Kvantumszámítógépes algoritmusok

Hallgatói jegyzetek

Ivanyos Gábor

előadásai alapján

Debreceni Egyetem, 2011 tavaszi félév

Tartalomjegyzék

1. fejezet

Bevezetés

A 2011.03.22-iki előadás alapján lejegyezte Barnák Albert

Richard Phillips Feynman 1982 felvetése: ki lehet-e használni az informatikában, hogy a kvantum és a valós világ teljesen eltér? Vannak kvantumjelenségek, amelyeket nem lehet klasszikus számítógépen (hatékonyan) szimulálni.

Áttörés: Peter Shor 1994: egészek faktorizációja kvantum-számítógéppel polinomidőben.

Charles H. Benett, Gilles Bassard 1984: Kvantum-kriptogáfia (kulcscsere). Nyilvános csatornákon történő titkosításra lehet használni. Jelenleg kvantum-kriptogáfia az egyetlen olyan olyan fejlemény a témakörben, amely a gyakorlatban hasznosítható, ami eszközökben létezik. Fotonpárok küldése Genfi-tó alatt, svájci népszavazás titkosítása. Kvantumtitkosítás van, kvantumszámítógép egyelőre nincs, és nem világos, hogy lesz-e valaha.

1.1. n dimenziós kvantumrendszer

$\sim \mathbb{C}^n$

A rendszer állapota egy n dimenziós egységvektor: $v \in \mathbb{C}^n$, amelyre $|v| = 1$.

Az előadássorozatban a Dirac-féle jelölést (ami sok minden másra is jó) az i-edik bázisvektor jelölésére használjuk: $v_i = |i\rangle$, ekkor

$$v = \sum_{i=1}^{n} \alpha_i \cdot |i\rangle$$

Úgy interpretáljuk, hogy a v vektor egyszerre van az összes lehetséges állapotban, valamilyen α_i súllyal. Az α_i együtthatókat amplitúdóknak nevezzük.

Mérés: Ha mérjük (más kifejezéssel megfigyeljük) a v vektort, akkor $|i\rangle$ -t $|\alpha_i|^2$ valószínűséggel kapjuk.

Egyenétékű állapotok: Ha $v' = \epsilon v$ valamely $\epsilon \in \mathbb{C}$ skalárra, akkor méréssel nem tudjuk v-t és v'-t megkülönböztetni (részletesebben ld. a kvantum-biteknél).

1.1.1. Műveletek állapotokon

- lineáris transzformáció legyen az n dimenziós térnek

- hossztartó legyen

Ha az előző kettő teljesül, akkor unitér transzformáció, azaz olyan transzformáció, amelynek U a mátrixára $UU^* = I$ teljesül, ahol $U^* = \overline{U^T}$.

Egyenétékű állapotok: Ha $v' = \epsilon v$ valamely $\epsilon \in \mathbb{C}$ skalárra, akkor méréssel nem tudjuk v-t és v'-t megkülönböztetni. (Az ilyen ϵ-t szokás globális fázisként interpretálni.)

1.1.2. Kvantum bitek (qubit, kubit)

A $\mathbb{C}^2$ két dimenziós komplex vektortér sztenderd bázisa, $|0\rangle$, $|1\rangle$

Az állapot egy egységvektor: $\alpha|0\rangle + \beta|1\rangle$, ahol $|\alpha|^2 + |\beta|^2 = 1$

a mérésen kívül alkalmazhatunk rájuk unitér transzformációkat.

1.1.3. Példák egy qubites műveletekre

Fázismanipuláció. Sztenderd bázisban felírt mátrixok: $\begin{pmatrix} 1 & \\ & -1 \end{pmatrix}$, $\begin{pmatrix} 1 & \\ & \epsilon_1 \end{pmatrix}$, $\begin{pmatrix} \epsilon_0 & \\ & \epsilon_1 \end{pmatrix}$

Ezekkel a komponensek amplitúdóját manipuláltuk. Természtesen $|\epsilon_i| = 1$.

A harmadik művelet a másodiknak skalárszorosa, ezért hatása lényegében ugyanaz (egyenétékű állapotot eredményez):

$$\epsilon(\alpha|0\rangle + \beta|1\rangle) = \epsilon\alpha|0\rangle + \epsilon\beta|1\rangle$$

és

$$|\epsilon\alpha|^2 = |\alpha|^2$$
$$|\epsilon\beta|^2 = |\beta|^2$$

Amiből az következik, hogy ugyanazt a valószínűségi eloszlást kapjuk, mint az ϵ -nal való szorzás előtt.

Az Hadamard-transzformáció. Ez a kvantumszámítógépeknél kiemelten fontos művelet.

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

ahol az $\frac{1}{\sqrt{2}}$ a normáló faktor.

Tulajdonságok:

A H mátrix alakulása a $|0\rangle$ báziselemre vonatkozóan

$$H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

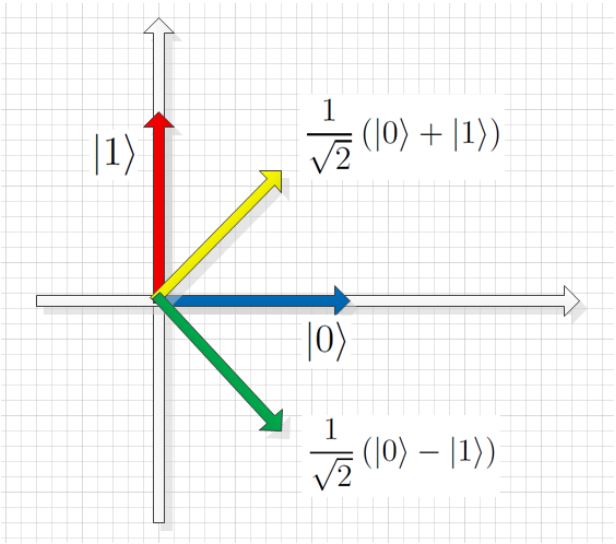
Ez a $|0\rangle$ és $|1\rangle$ uniform szuperpozíciója.

Kezdetben van a $|0\rangle$ állapot, ezt akárhogy mérjük mindig a 0-t kapjuk.

$$|0\rangle \xrightarrow{H} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

megmérjük az uniform szuperpozíciót, $\frac{1}{2}$, $\frac{1}{2}$ valószínűséggel kapjuk 0-t és 1-t.

Determinisztikus állapotból kvantum állapotot csinál, egyszerre van 0-ban és 1-ben is.



1.1. ábra. Az Hadamard-transzformáció: tükrözés a 22.5 fokos egyenesre

1.2. A BB84 protokoll

A BB84 egy kvantum kulcscsere séma, amit Charles Bennett és Gilles Brassard fejlesztett ki 1984-ben. A sémában Alice el kívánja küldeni Bobnak a saját privát kulcsát.

Az eljárás a következőképpen néz ki:

1. Alice vesz egy véletlen bitet (ami 0 vagy 1), ennek megfelelő sztenderd bázisvektort $\Phi = |0\rangle$ vagy $\Phi = |1\rangle$.

2. Alice vesz még egy véletlen bitet (ami 0 vagy 1),

$$\Phi' = \begin{cases} \Phi, & \text{ha ez a bit 0} \\ H\Phi, & \text{ha ez a bit 1.} \end{cases}$$

3. Alice elküldi a Φ' kvantumbitet Bobnak.

4. Bob sorsol egy véletlen bitet:

$$\Phi'' = \begin{cases} \Phi', & \text{ha ez a bit 0} \\ H\Phi', & \text{ha ez a bit 1.} \end{cases}$$

5. Bob megméri a Φ'' állapotot.

6. Bob megtárgyalja Alice-szal (egy nyilvános csatornán), hogy mi volt Alice második véletlen illetve Bob véletlen bitje.

   (=) Ha ezek egyeznek, akkor Alice és Bob megegyeztek a bitben.

   ($\neq$) Ha nem egyeznek, akkor mindketten eldobják és újrapróbálkoznak.

A (=) esetben $\Phi'' = \Phi$.
A ($\neq$) esetben $\Phi'' = H\Phi$. Ha $\Phi = |0\rangle$, akkor $\Phi'' = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$. Ha pedig $\Phi = |1\rangle$, akkor $\Phi'' = \frac{1}{\sqrt{2}}(1 \cdot |0\rangle + (-1) \cdot |1\rangle)$. Az 5. lépésbeni mérés utáni bit 50-50%-ban egyezik meg Alice első bitjével.
Ugyanez vonatkozik a külső megfigyelőre is. Ha a rossz mérést csinálta, akkor 50-50%-ban 0-1 bitje van. Csak akkor látta meg a jó bitet, ha ő is eltalálta Alice második bitjét, erre 50% esélye van.

Amikor Bob eltalálta Alice második véletlen bitjét, kettőjüknek lett egy "fél" titkos bitje. Ha ezt k -szor ismételjük, akkor k "fél" titkos bit keletkezett. A "fél" tikos bitek hasznosak: k ilyen bit XOR-ját képezve $1 - 2^k$ valószínűséggel egy "igazi" tikos bit kapható, de vannak gazdaságosabb módszerek, amelyekkel tényleg kb. $\frac{k}{2}$ igazi titkos bit nyerhető.

2. fejezet

Két bites rendszerek

A 2011 március 22-iki előadás alapján írta Simon Béla

2.1. Két bites rendszerek

A standard báziselemek

$$\begin{aligned}|00\rangle &= |0\rangle|1\rangle \\ |01\rangle &= |0\rangle|1\rangle \\ |10\rangle &= |1\rangle|0\rangle \\ |11\rangle &= |1\rangle|1\rangle\end{aligned}$$

négy dimenziós teret alkotnak: $\mathbb{C}^4 = \mathbb{C}^2 \otimes \mathbb{C}^2$

Uniform szuperpozíció:

$$\frac{1}{2}\left(|0\rangle|0\rangle + |0\rangle|1\rangle + |1\rangle|0\rangle + |1\rangle|1\rangle\right)$$

$|0\rangle|0\rangle$-ból ez a következõképp állítható elõ:

$$|0\rangle|0\rangle \overset{\text{H elsõ részre}}{\rightarrow} \frac{1}{\sqrt{2}}\begin{pmatrix}|0\rangle \\ + \\ |1\rangle\end{pmatrix}|0\rangle \overset{\text{H második részre}}{\rightarrow} \frac{1}{2}\begin{pmatrix}|0\rangle \\ + \\ |1\rangle\end{pmatrix}\begin{pmatrix}|0\rangle \\ + \\ |1\rangle\end{pmatrix}$$
$$= \frac{1}{2}\left(|0\rangle|0\rangle + |0\rangle|1\rangle + |1\rangle|0\rangle + |1\rangle|1\rangle\right)$$

2.1.1. Szorzatállapotok

Speciális vektorok a trenzorszorzat-térben $\phi \in \mathbb{C}^2$, $\psi \in \mathbb{C}^2$, amelyek $\phi \otimes \psi$ alakúak.

