeví být pouze obtížně realizovatelným, z čehož lze usuzovat na využití méně přesných, avšak snáze dosažitelných operací v případě reálné implementace kvantových algoritmů.

Nutně tak před námi vyvstává otázka, nakolik podobné využití nepřesných hradel ovlivní správnost naměřeného výsledku: Vzhledem k nemožnosti připravit počáteční superpozici právě N stavů budou Hadamardovými hradly generovány všechny $2^{\lceil \log_2 N \rceil}$ dílčí hodnoty kvantového registru o velikosti $\lceil \log_2 N \rceil$, kdy případy pro $m > N$ (podle dříve zavedeného formalismu) je nezbytně nutné redukovat. Porovnáme tudíž hodnoty m i N a na kvantovém registru s logickou hodnotou $m \geq N$ provedeme měření: Pokud získáme vlastní hodnotu systému odpovídající logické hodnotě 1, registr A se bude nacházet v superpozici hodnot $m \geq N$, a celý algoritmus je tudíž zapotřebí iterovat znovu. Pravděpodobnost, se kterou může situace nastat, je nicméně $p < \frac{1}{2}$: Pokud $p \geq \frac{1}{2}$, zároveň i $N \leq \frac{l}{2}$, kde vzhledem k výše řečenému $l = 2^{\lceil log_2 N \rceil}$. Dostáváme však spor, neboť zjevně $2^{log_2 N+1} > 2^{\lceil log_2 N \rceil}$.

Na vliv nepřesné Fourierovy transformace při vyčíslování Gaussových sum nahlédneme o několik stránek níže; zatím uveďme schéma algoritmu podle jeho návrhu v teoretické části,



Při redukci hodnot $m > N$ pomocí porovnání obou registrů je dále zapotřebí uvažovat příslušnou počáteční část obvodu, tedy

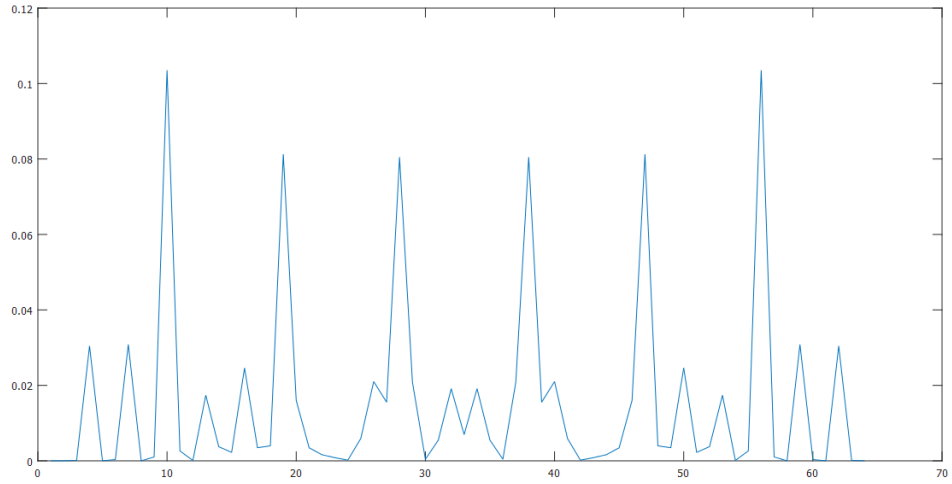


Kapitola 3

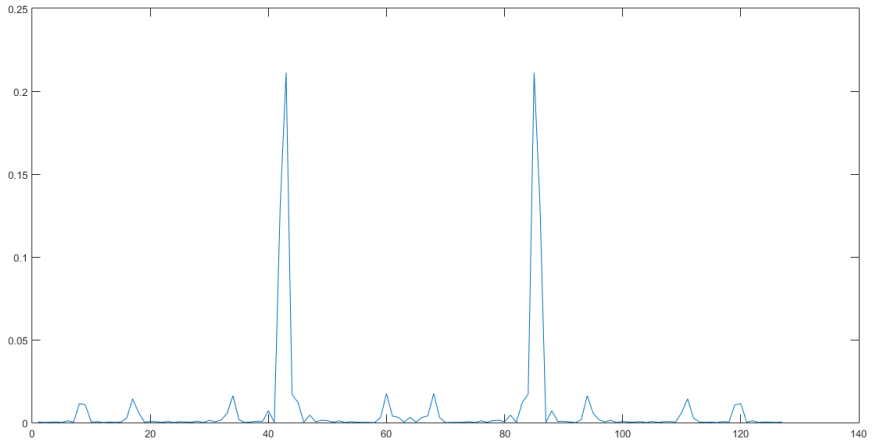
Simulace výpočtu

Konečně se pak pokusme průběh diskutovaného algoritmu nasimulovat v prostředí jazyka QCL. Jakkoli bude algoritmus v mnohých případech navracet faktor čísla N již po provedení prvního, resp. druhého měření, omezíme se pouze na případy po samotném vyčíslení Gaussových sum, na základě čehož budeme formulovat způsob interpretace výsledků při nemožnosti zkonstruování QFT arbitrární dimenze.

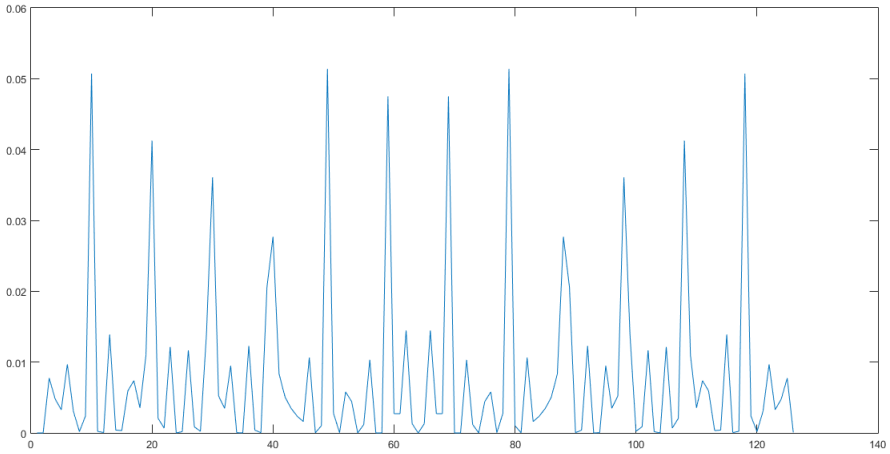
Diagramy níže pak odpovídají výslednému rozložení pravděpodobnosti naměření jednotlivých stavů registru pro daná N; nejprve čísla, která nejsou bezčtvercová



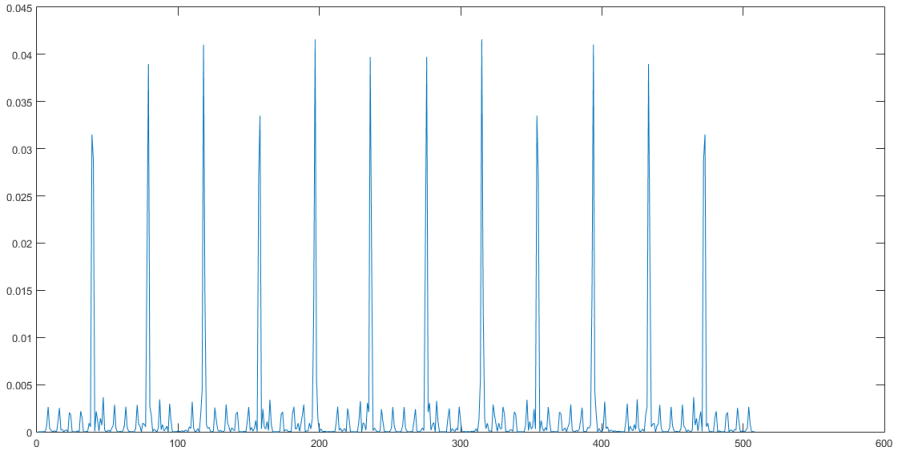
Obrázek 3.1: Případ pro $N = 63 = 7 \cdot 3^2$