Nem csak a báziselemek tenzorszorzata tartozik ide, hanem a báziselemek lineáris kombinációját is képezhetjük az elsõ térbõl, a második térbõl és disztributivitás alapján számolhatjuk a trenzorszorzatuk felírását a standard bázisban.

2.1.2. Összefonódott állapotok

Nem minden vektor szorzatalakú!
Például:

$$\frac{1}{\sqrt{2}}\left(|0\rangle|0\rangle + |1\rangle|1\rangle\right)$$

nem áll elõ szorzatként.
Ez az Einstein–Podolsky–Rosen-paradoxonból ismert úgynevezett EPR-pár. Az EPR paradoxon a kvantummechanika egyik nevezetes gondolatkísérlete, amelynek eredeti célja az elmélet nemteljességének demonstrálása volt, késõbb pedig a kísérleti ellenõrzésben játszott szerepet[1]. Egy két bites rendszer, ahol a két qubit egymástól távol helyezkedik el, nem szorzatalakúak (az állapotok nem függetlenek). Összefonódott állapot köti össze a két rendszert, annak ellenére, hogy távol vannak egymástól.
Látni fogjuk, hogy egy ilyen összefonódott pár segítségével küldhetõ át egy qubit egyik helyrõl a másikra.

2.1.3. A Bell-állapotok

Négy dimenziós térben speciális, egymásra merőleges egységvektorok (ortonormált bázist alkotnak).

$$\phi^+ = \frac{1}{\sqrt{2}}\left(|0\rangle|0\rangle + |1\rangle|1\rangle\right)$$
$$\phi^- = \frac{1}{\sqrt{2}}\left(|0\rangle|0\rangle - |1\rangle|1\rangle\right)$$
$$\psi^+ = \frac{1}{\sqrt{2}}\left(|0\rangle|1\rangle + |1\rangle|0\rangle\right)$$
$$\psi^- = \frac{1}{\sqrt{2}}\left(|0\rangle|1\rangle - |1\rangle|0\rangle\right)$$

A Bell-állapotok összefonódott állapotok, mindegyikük a lehető legnagyobb távolságra van a szorzatállapotoktól.
A ϕ^+ állapot tulajdonképpen az EPR-pár.

[1] http://hu.wikipedia.org/wiki/EPR-paradoxon

2.1.4. Nem lehet klónozni,

azaz egy qubitet nem lehet lemásolni egy másikba.
Állítás: $\nexists U : \mathbb{C}^{2x2} \rightarrow \mathbb{C}^{2x2}$ unitér transzformáció, hogy $\forall \phi \in \mathbb{C}^2$
$\phi \otimes |0\rangle \xrightarrow{U} \phi \otimes \phi$
(Azaz nem megy, hogy az üres hely helyére bemásoljuk a ϕ-t.)

Bizonyítás: Alkalmazzuk a kívánt szabályt a következő állapotokra:

$$\phi = |0\rangle, |1\rangle, \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

Ha nullára, illetve egyre végezzük:

$$|0\rangle|0\rangle \xrightarrow{U} |0\rangle|0\rangle$$
$$|1\rangle|0\rangle \xrightarrow{U} |1\rangle|1\rangle$$

A linearitás miatt a kettõ kombinációjából

$$\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)|0\rangle \rightarrow \frac{1}{\sqrt{2}}(|0\rangle|0\rangle + |1\rangle|1\rangle) \neq \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

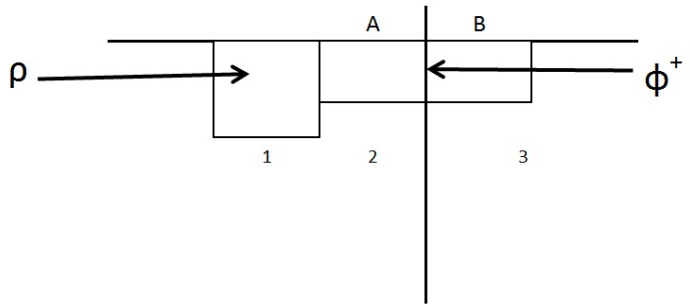
Annyire nem egyelők, hogy a baloldalon egy összfonódott állapot áll, míg a jobboldalon egy szorzatállapot

2.2. Teleportálás

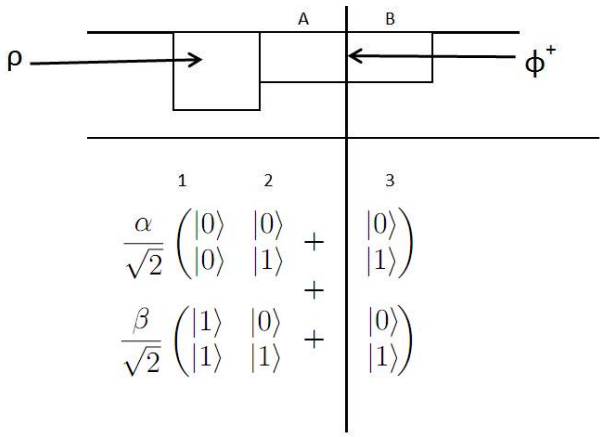
Egy kvantumbitet szeretnénk átvinni egyik helyrõl a másikra.

- Klónozás nincs
- Megmérni nem tudjuk (0-t vagy 1-et kapunk)
- Két klasszikus bit átküldésével végezzük (Egy EPR ϕ^+ állapot a két rész között szétosztva)

A és B között összefonódott 2 bit áll. A rendszer három bites, az elsõ és második bit között szorzatállapot áll. Alize szeretné átküldeni a ρ kvantum-állapotot Bob-nak.
$\rho = \alpha|0\rangle + \beta|1\rangle$
Az összeg tagjai kifejtve:
ρ lehet 0 vagy 1, a ϕ^+ szerint két lehetõség: 0 0, vagy 1 1, ez négy tagú összeg.



2.1. ábra. Összefonódott bitek



2.2. ábra. Négytagú összeg

Az irodalomban ismertetett változatokban úgynevezett Bell-mérést alkalmaznak. Ezen a kurzuson a méréseknek csak egy szűk körét definiáltuk, ezért a itt a Bell-mérést helyettesítjük egy transzformációval és a definiált nem egészen általános méréssel.
T transzformáció:

$$|0\rangle|0\rangle \rightarrow \phi^+$$
$$|1\rangle|1\rangle \rightarrow \phi^-$$
$$|0\rangle|1\rangle \rightarrow \psi^+$$
$$|1\rangle|0\rangle \rightarrow \psi^-$$

Ezt a T transzformációt az elsõ két qubiten végezzük el, szemléletesen az elsõ qubitet összefonódtatjuk a másodikkal és a harmadikkal. Az eredmény:

$$
\begin{aligned}
&\frac{\alpha}{2}|0\rangle|0\rangle|0\rangle \\
+&\frac{\alpha}{2}|1\rangle|1\rangle|0\rangle \\
+&\frac{\alpha}{2}|0\rangle|1\rangle|1\rangle \\
+&\frac{\alpha}{2}|1\rangle|0\rangle|1\rangle \\
&\frac{\beta}{2}|0\rangle|1\rangle|0\rangle \\
-&\frac{\beta}{2}|1\rangle|0\rangle|0\rangle \\
+&\frac{\beta}{2}|0\rangle|0\rangle|1\rangle \\
-&\frac{\beta}{2}|1\rangle|1\rangle|1\rangle
\end{aligned}
$$

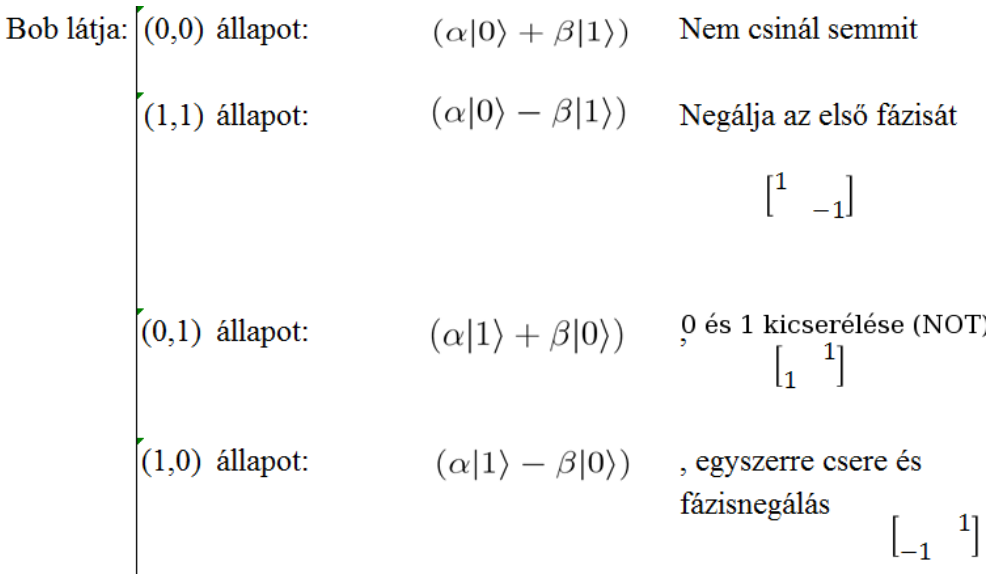
Azonos átalakítással kapjuk:

$$
\begin{aligned}
&\frac{1}{2}|0\rangle|0\rangle\left(\alpha|0\rangle+\beta|1\rangle\right) \\
&\frac{1}{2}|1\rangle|1\rangle\left(\alpha|0\rangle-\beta|1\rangle\right) \\
&\frac{1}{2}|0\rangle|1\rangle\left(\alpha|1\rangle+\beta|0\rangle\right) \\
&\frac{1}{2}|1\rangle|0\rangle\left(\alpha|1\rangle-\beta|0\rangle\right)
\end{aligned}
$$

A harmadik qubiten álló vektorok egségvektorok, ezért ha az első két bitet megmérjük, a négy lehetséges eredmény $(00, 01, 10, 11)$ mindegyike azonos valószínûséggel fordul elõ. Elsõ két bit mérése, átküldése Bob-nak (2.3. ábra). Az ábra harmadik oszlopában szereplő transzformációk segítségével Bobnál előáll mind a négy esetben Alice erdeti qubitje.
Összességében szükséges volt:

- EPR pár
- Két klasszikus bit küldése
- Alice két bitje a mérések következtében elromlott, Bob EPR pár része lett Alice eredeti qubitje.

Általános elv kvantum-kommunikációban: Egy qubit két klasszikus bitet ér.



2.3. ábra. Elsõ két bit mérése, átküldése Bob-nak

3. fejezet

Randomizált nyelvosztályok

A 2011 03.22-iki előadás alapján írta László Árpád

3.1. Függvények és nyelvek kapcsolata

Adott az f függvény, amelynek a bemenete legyen x. Feladat: kiszámítani az $f(x)$ értékét. Függvények bonyolultsága helyett nyelvek (igen-nem problémák) bonyolultságát fogjuk vizsgálni. Hogyan lehet áttranszformálni a függvények kiszámítási problémáját nyelvek eldöntési problémájára?