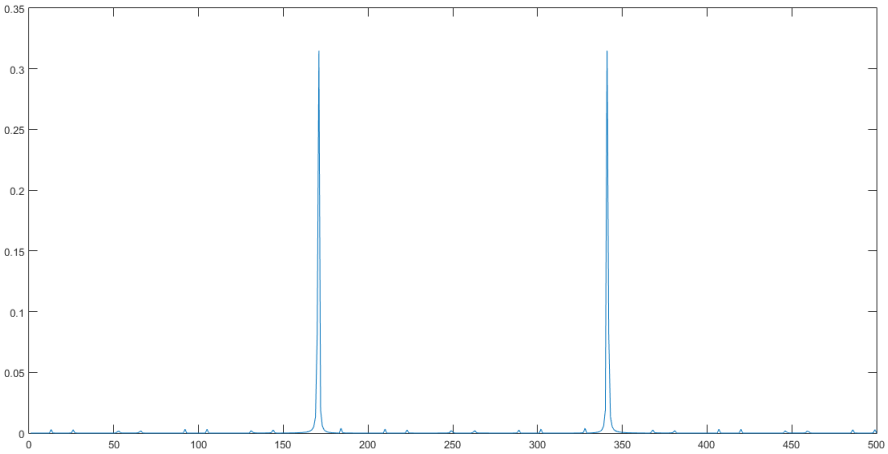
Obrázek 3.2: Případ pro $N = 75 = 3 \cdot 5^2$



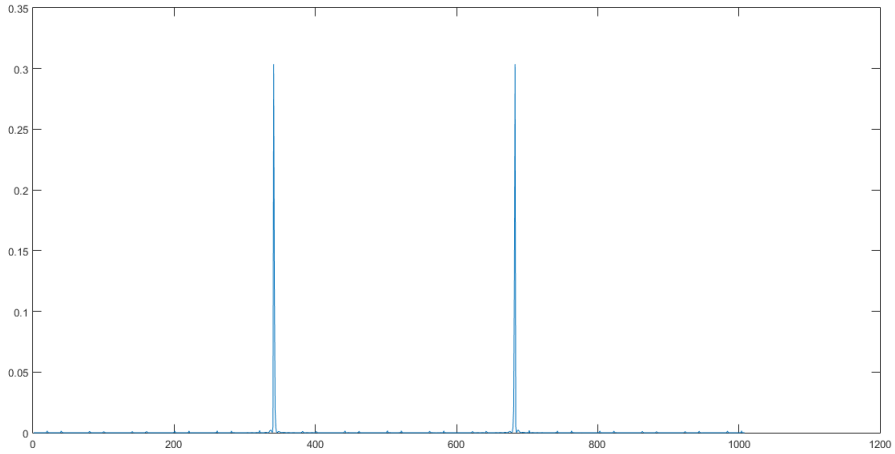
Obrázek 3.3: Případ pro $N = 117 = 17 \cdot 3^2$