I. Ötlet

Tekintsük az $f(x)$ függvény bitjeit. Feltételezzük, hogy adott egy korlát a bitek számára vonatkozólag. Ekkor a korláttal megegyező számú felismerési problémára vezetjük vissza a függvény kiszámítását. Ez az eljárás általában nem természetes.

Pl.: Egész számok faktorizációja - prímek nyelve

Legyen $x \in \mathbb{N}$, $f(x)$ az x prímtényezős felbontása, $x \in \mathbb{P}$ (ha x-nek nincs 1-től és x-től különböző osztója).
A *prímek nyelve*

$\in coNP$: osztó mint tanú

$\in NP$: Pratt-tétele, rekurzív tanú primitív elemekkel

$\in coRP$: Rabin-Miller teszt, Solovay–Strassen teszt

$\in P$: Agrawal-Kayal-Saxena (a prímség eldönthető polinom időben)

3.2. Az RP nyelvosztály

$L \in RP$ (Randomized polynomial time), ha $\exists M$ véletlent használó Turing-gép és $\exists c > \frac{1}{2}$(konstans), hogy

- ha $x \in L$, akkor $Pr(M$ elfogadja x-et$) \geq c$
- ha $x \notin L$, akkor M nem fogadja el x-et

Feltételezve, hogy M polinomidejű.
Tehát az "*igen*" válaszban megbízhatunk biztos, hogy nem téved. A "*nem*" válasz esetén, kevesebb mint 50% a hiba valószínűsége.

II. Ötlet

Hasonló a függvény bitjeinek kiszámításához, de most próbáljuk megkeresni az x-nek a legkisebb prímosztóját. A nyelv: $\{(x, y)|x$-nek $\exists y$-nál nem nagyobb prímosztója$\}$. Az x szám prímtényezős felbontására és pl. az AKS-prímtesztre támaszkodva könnyű látni, hogy ez a nyelv $NP \cap coNP$-be esik.
Az eldöntendő probléma az, hogy van-e ilyen x-nek y-nál kisebb prímosztója? Az első iteráció során megnézzük, hogy van-e x-nek a felénél nem nagyobb prím osztója. Ha van akkor kereshetjük tovább, ha nincs akkor az x egy prímszám. A következő lépésben y-nak x negyedét választjuk. Ha van annál kisebb osztója x-nek, a következő y érték $x/8$ lesz, ha nincs, akkor $3x/8$. Így folytatva bináris kereséssel $O(\log x)$ menetben megtaláljuk x legkisebb prímosztóját.

Fenti pédák mutatják, hogy függvények kiszámítási problémája gyakran hatékonyan visszavezethető eldöntési problémákra.

A Las Vegas nyelvosztály

$LasVegas = RP \cap coRP$.
Tehát van egy RP algoritmusunk aminek a "*nem*" válasza nem megbízható, valamit van egy $coRP$ algoritmusunk aminek a "*nem*" válasza megbízható. Ha a kettőt egymás mellé rakjuk, akkor egy olyan algoritmust kapunk, amely 50% valószínűséggel megbízható "*igen*" vagy "*nem*" választ ad, vagy pedig "*passz*", azaz nem sikerült a problémát eldönteni.
Az ilyen algoritmus esetén azt mondhatjuk, hogy egyik oldalon sincs hiba. RP illetve $coRP$ esetén az "*igen*" vagy a "*nem*" ágon elképzelhető hiba.

3.3. A BPP nyelvosztály

$L \in BPP$ (Bounded-error probabilistic polynomial), ha $\exists M$ véletlent használó polinomidejű Turing-gép és $\exists c > \frac{1}{2}$(konstans), hogy

- ha $x \in L$, akkor $Pr(M$ elfogadja x-et$) \geq c$

- ha $x \notin L$, akkor $Pr(M$ elutasítja x-et$) \geq c$

Tehát legalább c valószínűséggel megkapjuk a helyes választ. Ha k alkalommal ismételjük az algoritmust akkor a döntés az ami többször jön ki mint eredmény. (Hiba becslés: Chernoff-korlát, annak a valószínűségét becsli meg, hogy egy $c > \frac{1}{2}$ valószínűségű esemény k-szor mintavételezve milyen valószínűséggel ad a k eset felében vagy még több esetben rossz választ. Ez exponenciálisan kicsi lesz.)
Ezek azok a nyelvek amelyek hatékonyan felismerhetők randomizált algoritmussal.
Megjegyzés: az RP illetve Las Vegas nyelvosztályok esetében a $c > \frac{1}{2}$ feltevés nem lényeges, tetszőleges $c > 0$ is megteszi. Ismétléssel a hiba exponenciálisan csökkenthető. A BPP osztály esetében azonban (hacsak nincs váratlan egybeesés a bonyolultsági osztályok között), még az is lényeges, hogy a hiba valószínűsége az $\frac{1}{2}$-től el van választva:

A PP nyelvosztály

$L \in PP$ (Probabilistic polynomial time), ha $\exists M$ véletlent használó polinomidejű Turing-gép, hogy

- ha $x \in L$, akkor $Pr(M$ elfogadja x-et$) > \frac{1}{2}$

Megj.: $PP = coPP$

Állítás: $NP \subseteq PP$, pl.: $SAT \in PP$

Bizonyítás: Veszünk egy véletlen behelyettesítést,

- ha ez kielégíti a formulát akkor elfogadjuk.
- ha nem akkor $50 - 50\%$-ban sorsoluk az elfogadás és az elutasítás között.

Tehát ha a formulánk kielégíthető akkor van 50% esélyünk, hogy elfogadjuk plusz még egy exponenciálisan kicsi, hogy éppen egy kielégítő behelyettesítés húztunk.

3.4. A BQP nyelvosztály

$L \in BQP$ (Bounded-error quantum polynomial time), ha $\exists C_n$ $LOGSPACE$-uniform kvantum-áramkörsorozat (ld. a következő fejezetet), amelynek a hossza és a tármérete $n^{O(1)}$, valamint $\exists c > \frac{1}{2}$(konstans), hogy

- ha $x \in L$, akkor $Pr(C_n$ elfogadja x-et$) > c$
- ha $x \notin L$, akkor $Pr(C_n$ elutasítja x-et$) > 0$

A $LOGSPCE$-uniformitás azt jeleni, hogy van olyan 3 szalagos – kitüntetett input-, output-, illetve munkaszalaggal ellátott – Turing-gép, amely ha n-t kapja a bemeneti szalagján, az outputszalagra ráírja C_n áramkört (a kaput sorozatát a lábak kiosztásával együtt), úgy, hogy közben csak $O(\log n)$ munkaterületet használ. Uniformitás nélkül akár (Turing-géppel) eldönthetetlen problémákat is meg tudnánk oldani kvantum (sőt, Boole-) hálózattal: legyen L egy tetszőleges bináris nyelv és legyen C_n írja a kimeneti bitre azt, hogy 1 vagy 0, aszerint, hogy n bináris számjegyeinek a sorozata benne van-e L-ben vagy sem.

4. fejezet

Kvantum-hálózatok/áramkörök

A 2011.04.05.-i előadás alapján lejegyezte Vitéz László

4.1. Mátrixok tenzorszorzata

1. Definíció. *Mátrixok tenzorszorzatán*[1] *azt a mátrixot értjük, amelynek elsõ blokkja úgy áll elõ az adott A és B mátrixokból, hogy az A mátrix elsõ sorának elsõ elemével megszorozzuk a B mátrixot. Ezt követõen az A mátrix elsõ sorának második elemével megszorozzuk a B mátrixot, ami a tenzorszorzat elsõ sor második blokkját adja eredményül. Ezen lépéseket végrehajuk A összes elemére. Az eredménymátrix lesz a tenzorszorzat. Jelölése* $A \otimes B$.

A definícióból következik, hogy az eredménymátrix dimenziója megegyezik a két összeszorzandó mátrix dimenziójának szorzatával:$\mathbb{C}^s \otimes \mathbb{C}^t \equiv \mathbb{C}^{s \cdot t}$. Ezeknek megfelelõen felírhatjuk A tetszõleges, és az I identitásmátrix tenzorszorzatát.

$$A \otimes I = \begin{pmatrix} \begin{pmatrix} a_{1,1} & 0 & \cdots & 0 \\ 0 & a_{1,1} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & a_{1,1} \end{pmatrix} & \cdots & \begin{pmatrix} a_{1,n} & 0 & \cdots & 0 \\ 0 & a_{1,n} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & a_{1,n} \end{pmatrix} \\ \vdots & \ddots & \vdots & \\ \begin{pmatrix} a_{n,1} & 0 & \cdots & 0 \\ 0 & a_{n,1} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & a_{n,1} \end{pmatrix} & \cdots & \begin{pmatrix} a_{n,n} & 0 & \cdots & 0 \\ 0 & a_{n,n} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & a_{n,n} \end{pmatrix} \end{pmatrix},$$

[1] `http://www.math.bme.hu/~sszabo/NumerikusSzimbolikus/MatrixTenzorMultiply.pdf`

valamint

$$I \otimes A = \begin{pmatrix} \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & & \\ \vdots & & \ddots & \vdots \\ a_{n,1} & \cdots & & a_{n,n} \end{pmatrix} & 0 & \cdots & 0 \\ 0 & \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & & \\ \vdots & & \ddots & \vdots \\ a_{n,1} & \cdots & & a_{n,n} \end{pmatrix} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & & \\ \vdots & & \ddots & \vdots \\ a_{n,1} & \cdots & & a_{n,n} \end{pmatrix} \end{pmatrix}$$

Mátrixok fent definiált tenzorszorzása nem kommutatív. (Viszont a felcserélt tényezőkkel képzett tenzorszozat egy az eredetihez *hasonló* mátrixot eredményez – annak egy permutációmátrix szerinti kunjugáltja lesz.) A tenzorszorzás további tulajdonsága, hogy unitér mátrixok tenzorszorzata is unitér. Ezen tulajdonságot szemlélteti az alábbi példa.

1. Példa. *Tenzorszorozzuk össze a H Hadamard mátrixot saját magával!*

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

$$H \otimes H = \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}$$

4.2. Kvantum-hálózatok

A kvantum kapuk 1,2,3,4... konstans sok biten lévõ unitér transzformációknak tekinthetõek, melyek $U \otimes I$ identitásként hatnak az áramkörön.*(A hálózatot kapuk sorozata alkotja, melyek közötti kapcsolat drótozással alakítható ki.)* Formálisan:

$$(U_l \otimes I)(U_{l-1} \otimes I)...(U_2 \otimes I)(U_1 \otimes I)|input\rangle|munkaterlet(0...0)\rangle$$

Itt az $U_j \otimes I$ jelölés kissé pontatatlan: valójában a teret a j-edik kapu "lábai" és a maradék qbitek szerinti terek tenzorszozatára bontjuk, és eszerint képezzük az U_j és az I

tenzorszorzatát. Az eredmény egy ψ kvantumállapot, melyet a számolás végén megmérünk. $\psi = \sum_{s\in\{0,1\}^k} \alpha_s|s\rangle$, azaz s-et$|\alpha_s|^2$ valószínúséggel kapjuk. Nyelv felismerésénél nem fontos az egész s, annak csupán a kitüntetett 0-ik bitje, az output bit.

$$output = \begin{cases} 0, & p = \sum_{r=\{0,1\}^{k-1}} |\alpha_{0r}|^2 \text{ valószínûsséggel } \textbf{elutsítjuk} \text{ a bemenetet} \\ 1, & p = \sum_{r=\{0,1\}^{k-1}} |\alpha_{1r}|^2 \text{ valószínûsséggel } \textbf{elfogadjuk} \text{ a bemenetet} \end{cases}$$

ahol l az idõ, k pedig a tár (circuit) mérete, azaz a felhasznált kapuk darabszáma.

4.3. Részleges mérés

Ez a mérési forma olyankor használatos, amikor nem az egész rendszer állapotára, hanem mondjuk csupán egy kitüntetett bit értékére vagyunk kíváncsiak. Még inkább olyankor, amikor tovább nem alkalmazunk a kitüntetett bitet érintő kvantum-műveletet. A mérést követõen ezt a bitet már nem tekintjük a rendszer (a "kvantumállapot"), részének. Az eljárás úgy értelmezhető, hogy itt csupán csak a mért bit, nem pedig az egész rendszer kerül ki a kvantumállapotból. A részleges mérés nem része a kurzuson definiált számítási modellnek, arra használjuk, hogy a jelölést egyszerűbbé tegyük. A méréskor felbontjuk a rendszert egy 1, illetve egy $k-1$ bites rendszer tenzorszorzatára.

$$\mathbb{C}^{2^k} = \mathbb{C}^2 \otimes \mathbb{C}^{2^{k-1}}$$

A ψ vektorunk ennek megfelelõen a következőképpen bontható fel:

$$\psi = |0\rangle \otimes \varphi_0 + |1\rangle \otimes \varphi_1 \text{ ,ahol}$$

$$\varphi_0 = \sum_{r=\{0,1\}^{k-1}} \alpha_{0r} \cdot |r\rangle \text{ ,és } \varphi_1 = \sum_{r=\{0,1\}^{k-1}} \alpha_{1r} \cdot |r\rangle$$

Ha csak az elsõ bit érdekelne bennünket, az $|\varphi_0|^2$ valószínûséggel 0, illetve $|\varphi_1|^2$ valószínûséggel 1 lenne. Ha a maradék rendszer sorsát is követni szeretnénk, a mérés során történteket úgy éretelmezzük, a rendszer a $|0\rangle \otimes \varphi_0$ állapotba kerül $|\varphi_0|^2$ valószínúséggel, vagy az $|1\rangle \otimes \varphi_1$ állapotba $|\varphi_1|^2$ valószínûséggel.

4.4. Klasszikus számítások kvantumszámítógépen

Ahhoz, hogy a kvantumszámítógépeken számolhassunk szükségünk van mûveletekre. A klasszikus mûveletek (NOT,OR,AND,XOR,...) azonban nem feltételnül használhatóak a

megszokott értelemben, hisz némelyikük nem ivertálható, azaz nem unitér mûvelet. Ilyen például a VAGY mûvelet.

$$x, y \to x, x \vee y$$
$$|x\rangle, |y\rangle \to |x\rangle, |x \vee y\rangle$$
$$|1\rangle, |0\rangle \to |1\rangle, |1\rangle$$
$$|1\rangle, |1\rangle \to |1\rangle, |1\rangle$$

Látható, hogy a mûvelet egyszerû kiterjesztése nem célravezetõ, mivel két különbözõ állapotnak ugyanaz lett a képe. A kvantummechanikához igazodó számítások a reverzibilis számítások, melyet Tommaso Toffoli, és Charles H. Bennett dolgozott ki a '70-es években. A reverzibilis mûveletek nagyban hasonlítanak a klasszikusakhoz, azonban mint nevükben is benne van, az eredménybõl minden esetben visszaállítható az eredeti állapot. Ezen mûveleteket felfoghatjuk úgy is, mint bitsorozat permutáló mûveleteket. Alapmûveletek:

- I: $|X\rangle \to |X\rangle$-identitás mûvelet, a 0-t 0-ba, az 1-et 1-be képezi
- NOT: $|X\rangle \to |X\rangle$-negáció, a 0-t 1-be, az 1-et 0-ba képezi
- XOR: $|X\rangle|Y\rangle \to |X\rangle|X \vee Y\rangle$-kizáró vagy mûvelet, szokták kontrollált negációnak is hívni(CNOT). Y-t negáljuk, ha X 1, máskülönben hagyjuk változatlanunl
- OR: $|X\rangle|Y\rangle|Z\rangle \to |X\rangle|Y\rangle|(X \text{ v } Y) \text{ xor } Z\rangle$-vagy mûvelet. Amennyiben $Z = 0$, akkor $X \vee Y.Z$ miatt a mûvelet invertálható.

1. Tétel. *Legyen $f = \{0,1\}^n \to \{0,1\}^k$ T hosszú Boole hálózattal számítható. Ekkor létezik $O(T)$ hosszú kvantumhálózat, ami*

$$|X\rangle|Y\rangle \to |X\rangle|Y \; xor \; f(x)\rangle.$$

Pontosabban a halózat által megvalósított leképezés a következőt tudja:

$$|X\rangle|0\rangle|0\rangle \to |X\rangle|f(x)\rangle|0\rangle,$$

ahol az új tag a bal oldalon a mumnkaterületet jelenti. A művelet végrehajtódik, majd visszaállítja a munkaterületet az eredeti állapotába. A munkaterület hossza $O(T)$, azaz a munkaterületet $|0...0\rangle$ módon lehet szemléletesebben ábrázolni. A munkaterület kitakarítása egy alapvetõen fontos feladat. Szerepe a külvilág elszigetelése, ami viszont nehéz feladat. Ha a munkaterületet nem takarítanánk ki, akkor annak qubitjei a külvilággal összefonódott állapotban maradhatnak, ami nagyon megnehezítené az elvégzett számítás elemzését. Itt jegyezzük meg, hogy a külvilággal való spontán összefonódás megakadályozás az egyik legnehezebb feladat a kvantumszámítógépek építésében.
A takarítás menete:

1. kiindulási állapot: a mûvelet Z-piszkot hagyott a munkaterületen $|X\rangle|0\rangle|0\rangle \xrightarrow{S} |X\rangle|f(x)\rangle|Z\rangle$

2. vegyünk fel még egy extra munkaterületet $|X\rangle|0\rangle|0\rangle|0\rangle \xrightarrow{S} |X\rangle|f(x)\rangle|Z\rangle|0\rangle$

3. XOR a II és a IV tag között $|X\rangle|f(x)\rangle|Z\rangle|0\rangle \xrightarrow{xor} |X\rangle|f(x)\rangle|Z\rangle|f(x)\rangle$

4. végezzük el S^{-1} az I,IV és III targra $|X\rangle|f(x)\rangle|Z\rangle|f(x)\rangle \xrightarrow{S^{-1}} |X\rangle|f(x)\rangle|0\rangle|0\rangle$

A mûvelet bináris kiterjesztésével kajuk meg a kvantumpárhuzamosság metatételét.

2. Tétel. *Ha f(x) T idõben számolható Boole függvénnyel, akkor létezik olyan O(T) idejû kvantum hálózat, ami* $\sum_{x\in\{0,1\}^n} \alpha_i|x\rangle|0\rangle|0\rangle \rightarrow \sum_{x\in\{0,1\}^n} \alpha_x|x\rangle|f(x)\rangle|0\rangle$

A tétel nem mást mond ki, mint hogy $f(x)$ kvantumhálózattal párhuzamosan számítható ki az összes lehetséges x-re. A (kitakarított) munkaterület szerepeltetésétől a továbbiakban eltekintünk, annak mérete a hálózat kapuinak számával arányosra (vagy még kisebbre) vehető.

4.4.1. BPP hálózat szimulációja BQP hálózattal

	BPP hálózat	BQP hálózat
input	x	x
munka	0	0(nagyobb)
véletlen bitek (r db)	Z	0

A kvantumhálózatnak a mûködéséhez nagyobb munkaterületre van szükség. Véletlen kvantumbiteket az alábbi módon tudok elõállítani, ami nem más mint Z-k szuperpozíciója.

$$H \otimes r \rightarrow \frac{1}{2^{r/2}} \sum_{Z\in\{0,1\}^2} |Z\rangle$$

4.5. Különbség a klasszikus és kvantumgépek között

Klasszikus gépek esetén véges kapukészlettel dolgozhatunk. Kvantumgépeknél 1-2-3 qubites unitér transzformációkat használunk, ezek végtelen sokan vannak. Ezen transzformációk azonban helyettesíthetõek bizonyos alkalmas véges készletekkel (ún. univerzális kapukészletekkel):

3. Tétel. *Léteznek olyan véges kapukészletek, amelyekre tetszõleges U 1,2,3 qubites unitér transzformáció ϵ-közelíthetõ $O(\frac{1}{\epsilon})$ kapu szorzatával a véges készletbõl.*

Itt ϵ-közelíthetõségen azt értjük, hogy U'-t alkalmazva bármilyen v vektorra, majdnem ugyanazt az eredményt kapom, mint ha U-t alkalmazuk volna v-re. A hiba nagysága ϵ. Formálisan:

$$|U'v - Uv| < \epsilon, \forall |v| = 1$$

Ha a számításban a T darab kaput egyenként ϵ-közelítõvel helyettesítjük, akkor a végeredményben legfeljebb $T \cdot \epsilon$ eltérés tapasztalható. Így δ összhibához az egyes lépésekben $\frac{\delta}{T}$-közelítés szükséges. Ez a kapuk számának $O(\frac{T}{\delta})$-szoros növelésével érhető el.

Tegyük fel, hogy a ψ helyett a ψ' állapotot kapjuk, ahol $|\psi - \psi'| < \delta$. Ekkor

$$\psi = |0\rangle \otimes \varphi_0 + |1\rangle \otimes \varphi_1$$
$$\psi'' = |0\rangle \otimes \varphi_0' + |1\rangle \otimes \varphi_1'$$

A ψ állapot esetén 0-t a $|\varphi_0|^2$, ψ' esetén pedig $|\varphi_0'|^2$ valószínûséggel kapjuk meg az első bit mérése eredményeként.

$$\delta^2 \geq |\psi - \psi'|^2 = ||0\rangle \otimes (\varphi_0 - \varphi_0') + |1\rangle \otimes (\varphi_1 - \varphi_1')|^2 = |\varphi_0 - \varphi_0'|^2 + |\varphi_1 - \varphi_1'|^2$$

(az utolsó egyenlőség azért igaz, mert a két vektor merõleges egymásra). Innen

$$|\varphi_0 - \varphi_0'| \leq \delta$$
$$|\varphi_1 - \varphi_1'| \leq \delta$$

és így

$$|\varphi_0'| \geq |\varphi_0| - \delta$$
$$|\varphi_1'| \geq |\varphi_1'| - \delta$$

Innen

$$|\varphi_0'|^2 \geq |\varphi_0|^2 - 2\delta$$
$$|\varphi_1'|^2 \geq |\varphi_1'|^2 - 2\delta$$

Tegyük fel, hogy ϕ esetén a helyes válasz valószínűsége legalább $C > \frac{1}{2}$. Ekkor a fentiak miatt ϕ' esetén a helyes válasz valószínűsége legalább $C - 2\delta$. Így, ha

$$\delta < \frac{C - \frac{1}{2}}{2}$$

, akkor C helyett legalább $C' = C - 2\delta > \frac{1}{2}$ valószínűséggel kapunk helyes választ.