Obrázek 3.4: Případ pro $N = 325 = 13 \cdot 5^2$

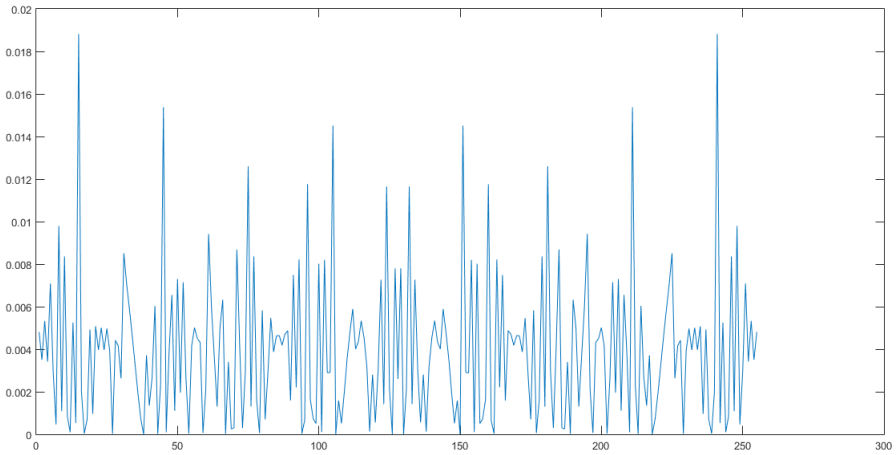


Obrázek 3.5: Případ pro $N = 507 = 3 \cdot 13^2$

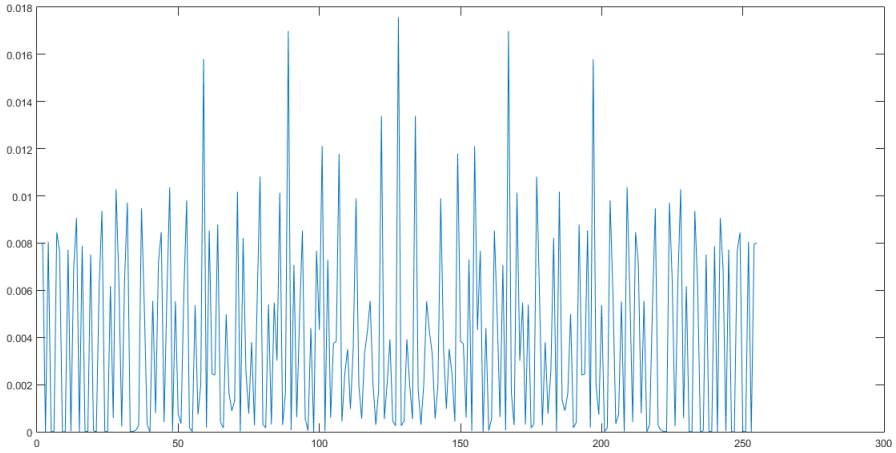


Obrázek 3.6: Případ pro $N = 867 = 3 \cdot 17^2$

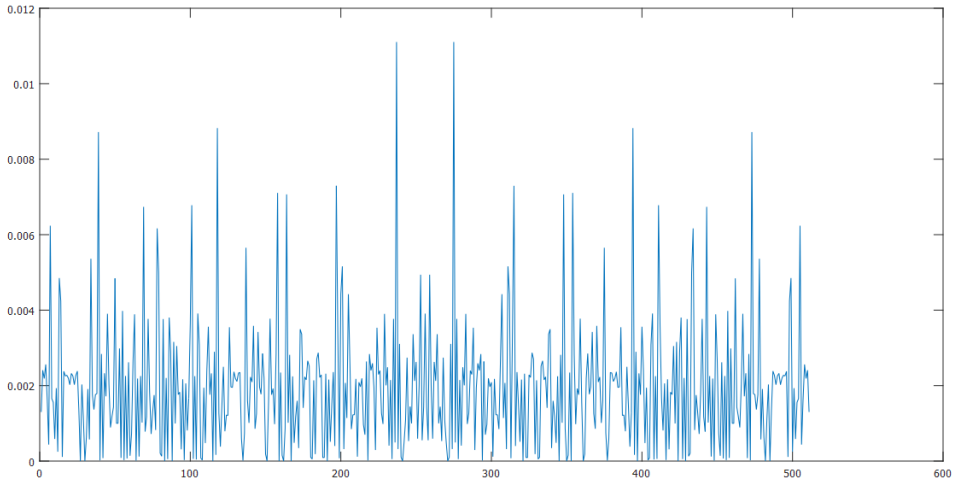
Naopak pro bezčtvercová čísla



Obrázek 3.7: Případ pro $N = 247 = 13 \cdot 19$



Obrázek 3.8: Případ pro $N = 255 = 3 \cdot 5 \cdot 17$



Obrázek 3.9: Případ pro $N = 323 = 17 \cdot 19$

Jak se tedy ukazuje, výsledný stav kvantového systému po provedení vyčíslení Gaussových sum do značné míry koresponduje s ideálním případem, jak jsme ho postulovali v teoretické části. Předem je zapotřebí přihlédnout ke skutečnosti, že hodnoty jsou souměrné přes „polovinu“ spektra, respektive přes $\frac{2^n}{2}$, kde n je velikost kvantového registru odpovídající algoritmu pro dané N; naměřené stavy větší než $\frac{2^n}{2}$ je nicméně možné jednoduše odečíst od 2^n, a tak získat správnou hodnotu.

Zcela podle našich prvotních očekávání je možné dále pozorovat, že v případě $N = pq^2$ nastává maximum amplitudy pravděpodobnosti v okolí jeho čtvercové části q^2, menší amplitudy pak odpovídají rozmístění násobků q. Tuto situaci lze nejlépe pozorovat v případech $N = 63$ a $N = 507$, což je možné jednoduše vysvětlit: Obě čísla se přibližují mocnině dvojky, resp. implementovaná 2^n dimenzionální kvantová Fourierova transformace do značné míry aproximuje příslušnou N-dimenzionální verzi. Tímto způsobem je možné následně interpretovat skutečnost, že v případech mocninám dvojky poněkud vzdálenějších čísel ($N = 75, 117, 867$) se rozložení amplitud pravděpodobnosti uvažovanému ideálnímu stavu rychle vzdaluje.

Přesto však předpokládáme možnost, jak lze navzdory odchylkám ve výsledném stavu dospět ke zdárnému řešení problému: Ukazuje se, že lokální i globální maxima amplitud jsou od sebe i přes posunutí celého spektra vzdálena o periodu, která s velkou pravděpodobností představuje hledaný faktor.[1] Pak je opakováním algoritmu možné tato maxima naměřit a následným odečtením hodnot získat číslo, které s jistou pravděpodobností sdílí s N společného dělitele. Jakkoli se pak podobný způsob může zdát poněkud neexaktním, vzhledem k již zmiňovanému posunutí celého spektra se právě periody amplitud ukazuje jako nejjistější způsob vedoucí k přibližnému určení oblasti, v níž se dané číslo sdílející s N netriviální faktor nachází.

Poslední dva diagramy pak ilustrují situaci pro N bezčtvercové. V porovnání s předchozími případy je rozložení amplitud pravděpodobnosti zjevně zcela odlišné povahy, kdy minima v okolí faktorů N „experimentálně“ dokazují správnost našich výchozích předpokladů. Jakkoli je při rozrůzněnosti pravděpodobností příslušných jednotlivým stavům nelehké konkrétní minima pozorovat, při detailní analýze se ukazuje, že kupříkladu pro $N = 255$ nastávají propady přesně v takových bodech, kdy $a \mid N$. Neboť číslo 255 přímo sousedí s osmou mocninou dvojky, diskutovaný případ jen dále potvrzuje shora formulovanou domněnku o možnosti aproximace operátoru QFT.

Výsledky simulací pro jednotlivá N ve formě tabulky, příslušných diagramů v plném rozlišení i *.mat* souboru pro práci s daty v prostředí MATLAB lze pak nalézt v příloze práce.

[1] De facto se nemusí nutně jednat o faktor, ale o každé takové číslo a, pro které $(a, q) > 1$; Eukleidovým algoritmem je pak možné násobek q jednoduše vypočítat.

Závěr

V teoretické části práce jsme se pokusili o nastínění problematiky kvantových výpočetních systému nejen po stránce příslušného matematického formalismu, ale na základní úrovni i z hlediska jejich fyzikální podstaty. Objasnili jsme specifické vlastnosti kvantových systémů, které vyplývají z jejich podstaty coby částic mikrosvěta, a představili jsme základní teoretické modely pro exaktní popis podobných systémů. Zabývali jsme se elementární jednotkou kvantových výpočtů, tj. qubitem a jeho možností znázornění jeho stavu jako vektoru v komplexním Hilbertově prostoru, stejně tak jsme však neopomněli charakteristiku kvantových počítačů přiblížit pomocí tří hlavních fyzikálních fenoménů, s nimiž se při konstrukci algoritmů nutně setkáváme, a sice masivního paralelismu, entanglementu a interference vlnových funkcí částic.

Dále jsme se zabývali možným řešením problému bezčtvercovosti čísla, kdy se jedná o jeden z těžkých, tzv. NP problémů, na který lze převést některé výpočty z oblasti teorie čísel. Jakkoli jsme ve své práci vyšli z myšlenky již dříve publikované v článku [2], problematiku se nám podařilo blíže rozpracovat, a tak zachytit v celé její komplexnosti: Spolu s nastíněním studované problematiky a zmíněním některých zajímavých vlastností bezčtvercových čísel jsme podali vlastní důkazy několika vztahů vyplývajících z Gaussových sum, jejichž specifických vlastností jsme při řešení problému dále využili.

Na matematických východiskách jsme následně vystavěli kvantový algoritmus, jehož jednotlivé kroky jsme z teoretického hlediska ozřejmili, a podrobnou analýzou jsme dospěli k závěru, že na kvantovém počítači lze problém bezčtvercovosti efektivně řešit v polynomiálním čase s pravděpodobností $p = 1$, díky čemuž je podané řešení možné zařadit do složitostní třídy *Exact Quantum Polynomial (EQP).*

Diskutovaný algoritmus jsme dále implementovali v jazyce QCL; nejprve jsme nastínili některé z možností urychlení kvantových výpočtů, realizovali obvody pro elementární matematické operace a následně navrhli efektivní postupy pro výpočet největšího společného dělitele (GCD) a Jacobiho sym-

bolu, kdy se v obou případech jedná o stěžejní funkce z oblasti teorie čísel.