4.5.1. Univerzális kapuk

Fentebb kimondtuk, hogy alkamlas véges kapukészletekkel ugyanaz a nyelv definiálható, mint végtelennel. Olyan kapukészletet, mellyel közelíteni tudunk 1,2,3 bites tetszõleges kapukat univerzális kapuknak nevezzük. Ilyen univerzális kapukészlet:

1. Elsõ kísérlet:
   - Az ún. Toffoli kapu egy speciális 3 bites kapu (CCNOT)
   - még néhány egy bites kaput

2. Második, szebb kapukészlet

   - CNOT $|X\rangle|Y\rangle \to |X\rangle|X \text{ xor } Y\rangle$
   - H, az Hadamard-transzformáció
   - $\begin{pmatrix} 1 & 0 \\ 0 & Z \end{pmatrix}$, ahol $|Z| = 1$, azaz Z egy 1 abszolút értékű komplex szám

5. fejezet

A Deutsch-Jozsa algoritmus

6. fejezet

Grover algoritmusa

A 2001.04.12-iki előadás alapján írta Dulai József
Lov Grover 1996 -ban publikálta ezt az algoritmust.

6.1. A keresési feledat

- Adva van egy N(kettő hatványa vagy kettő hatványára kerekített érték): N$\rightarrow 2^n$.
- Van egy f függvényünk$\{1, ..., N\} \rightarrow \{0, 1\}$.
- $\exists! x_0 \in \{1, ..., N\}$, ahol$f(x_0) = 1$ (egy ilyen elem van). A többi n-1 elem nulla, azaz$f(x) = 0$.
- Az f orákulummal adott: $|x> |y> \xrightarrow{U_{x_0}} |x> |y$ XOR $f(x)>$, x és y n qubites.

Ezek az előírt módon viselkednek.
A feladatunk, hogy keressük meg az $x_0 - t$**!**.

1. Klasszikusan: legrosszabb esetben $(N - 1)$ kérdés kell. Ez determinisztikus.
2. Randomizáltan: $O(N)$ a hiba valószínűségétől függően ez elégséges. Minimum $\Omega(N)$kell.

Heurisztikusan igazolva: $\frac{N}{100}$ esetén, mitől függően választhatjuk ki, valamint véletlen esetet alkalmazó algoritmust használva. Sorsoljuk az 1 értéket. Véletlen beleesik az $\frac{N}{100}$ -ba, így $\frac{1}{100}$ az esély, hogy megtaláljuk.

Kvantum esetén: Párhuzamosan tesszük fel a kérdéseket. Tehát a szerkezetünkbe behelyezünk egy kvantumállapotot és adott számú műveletet, majd egy mérést végzünk el rajta.

6.2. Összetevők

1. Tükrözések.
2. Forgatások.

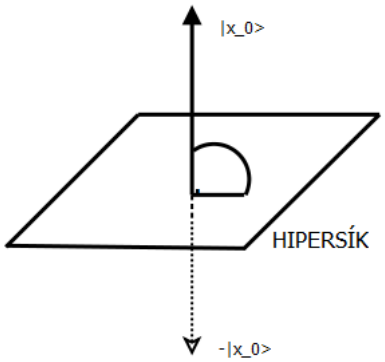
Tükrözés(1)

f- et számítsuk ki "fázisban", a múlt órán tanultak alapján
$|x> \xrightarrow{T_{x_0}} (-1)^{f(x)}|x>$. A T_{x_0} leképezés az $|x_0>$-re merőleges hipersíkra való tükrözés:

1. Ha $x \neq x_0$, akkor a T_{x_0} az x -nek megfelelő $|x>$ bázisvektorra a $T_{x_0}|x>=|x>$

2. Ha $x = x_0$, akkor $T_{x_0}|x>=-|x>$

Az x_0-ra merőleges bázisvektorokat önmagukba viszi, az x_0-ra párhuzamos vektorokat az ellentetjébe viszi.



6.1. ábra. Hipersíkra való tükrözés(1)

A $T_{|x_0>}$ transzformáció az $|x_0>^{\perp}$ hipersíkra való tükrözés.

Tükrözés(2):

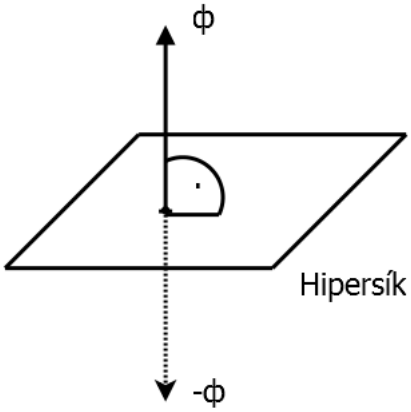
A

$$\Phi = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x>$$

uniform szuperpozícióra $\perp$ hipersíkra való T_Φ tükrözés,
A T_Φ implementációja $N = 2^n$ esetén:

- $H^{\otimes n}|0> \longleftrightarrow \Phi$.

  \+ $H^{\otimes n}|0>=\Phi$.



6.2. ábra. Hipersíkra való tükrözés(2)

+ $H^{\otimes n}\Phi = |0>$

A terünk, amiben dolgozunk:$N = 2^n$, azaz n darab 2 dimenziós térnek a tenzorszorzata. Az n darab Hadamard tarnszormáció tenzorszorzatát jelöljük $H^{\otimes n}$-nel. A csupa nulla bitet, az összes összes uniform szuperpozícióba fogja vinni. A n darab qubitre történő Hadaramd-transzformáció egymás utáni alkalmazásával kapható, bonyolultsága tehát: n.

- $T_{|0>} = |0>^{\perp}$-re való tükrözés. Fix transzformáció, békén hagyja $|x>$-et, ha $x \neq 0$, a nem nulla , egyébként tükrözi (előjelet kap, ha $x \neq 0$)

$$T_{|0>} = \begin{cases} |x> & \text{,ha x}\neq 0 \\ -|x> & \text{,ha x}=0 \end{cases}$$

Hatékonyan implementálható n qubites művelet. Két $H^{\otimes n}$-ből és a $T_{|0>}$-ból összerakjuk a $\Phi^{\perp}$-re való tükrözést:

1. Van egy vektorunk (v), ezt szeretnénk Φ-re merőleges hipersíkra tükrözni. A $H^{\otimes n}$ tarnszformáció a csupa nullából álló vektorba viszi Φ vektort és ezt $T_{|0>}$ -val tükrözzük, majd ezek után visszavisszük az eredeti állapotba. Azaz:
v$\xrightarrow{H^{\otimes n}}\xrightarrow{T_{|0>}}\xrightarrow{H^{\otimes n}}$

Ellenőrizhető, hogy

$$T_{\Phi}v = H^{\otimes n}T_{|0>}H^{\otimes n}v, \text{ ha } v = |0>, ..., |N-1>$$

. A linearitás miatt így

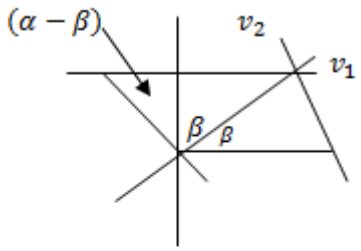
$$T_{\Phi}v = H^{\otimes n}T_{|0>}H^{\otimes n}$$

Tehát T_{Φ} hatékonyan megvalósítható O(n) művelettel
Van két tükrözésünk az orákulum által: T_{x_0} és T_{Φ}. Ezek hipersíkra való tükrözések. Mi van akkor, ha két tükrözést egymás után hajtjuk végre?

- T_{v_1} $v_1^{\perp} - re$ való tükrözés.
- T_{v_2} $v_2^{\perp} - re$ való tükrözés.

Mi történik 3d térben?

- Két sík metszete egyenes lesz. Csupa fix pontból áll. $v_1^{\perp}$ és $v_2^{\perp}$ fixen maradnak.



Olyan transzformáció, aminél a vektorok nullától vett távolsága nem változik. Ez egy forgatás lesz. A forgatás szöge 2α, ahol α a két tükrözés tengelye (ill. ezek normálvektora) átlal befogott szög,

Általában $v_1^{\perp}$ és $v_2^{\perp}$ metszete N-2 dimenziós altér fixen marad, a rá merőleges síkon a normálvektorok szögének 2szeresével való forgatás történik.

7. fejezet

Grover II. rész

A 2011.04.12-iki előadás alapján írta Fazekas Ádám

7.1. Grover algoritmusa

7.1.1. Egy „jó" elem keresésére

Kiindulunk a Φ uniform szuperpozícióból, és alkalmazzuk rá a két forgatás, $T_{|x_0\rangle}$ és T_Φ, szorzatát. Ezen mûvelet alkalmazsával a Φ és $|x_0\rangle$ által kifeszített síkon maradunk és azt reméljük, hogy többszöri alkalmazásával eljutunk az $|x_0\rangle$-ba.
Sajnos ez egy nem stabil forgatás:
Az $|x_0\rangle$ és Φ szögének koszinusza a két egységvektor skalárszorzata:

$$\cos\alpha = (|x_0\rangle, \Phi)$$

$$\Phi = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle,$$

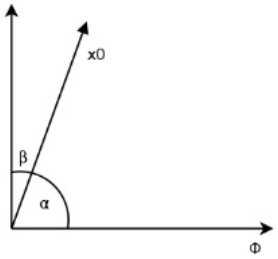
így

$$\cos\alpha = \frac{1}{\sqrt{N}}$$

Mivel N nagy, ezért α a derékszöghöz áll közel. A forgatási mûveletünk mértéke 2α, ezért az 180°-hoz áll közel.
A 360°-os forgatáshoz kétszer kellene elvégezni a mûveletet, de ehhelyett egy hatékonyabb megoldást alkalmazunk, a $T_{|x_0\rangle} \cdot T_\Phi$ forgatás helyett a $-T_\Phi \cdot T_{|x_0\rangle}$ transzformációt alkalmazuk. A Φ és $|x_0\rangle$ által kifeszített síkra megszorítva az egy $\pi - 2\alpha$ szöggel történő forgatás.

$$\beta = \frac{\pi}{2} \cdot \alpha$$

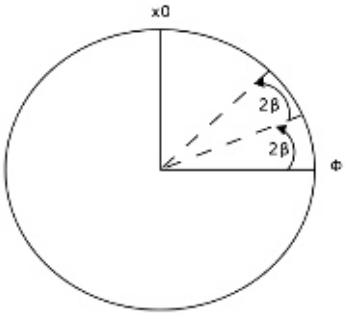
$$\sin\beta = \frac{1}{\sqrt{N}} \quad \Rightarrow \quad \beta \approx \frac{1}{\sqrt{N}}$$



7.1. ábra. Forgatás szöge

β jól közelíti $\frac{1}{\sqrt{N}}$-t.

1.) Forgatás: $\pi - 2\alpha$-val, azaz $\pi - (\pi - 2\beta)$ vagyis 2β szöggel.



7.2. ábra. Forgatás

Indulunk Φ-bõl és 2β forgatásokkal megyünk $|x_0\rangle$ felé.

$$\alpha \approx \frac{\pi}{2}, \text{ így } \frac{\alpha}{2\beta} \approx \frac{\frac{\pi}{2}}{\frac{2}{\sqrt{N}}} = \frac{\pi}{4}\sqrt{N}$$

Tehát $\frac{\pi}{4} \cdot \sqrt{N}$ lépésben nagyon közel jutunk $|x_0\rangle$-hoz, jelöljük ezt az állapotot ψ-vel. $\psi - |x_0\rangle$ kicsi.

$$\psi = \sum_{x=0}^{N-1} \alpha_x |x_0\rangle, \quad \text{ahol} \quad \alpha_x \in R$$

Ekkor $\alpha_{x_0} \approx 1$ és

$$\sum_{x \neq x_0} |\alpha_x|^2$$

kicsi, tehát $\frac{1}{2}$-nél bõven nagyobb valószínûséggel kapunk $|x_0\rangle$-t ha a ψ állapotot megmérjük. $O(\sqrt{N})$-nél kevesebb lépésre volt szükség, csak számolási mûveletek és egy orákulumhoz fordulás segítségével.

Az algoritmus teljesítménye sajnos nem javítható lényegesen, de jól általánosítható.

7.1.2. Általánosítások több elemre

Egy „jó” elem helyett M jó elemünk van:
Létezik pontosan M darab $x_1, \ldots, x_M$, hogy $f(x_1) = \cdots = f(x_M) = 1$, a többi x-re pedig $f(x) = 0$ úgy, hogy $M << N$.

1. Keressünk egy jó elemet
2. Közelítsük a jó elemek uniform szuperpozícióját

Mindkét változat megoldható, a bonyolultsága $O(\sqrt{\frac{N}{M}})$, ennyi kvantumkérdés szükséges. $N = 4$ esetében pont egy kérdésnyit forgatunk. 4 hatványaira pontosan kijön a jó elemek uniform szuperpozíciója, egyébként csak közelítőleg.

7.2. Alsó becslés a keresési feladatra

N elem közül egy jó, az x_0. Ennek a megkeresésére felhasználjuk az U_{x_0} orákulumot.

$$U_{x_0} |x\rangle |y\rangle = \begin{cases} |x\rangle |y\rangle & \text{ha } x \neq x_0 \\ |x\rangle |\text{NOT} y\rangle & \text{ha } x = x_0 \end{cases}$$

$$\Phi_{x_0} = V_{L+1} \cdot U_{x_0} \cdot V_L \cdot U_{x_0} \ldots V_2 \cdot U_{x_0} \cdot V_1 \cdot U_{x_0} \cdot |0\rangle$$

Valamilyen definit ($|0\rangle$) állapotból indulunk ki és minden számolási lépés (V_i) után van egy orákulumhívás (U_{x_0}.

$$\Phi_{x_0} = \sum_{x=0}^{N-1} a_x \otimes |x\rangle$$

Ezt megmérve, x valószínûsége az $|a_x|^2$(az a_x hosszának a négyzete).
Ha x_0 helyett egy x_0'-t alkalmazunk, akkor

$$\Phi_{x_0'} = \sum_{x=0}^{N-1} a_x' \otimes |x\rangle$$

kapunk, és U_{x_0} helyett $U_{x_0'}$-t használjuk. Ha helyesen akarjuk megmondani az x_0 és x_0'-t, akkor Φ_{x_0}' és Φ_{x_0}' szögének minnél jobban közelítenie kell a derékszöget, tehát a két vektor távolsága nem lehet túl kicsi.

Az Φ_{x_0} és $\Phi_{x_0'}$ távolságát úgy fogjuk megbecsülni, hogy mindkettőt összevetjük azzaz az állapottal, ami akkor jön létre, amikor az U_{x_0} illetve $U_{x_0'}$ orákulumot az "üres" orákulummal, az identikus transzformációval ($|x\rangle |y\rangle \longmapsto |x\rangle |y\rangle$) helyettesítjük.

8. fejezet

Grover III. rész

A 2011.04.12-iki el?adás alapján írta Vámos Dániel

8.1. Grover alsó becslés - folytatás

$$\Phi_{x_0} = U_{x_0}V_{L+1}U_{x_0}V_L \cdot \cdots \cdot U_{x_0}V_2U_{x_0}V_1|0>,$$
$$\Phi_{x_0'} = U_{x_0'}V_{L+1}U_{x_0'}V_L \cdot \cdots \cdot U_{x_0'}V_2U'_{x_0}V_1|0>,$$

Tegyük fel, hogy $L << \sqrt{N}$.

Az orákulumot identitásra cseréljük: $V_{L+1}\mathrm{I}V_L\mathrm{I}....\mathrm{I}V_2\mathrm{I}V_1|0>$. Ebben a sorozatban az i-edik lineáris elõtti állapotot jelöljük a következőképp:

$$\sum_{y=0}^{1}\sum_{x=0}^{N-1} a_i(x,y) \otimes |x> |y>,$$

ahol az "$a_?(??)$" az állapotnak az a része, ami nem az orákulum inputja.
Ezt átírva kapjuk:

$$\sum_{x=0}^{N-1} b_i(x) \otimes |x>, \text{ ahol } b_i(x) = a_i(x,0) \otimes |0> + a_i(x,1) \otimes |1>$$

Mivel $\sum_{x=0}^{N-1} |b_i x|^2$ minden egyes i-re, így

$$\sum_{i=1}^{L}\sum_{x=0}^{N-1} |b_i x|^2 = L$$

Ezért

$$\sum_{x=0}^{N-1}\sum_{i=1}^{L} |b_i(x)|^2 = L$$

Itt a belsõ összegek átlaga $\frac{L}{N}$
Ilyen feltétel mellett $\exists x_0 \neq x_0'$, amelyekre a megfelelő belső összzeg legfeljebb az átlag kétszerese:

$$\sum_{i=1}^{L} |b_i(x_0)|^2 \leq \frac{2L}{N} \ , \ \sum_{i=1}^{L} |b_i(x1_0)|^2 \leq \frac{2L}{N}$$

e két összeg esetén kapjuk meg.
Van egy korlátunk a hossznégyzet összegre és ebbõl megbecsülni a hossznégyzetet a következõ módon lehet:

$$\sum_{i=1}^{L} |b_i(x_0)|^2 = L \cdot \frac{1}{L} \sum_{i=1}^{L} |b_i(x_0)| \leq L \cdot \sqrt{\frac{\sum_{i=1}^{L} |b_i(x_0)|^2}{L}}$$

számtani|négyzetes közepe $\leq \sqrt{L} \cdot \frac{\sqrt{2L}}{\sqrt{N}} = \frac{\sqrt{2} \cdot L}{\sqrt{N}}$

Tehát:

- $x_o - ra$ és $x_0' - ra$ igazak:

$$\sum_{i=1}^{L} |b_i(x_0)| \leq \frac{\sqrt{2}L}{\sqrt{N}} \text{és} \sum_{i=1}^{L} |b_i(x_0')| \leq \frac{\sqrt{2}L}{\sqrt{N}}$$

Hamis futás

Kiindultunk a nulla állapotból.
Aztán V_1,orákulumhívás($\mathbf{I}$), $V_2, ..., V_L \mathbf{I} V_{L+1}$. $U_{L+1} \mathbf{I} V_L, ..., \underbrace{\mathbf{I} V_2 \underbrace{\mathbf{I} V_1 |0>}_{A_1}}_{A_2}$

Itt $A_1, A_2, \ldots$ állapotokat jelölnek és az A_{i+1n} úgy írhatjuk fel, hogy $A_{i+1} = \mathbf{I} V_{i+1} A_i$ és az$A_0 = |0>$ rekurziónk van.

Normális futás

Ha a rejtett elem x_0, akkor az identitás:

$V_{L+1} U_{x_0} V_L .... \underbrace{U_{x_0} V_2 \underbrace{U_{x_0} V_1 |0>}_{B_1}}_{B_2}$.

Megpróbáljuk a B_i és A_i különbségét megbecsülni:
$|B_i - A_i|$=?
$|B_0 - A_0| = 0$

$|B_{i+1} - A_{i+1}| = |U_{x_0} V_{i+1} B_i - \mathbf{I} V_{i+1} A_i| = |U_{x_0} V_{i+1}(B_i - A_i) + U_{x_0} V_{i+1} A_i - V_{i+1} A_i| = |B_{i+1} - A_{i+1}| \leq |(U_{x_0} - \mathbf{I}) V_i A_{i-1}|$,
ahol $V_i \cdot A_{i-1} = \sum_x b_i(x) \otimes |x>$. Erre alkalmazhatjuk mind az identitást, mind az U_{x_0}-t: $(U_{x_0} - \mathbf{I}) V_i A_{i-1}$

- Ha $x \neq x_0$, akkor azU_{x_0} és $\mathbf{I}$ egyenlõ. Ekkor $(U_{x_0} b_i(x_0) \otimes |x_0> - b_i(x_0 \otimes |x_0>) = b_i(x_0)(a_i(x_{0,0}) \otimes |0> + a_i(x_{0,1}) \otimes |1>$

$$U_{x_0} b_i(x_0) \otimes |x_0> = (a_i(x_{010}) \otimes |1> + (x_{011})|0>) \otimes |0> .$$

Ezért

$$|U_{x_0} b_i(x_0) \otimes |x_0> - b_i(x_0) \otimes |x_0>|^2 =$$
$$|a_i(x_{0,0}) \otimes |1> + a_i(x_{0,1}) \otimes |0> - a_i(x_{0,0}) \otimes |0> - a_i(x_{0,1}) \otimes |1>|^2 = 2|a_i(x_{0,0}) - a_i(x_{0,1})|^2$$

Az egész hossza:$\sqrt{2}|a_i(x_{0,0}) - a_i(x_{0,1})|$ és ez becsülhetõ a $b_i(x_0)$ hossz segítségével: Becsüljünk a hosszak összegével: $\sqrt{2}(|a_i(x_{0,0})| + |a_i(x_{0,1})|) \leq \sqrt{2}(|a_i(x_{0,0})| + |a_i(x_{0,1})|) \leq 2\sqrt{2}|b_i(x_0)|$

Indukcióval kapjuk, hogy

$$|B_L - A_L| \leq C \sum_{i=1}^{L} |b_i(x_0)|$$

alkalmas $C > 0$ konstanssal. Az x_0 választása miatt ekkor

$$|B_L - A_L| \leq C \frac{L}{\sqrt{N}}.$$

Végeredmények

1. x_0-ás eset:$V_{L+1} B_L$
   - hamis eset: $V_{L+1} A_L$
   - különbség: $|V_{L+1}(B_L - A_L)| \leq C\frac{L}{\sqrt{N}}$, ahol V_{L+1} unitér (hossztartó).
2. x_0 helyett x_0' esetében:
   - különbsége $\leq C\frac{L}{\sqrt{N}}$
3. x_0-ás és az x_0'- ás eset különbsége(távolsága):túl kicsi, azaz sokkal kisebb, mint $2C\frac{L}{\sqrt{N}}$,mert az L sokkal kisebb, mint a $\sqrt{N}(L << \sqrt{N})$

<u>Konklúzió:</u>

* Ez ellentmondás (amiatt, hogy $L << \sqrt{N}$).