Jakkoli jsme v teoretické části postulovali správnost výsledků s pravděpodobností $p = 1$ a možnost provedení výpočtu v polylogaritmickém čase, při následné implementaci algoritmu těchto charakteristik nedosáhli: Časová složitost vzrostla z $\mathcal{O}(\log^2 N)$ na $\mathcal{O}(N \log N)$ z důvodu nestability logaritmické kvantové sčítačky, které jsme pro dosažení uvažované časové efektivity zamýšleli využít; nakonec tak bylo nutné implementovat robustnější obvod s lineární časovou komplexitou.

Po nasimulování běhu algoritmu pro daná N se ukázalo, ža jakkoli výsledné rozložení amplitud pravděpodobnosti koresponduje s našimi původními předpoklady, je v některých případech možné pozorovat odchylky od ideálního stavu, které způsobuje nutnost aproximace N-dimenzionální kvantové Fourierovy transformace pomocí snáze implementovatelné 2^n dimenzionální verze. De facto jsme se tak setkali s podobným problémem jako v případě Shorova algoritmu, kdy obtížnost sestrojení právě N-dimenzionální QFT ve výsledku taktéž výrazně ovlivňuje charakteristiku implementovaného algoritmu.

Přesto jsme však navrhli možný způsob řešení, jak lze i při diskutovaných odchylkách od ideálního stavu problém bezčtvercovosti stále vyřešit; do budoucna se pak zaměříme právě na možnost algoritmické implementace QFT pro arbitrární velikost N a blíže analyzujeme chyby, s nimiž jsme se při testování logaritmické sčítačky setkali.

V intenzivním studiu problematiky kvantových výpočetních systémů pak plánujeme i nadále pokračovat; konkrétně se zamýšlíme zaměřit na zodpovězení otázky, zdali lze Gaussových sum využít i pro účel faktorizace libovolného čísla.

Literatura

[1] P. W. Shor, "Why haven't more quantum algorithms been found?," *Journal of the ACM (JACM)* **50**(1), pp. 87–90, 2003.

[2] J. Li, X. Peng, J. Du, and D. Suter, "An Efficient Deterministic Quantum Algorithm for the Integer Square-free Decomposition Problem," *ArXiv preprint 1108.5848* , Aug. 2011.

[3] B. Oemer, "Quantum Programming in QCL," *TU Wien* , 2000.

[4] "Quantum Turing machine. Encyclopedia of Mathematics." http://www.encyclopediaofmath.org/index.php?title=Quantum_Turing_machine&oldid=31935, prosinec 2015.

[5] D. Deutsch, "Quantum theory, the Church-Turing principle and the universal quantum computer," in *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, **400**(1818), pp. 97–117, The Royal Society, 1985.

[6] S. Aaronson, "Quantum Computing, Postselection, and Probabilistic Polynomial-Time," *ArXiv preprint quant-ph/0412187* , 2007.

[7] M. Lanzagorta and J. Uhlmann, *Quantum Computer Science (Synthesis Lectures on Quantum Computing)*, Morgan and Claypool Publishers, 11 2008.

[8] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition*, Cambridge University Press, 10 anv ed., 1 2011.

[9] J. Višňák, "Kvantově chemické algoritmy pro kvantové počítače," *MFF UK* , 2012. Diplomová práce.

[10] C. M. Dawson and M. A. Nielsen, "The Solovay-Kitaev algorithm," *ArXiv preprint quant-ph/0505030* , 2007.

[11] E. W. Weisstein, "Squarefree. From MathWorld–A Wolfram Web Resource." http://mathworld.wolfram.com/Squarefree.html, březen 2016.

[12] B. C. Berndt, R. J. Evans, and K. S. Williams, *Gauss and Jacobi Sums*, Wiley-Interscience, 1 ed., 6 1998.

[13] H. Cohen, *Number Theory: Volume I: Tools and Diophantine Equations (Graduate Texts in Mathematics)*, Springer, 2007 ed., 5 2007.

[14] C. Moore and M. Nilsson, "Parallel Quantum Computation and Quantum Codes," *ArXiv preprint quant-ph/9808027* , 2009.