Megjegyzések:

* Az egyes lépésekben elkövetett hibák összeadódnak.

* $O(\sqrt{N})$ méretű adatbázissal nem lehet helyettesíteni a lekérdezéseket.

* Nem elég fix szuperpozícióra feltenni a kérdést.

* A nyereség a polinomiálisnál nem jobb.

9. fejezet

Simon algoritmusa

A 2011. 04. 19.-ei előadás alapján készítette Tóthfalusi Tamás

9.1. A feladat

Adott (orákulummal) egy olyan $f\{0,1\}^n \to \{0,1\}^n$ függvény, amely azt tudja, hogy $\exists$ u ϵ $\{0,1\}^n \setminus \{0...0\}$, hogy f(x) = f(y) $\Leftrightarrow$ x = y vagy x = y XOR u.

A kvantum orákulum: U_f : $|x\rangle|0\rangle \to |x\rangle|f(x)\rangle$.

A feladat, hogy keressük meg az u-t!

9.2. Gyors megoldás kvantumgépen

Kiindulunk a következő két regiszteres kezdőállapotból. Alkalmazzuk az első registerre az Hadamard transzformációt. Ezzel a 0,1 sorozatoknak az uniform szuperpozícióját nyerjük.

$|0\rangle|0\rangle \longrightarrow^{H^{\otimes n}} \frac{1}{\sqrt{2^n}} \sum_{x\epsilon\{0,1\}^n} |x\rangle\ |0\rangle \longrightarrow^{U_f}$

Ezután meghívjuk az U_f uniform transzformációt, azaz párhuzamosan lekérdezzük az összes függvény értékét:

$\longrightarrow^{U_f} \frac{1}{\sqrt{2^n}} \sum_{x\epsilon\{0,1\}^n} |x\rangle\ |f(x)\rangle \longrightarrow$

Következő lépésben csoportosítjuk f értékei szerint az adatokat. Azokat gyűjtjük össze, amikor az f(x) értéke egy előre meghatározott y.

$\frac{1}{\sqrt{2^{n-1}}} \sum_y (\frac{1}{\sqrt{2}} \sum_{x\epsilon\{0,1\}} |x\rangle)\ |y\rangle =$

$\frac{1}{\sqrt{2^{n-1}}} \sum_y$ ($\frac{1}{\sqrt{2}}(|x_y\rangle$ + $|x_y$ XOR u$\rangle$)),

ahol x_y = min{x | f(x) = y}.

Rögzített y-ra:

$\frac{1}{\sqrt{2}}(|x_y\rangle$ + | x_y XOR u$\rangle) \in \mathbb{C}^{2^n}$

Ezen vektor tulajdonsága, hogy u-val XOR-ra invariáns.
Jelölés : (XOR_u)
Ezen tulajdonság segítségével próbálunk az u-ra valamilyen információt nyerni.

Tény:

$\mathbb{C}^{2^n}$ térnek van olyan ortonormált bázisa, amelynek elemei az összes XOR_u típusú transzformáció közös sajátvektorai. Tehát minden egyes báziselem sajátvektor.

Ha ebben a bázisban felírjuk ezt az invariáns állapotot:
$\frac{1}{\sqrt{2}}(|x_y\rangle + |x_y\ XOR_u\rangle)$, akkor csak olyan bázisvektoroknak lesz nem nulla együtthatója, amelyeknél az XOR_u megfelelő sajátértéke 1. Ezen vektorok adják az információt az u-ról.

XOR_u a következőt tudja: $|x\rangle \longrightarrow |xXORu\rangle$

XOR_u^2 : |x$\rangle \longrightarrow$ |x$\rangle$ a báziselemeket fixen tartja, vagyis ez az identikus lineáris transzformáció.

Legyen M egy mátrix, f(M) = 0 egy f polinomra.
λ sajátértéke az M-nek => f(λ) = 0.

$\exists$ v vektor, hogy Mv = λv, Iv = 1v, $M^2v = \lambda^2 v$
Bizonyítás:
M^2v = M(Mv) = M(λv) = λ(Mv) = $\lambda\lambda v = \lambda^2 v$.
...
$M^k v = \lambda^k v \Longrightarrow$ f(M)v = f(λ)v
azaz f(M) = 0, $f(M)v = 0$.
Mivel f(M) = 0 $\Longrightarrow$ 0v = f(λ)v $\Longrightarrow$ f(λ)=0.
XOR_u^2 - I = 0
$\lambda^2 - 1 = 0$, minden λ sajátértékre.
$\lambda = \pm 1$.
A 0-val való XOR-olásnáll minden sajátérték 1 lesz, egyébként a sajátértékek fele +1, fele -1 lesz, a vektorok párban vannak. $\Longrightarrow$ A teret 2 dimenziós alterek összegére lehet felbontani. A felbontásból olyan sajátvektorokat kapunk, amelyek meghatározzák az u-t.
□

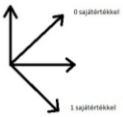
n = 1 eset

standard bázis: $|0\rangle$ $|1\rangle$
2 XOR-oló műveletünk van:
XOR_0 = I,
$XOR_1 = |0\rangle \Leftrightarrow |1\rangle$.

Ha felrajzoljuk:

Az XOR_1 transzformáció a következő két vektort cseréli meg:

Az XOR_1 a szögfelezőre való tükrözés. XOR_1 sajátvektorai valamilyen olyan vektor, amely a tükröző tengelyre esik és 0 vagy 1 sajátértékkel rendelkezik.



0 sajátérték: $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$
1 sajátérték: $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$

1 bites esetben:
A közös sajátbázisba való transzformáció az Hadamard transzformáció. (Vagyis ha fel szeretnénk írni egy, a standard bázisban adott vektort a sajátvektorokkal adott rendszerben, akkor az Hadamard transzformációt kell alkalmazni a koordinátákra.)

n-bites esetben:
$H^{\otimes n}$-et használjuk, mely n biten bitenkénti Hadamard transzformációk egymásutániságát jelenti.

$H^{\otimes n}$ számolása képlettel:

$H^{\otimes n}$ $|\mathrm{x}\rangle$ = ?
$H^{\otimes n}$ $|\mathrm{x}\rangle = \sum_{y=\{0,1\}^n} \alpha_{x,y}$ $|\mathrm{y}\rangle$.

n = 1 esetben:

$\mathrm{H}|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$

$H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$
$\alpha_{x,y} = \frac{1}{\sqrt{2}} \cdot 1$, ha x = 0, vagy y = 0
$\alpha_{x,y} = \frac{1}{\sqrt{2}} \cdot (-1)$, ha x = y = 1

Másképpen:
$\alpha_{x,y} = \frac{1}{\sqrt{2}} \cdot (-1)^{x \cdot y}$

n = 2 esetben:

x = (x1, x2), y = (y1, y2).
$|x\rangle = |x1\rangle \otimes |x2\rangle$
$|y\rangle = |y1\rangle \otimes |y2\rangle$

Jelöljük $(H \otimes H)_{x,y}$-nal a $H \otimes H$ transzformácó mátrixának az "x-edik" sorának "y-adik" elemét. Ekkor: $(H \otimes H)_{x,y} = H_{x1,y1} \cdot H_{x2,y2} = \frac{1}{2}(-1)^{x1 \cdot y1} \cdot (-1)^{x2 \cdot y2} =$
$\frac{1}{2}(-1)^{x1y1 \oplus x2y2}$, ahol $\oplus$ = XOR.

Általában a következő képpen írható fel az Hadamard transzformáció:

$H_{x,y}^{\otimes n} = \frac{1}{\sqrt{2^n}}(-1)^{\sum_{i=1}^{n} x_i y_i mod 2}$
Legyen (x,y) = $\sum_{i=1}^{n} x_i y_i$ mod 2

Másképpen fogalmazva:

$$H^{\otimes n}|x\rangle = \frac{1}{\sqrt{2^n}} \sum_{z \epsilon \{0,1\}^n} (-1)^{(x,y)}|z\rangle$$

Folytatjuk tovább a számítást:

$$\frac{1}{\sqrt{2^n}} \sum_y \frac{1}{\sqrt{2}}(|x_y\rangle + |x_y \oplus u\rangle)|y\rangle \longrightarrow^{H^{\otimes n}}$$

Az első regiszteren dolgozunk.
$\longrightarrow \frac{1}{\sqrt{2^{n-1}}} \sum_y (\frac{1}{\sqrt{2^{n+1}}} \sum_{z \epsilon \{0,1\}^n} ((-1)^{(x_y,z)} + (-1)^{(x_y \oplus u,z)})|z\rangle)|y\rangle$
Kiszámoljuk, hogy mely együtthatók lesznek 0-ák illetve nem 0-ák:

$(-1)^{(x_y,z)} + (-1)^{(x_y \oplus u,z)} = 0$ ax $(u, z) \equiv 1$ mod 2 esetben, különben $\frac{\pm 1}{\sqrt{2^{n-1}}}$, így az összeg:

$\frac{1}{\sqrt{2^{n-1}}} \sum_y (\frac{\pm 1}{\sqrt{2^{n-1}}} \sum_{z \in u^\perp} |z\rangle)|y\rangle$,

ahol $u^\perp = \{z \epsilon \{0,1\}^n | \sum u_i z_i = 0 \text{ mod } 2\}$.

Mérés:

Megmérjük az eredményt. Azt látjuk, hogy olyan (z,y) pár, amelyre teljesül, hogy (z,y)=0. Ezek a párok egyforma valószínűséggel jönnek elő.
Első tagként az olyan z-k, melyekre (z,u) = 0 (mod 2). Ezek egyenletes valószínűséggel jönnek ki.

Az eredmény tehát az $u^{\perp}$ egy véletlen eleme.

Ezt ismételev O(nlog n)-szer, olyan elemeket kapunk $u^{\perp}$-ből, amelyek együtt nagy valószínűséggel generálják az $u^{\perp}$-t (generált altér).