[15] "Quantum Fourier Transformation / Schéma obvodu." https://en.wikipedia.org/wiki/Quantum_Fourier_transform, březen 2016.

[16] T. G. Draper, S. A. Kutin, E. M. Rains, and K. M. Svore, "A logarithmic-depth quantum carry-lookahead adder," *Quant. Inf. Comp. Vol.* **6,**, pp. No.4–5,pp.351–369, 2006.

[17] V. Vedral, A. Barenco, and A. Ekert, "Quantum Networks for Elementary Arithmetic Operations," *ArXiv preprint quant-ph/9511018* , 2009.

[18] S. A. Cuccaro, T. G. Draper, S. A. Kutin, and D. P. Moulton, "A new quantum ripple-carry addition circuit," *ArXiv preprint quant-ph/0410184* , 2007.

[19] M. Saeedi and I. L. Markov, "Quantum Circuits for GCD Computation with $O(n \log n)$ Depth and O(n) Ancillae," *ArXiv preprint 1304.7516* , Apr. 2013.

[20] J. Shallit and J. Sorenson, "A Binary Algorithm for the Jacobi Symbol," *ACM SIGSAM Bulletin* **27**, pp. 4–11, 1993.

[21] W. van Dam and G. Seroussi, "Efficient Quantum Algorithms for Estimating Gauss Sums," *ArXiv preprint quant-ph/0207131* , 2007.

Příloha A. Seznam použitého softwaru

Pro vypracování práce jsme využili následujících programů

Program	**Účel**
Quantum Computation Language 0.6.4	Simulace algoritmu
Cygwin 2.0.4	Emulace prostředí Unixu pro běh QCL
MathWorks MATLAB r2015a	Vykreslení grafů
Visual Studio 2015 Community	IDE programu pro dílčí analýzy chodu algoritmu
QTool	Vlastní program v C# pro provádění analýz
LyX 2.1	Textový procesor
QCircuit	Vykreslení kvantových obvodů v LaTeX

Vlastní program QTool se pak nachází v příloze práce.

Příloha B. Zdrojové kódy v QCL

Přiložené soubory obsahují zdrojové kódy odpovídající implementaci výše prezentovaných kvantových funkcí v jazyce QCL. Simulační prostředí a interpret zdrojových kódů lze stáhnout online na adrese QCL - A Programming Language for Quantum Computers, pro případ využití programu na systému Windows je možné postupovat podle návodu dostupného na Running QCL on Windows. Pro vyčerpávající referenci si dovolíme odkázat na předchozí uvedenou adresu, pouze pak zmiňme, jak lze zdrojové kódy do programu „nahrát“: Soubory s kódy je možné načíst pomocí příkazu `include "soubor.qcl"`.

Důvod, proč lze každý soubor nalézt ve dvou verzích, je ten, že při testování algoritmu jsme se opakovaně setkali s chybou na straně QCL v podobě špatného „garbage collectoru“, je-li tak možné o kvantových registrech hovořit. Jednoduše řečeno, ukázalo se zapotřebí všechny používané registry deklarovat již na počátku algoritmu, neboť při dynamickém vytváření instancí v průběhu výpočtu se stávalo, že dva různé registry ve skutečnosti ukazovaly na stejné místo v paměti. Adresář *„puvodni“* tak obsahuje optimalizované kódy podle návrhu výše, *„upravene“* pak kódy zjednodušené pro bezchybný běh v QCL.

Příloha C. Výsledky simulací

K práci dále přikládáme výsledky simulací chodu algoritmu pro jednotlivá N. Data lze nalézt jak v tabulce ve formátu *.csv*, tak i vizualizované podobě ve formátu *.png*. Všechna data lze pak jednoduše načíst do prostředí MATLAB pomocí souboru *„matlab_data.mat“*.

Podotkněme, že hodnoty odpovídají pravděpodobnostem naměření příslušných stavů, podle definice se tak jedná o druhé mocniny amplitud pravděpodobnosti. Přestože jsme v kapitole věnované závěrečnému testování oba pojmy zaměňovali, činili jsme tak s ohledem na výsledné rozložení amplitud, které je v obou případech stejné. Amplitudy pravděpodobnosti mohou ve skutečnosti nabývat i záporných hodnot, čímž by se obě spektra zjevně odlišovala, nicméně v případě simulace má pro nás skutečný význam pouze pravděpodobnost naměření daných stavů.