Generált altér: az a legszűkebb altér, amely minden vektorunkat tartalmazza.

Összes lehetséges összeget képezve kapjuk meg a generált altér vektorait. Ezen vektorokra merőleges vektorok csak a 0 és az u.

Összefoglalva

$|0\rangle|0\rangle \longrightarrow^{H^{\otimes n}} \longrightarrow^{Uf} \longrightarrow^{H^{\otimes n}} \longrightarrow^{Mrs}$ } O(n log n) -szer ismételve.

Egy ilyen sorozatban 1 bit információt nyerünk az ismeretlen u-ról,
O(n log n) lépésben pedig az összes bitjéről információt nyerünk.
Ez n-ben polinomiális művelet, amellyel nagy valószínűséggel meg tudjuk határozni a titkos u vektort.
Az y képletekne keresztül történő hurcolása helyett alkalmazhattunk volna *részleges mérést* is, ugyanazt az eredményt kaptuk volna:

$|0\rangle|0\rangle \longrightarrow^{H^{\otimes n}} \longrightarrow^{Uf} \longrightarrow$Mérés$\longrightarrow^{H^{\otimes n}} \longrightarrow$Mérés

10. fejezet

A Yao-elv

A 2011 04.19.-ei és 26.-ai előadás alapján készítette Labancz Anita

A Yao-elv

A legjobb randomizált algoritmus költsége a számára a legrosszabb inputon egyenlő vagy ugyanaz, mint a legrosszabb véletlen eloszlású inputon vett legjobb determinisztikus algoritmus költsége.

Értlmező megjegyzések

1. Az algoritmus korekt (jó), ha legalább $1-\varepsilon$ valszínűséggel adjon korrekt (jó) választ(ez a randomizáltra vonatkozik).

2. Az algoritmus az eloszlás szerint súlyozva, az input legalább $\geq 1-\varepsilon$ részén adjon korrekt választ(ez a determinisztikusra vonatkozik).

10.1. Játék

Két résztvevős zérusösszegű játék, ahol adva van egy A mátrixunk, $A = (a_{ij})$, ami $K \times L$-es méretű. Van egy S sorjátékosunk és egy O oszlopjátékosunk. Ha S az i-edik sort választja és az O a j-edik oszlopot, akkor az S megnyeri az a_{ij} mennyiségű aranyat, az O pedig elveszti ezt a mennyiségű aranyat(ettől lesz zérusösszegű a játék).

Példa. A fociban a 11-es rúgás. S lesz a rúgó, O pedig a kapus.

Az A mátrixunk a következő lesz:

$$\mathbf{A} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

A sor illetve oszlopindex: bal-jobb

$$\text{Nincs gól} \quad = \begin{cases} a_{\text{bal,bal}} & S \text{ nyeresége } 0 \\ a_{\text{jobb,jobb}} & O \text{ vesztesége } 0 \end{cases}$$

$$\text{Gól} \quad = \quad \begin{cases} a_{\text{bal,jobb}} & S \text{ nyeresége } 1 \\ a_{\text{jobb,bal}} & S \text{ nyeresége } -1 \end{cases}$$

Kevert stratégia:

Az S játékos startégiája olyan $\underline{x} = \mathbb{R}^K$, ahol $x_i \geq 0$ (x_i valószínűséggel választja az i-dik sort) és $\sum x_i = 1$. Az O játékosé pedig $\underline{y} \in \mathbb{R}^L$ ($y_j \geq 0$, $\sum y_j = 1$.)

S nyeresége $\sum a_{ij} x_i y_j$. S abban érdekelt, hogy ez a mennyiség nagy legyen. O vesztesége $\sum a_{ij} x_i y_j$. O abban érdekelt, hogy ez a mennyiség kicsi maradjon.

S rögzített x stratégiája esetén O a legjobb stratégiáját akkor játssza, ha a következő minimumot keresi:

$$\min_{\underline{y}} \sum_{i=1}^{K} \sum_{j=1}^{n} a_{ij} x_i y_j = min_{\underline{y}} \sum_{j=1}^{L} y_j \left(\sum_{i=1}^{K} a_{ij} x_i \right) = min_j \sum_{i=1}^{L} a_{ij} x_i$$

S elérhető maximális nyeresége

$$max_{\underline{x}} min_{\underline{y}} \sum_{i,j} a_{ij} x_i y_j = \left(= max_{\underline{x}} min_j \sum_i a_{ij} x_i \right)$$

O elérhető minimum vesztesége

$$min_{\underline{y}} max_{\underline{x}} \sum_{i,j} a_{ij} x_i y_j = \left(= min_{\underline{y}} max_{\underline{j}} \sum_j a_{ij} y_j \right)$$

Példa. A tizenegyesrúgásoknál mindkét fél akkor jár a lgjobban, ha $1/2 - 1/2$ valószínűséggel választ a "bal-jobb" közül.

10.2. A minimax-tétel

(Neumann János 1928)

$$max_{\underline{y}} min_{\underline{x}} \sum_{i,j} a_{ij} x_i y_j = max_{\underline{x}} min_{\underline{y}} \sum_{i,j} a_{ij} x_i y_j$$

A belső max esetén ismert y minden egyes koordinátáját x-ben kell maximalizálni.

Tiszta belső stratégiára átfogalmazva

x_i alapján kell a számok súlyozott közepét venni.

$$\min_{\underline{y}} max_{1 \leq i \leq K} \sum a_{ij} y_j$$

$$\max_{\underline{x}} min_{1 \leq j \leq L} \sum a_{ij} x_i$$

10.3. A Yao-elv bizonyítása

Alkalmazzuk a minimax-tételt a következő szituációban: Legyen $0 < \epsilon < 1$. Olyan algoritmusokat szeretnénk optimalizálni, amelyek legfeljebb ϵ hibával működnek. A algoritmusok költsége legyen c, amelyek determinisztikusak. Két játékos játszik, L lehetséges inputtal. Az ilyen algoritmusok közül $\underline{x}$ súlyozás szerint választ egyet véletlenül. O játékos n lehetséges bemenetek közül $\underline{y}$ súlyozás szerint választ egyet.

$$a_{ij} = \begin{cases} 1, & \text{ha i-edik algoritmus j-edik inputon helyes} \\ 0, & \text{egyébként} \end{cases}$$

Erre alkalmazva a Minimax-tételt:

$$P_{\underline{y}}(\text{legjobb } i\text{-edik algoritmus helyes értéket ad}) = min_{\underline{y}} max_{1 \leq i \leq K} \sum a_{ij} y_j$$

$$P_{\underline{y}}(\text{a legrosszabb algoritmus értékét adja}) = max_{\underline{x}} min_{1 \leq j \leq L} \sum a_{ij} x_i$$

Miért jó nekünk ez? Könnyebb kezelni azokat az algoritmusokat, amik determinisztikusak és csak az inputjuk véletlen.

11. fejezet

A Simon-probléma klasszikus bonyolultsága

A 2011 04.26-iki előadás alapján írta Csernusné Ádámkó Éva

Simon probléma:

Adott egy f függvény, $f : \{0,1\}^n \to \{0,1\}^n$, úgy, hogy $\exists!$ u $= \{0,1\}^n$, hogy
$f(x) = f(y) \Leftrightarrow \begin{cases} x = y & \text{vagy} \\ x = (x & \text{XOR } y). \end{cases}$

Igazolni akarjuk a következõt:

A legjobb randomizált algoritmus költsége $\Omega(2^{\frac{n}{2}})$, vagyis annak költsége, hogy hány helyen kérdezzük meg $f(x)$-et.

Megkérdezzük $f(x)$-et N helyen - N db x-re - például ott, ahol $N << 2^{n-1}$, és ha nem találunk ütközést, vagyis $x_1, \ldots, x_N$-re $f(x_i) \neq f(x_j)$ ha $i \neq j$, akkor nagyon sok lehetséges tippünk marad az u-ra. Az összes olyan u szóba jöhet, ami nem $(x_i \; XOR \; x_j)$ alakú. A játék arra megy ki, hogy ne találjunk ütközést a lekérdezettek között. Legyen $N < 2^{\frac{n}{2}}$, vagyis $\frac{N}{2} < 2^n$ így biztosan sok lesz a lekérdezettek között amelynél nem lesz ütközés.

Determinisztikus algoritmus.

- Tfh. legfeljebb N kérdést teszünk fel, ahol $N < \frac{1}{4} \cdot 2^{\frac{n}{2}}$

"Nagyon gonosz" bemenet.

- egyenletes valószínűséggel véletlenül választunk egy u vektort, úgy hogy $u \epsilon \{0,1\}^n \setminus (0, \ldots, 0)$, és ezután egy egyen;etes valószínűséggel egy f függvényt a következõképpen $(x, x \; XOR \; y) \to f(x) \epsilon \{(0,1)\}^n$
(különbözõhöz különbözõt rendel)

- az egyenletes eloszlás "elég gonosz" lesz
- azt állítjuk, hogy ha ilyen kevés kérdést tesz fel, akkor az algoritmus jó eséllyel hibázik,
- Világos, hogy ha nincs ütközés, akkor legalább $\frac{1}{2}$ valószínûséggel fog hibázni
- tegyük fel , hogy az algoritmus elsõ k lépésében a kérdések az $x_1, x_2, \ldots, x_k$ helyekre vonatkoznak
- ha eddig volt ütközés, (vagyis valamely $i \neq j$-re $f(x_i) = f(x_j)$), akkor az algoritmus nyert és válaszként az egyetlen lehetõség az $u = (x_i \; XOR \; x_j)$
- különben azt mondjuk, hogy $x_1, x_2, \ldots, x_k$ egy ütközésmentes sorozat
- következik a $k+1$-edik lépés:
  az algoritmus választ egy x_{k+1} helyet, ahol megkérdezi a függvényértéket, ez függ az $f(x_1), f(x_2), \ldots, f(x_k)$ értékektõl (ez indirekt függ a véletlentõl is)
  próbáljuk meg megbecsülni azt a valószínûséget, hogy ez a sorozat ütközésmentes, feltéve, hogy az elsõ k ütközésmentes:

$$\begin{aligned} Prob(x_1, x_2, \ldots, x_{k+1} \text{ ütközésmentes} \quad &| \quad x_1, x_2, \ldots, x_k \text{ ütközésmentes }) \\ &= \quad 1 - \frac{k}{2^n - 1 - \binom{k}{2}} \\ &\geq \quad 1 - \frac{k}{2^{n-1}}, \text{ ugyanis} \end{aligned}$$

$$k < N < \frac{1}{4} 2^{\frac{n}{2}} \Rightarrow \binom{k}{2} << \frac{1}{8} \cdot 2^n \Rightarrow \binom{k}{2} + 1 < \frac{1}{2} \cdot 2^n$$

$$\begin{aligned} Prob \quad & (x_1, x_2, \ldots, x_N \text{ ütközésmentes }) \\ = \quad & \prod_{k=1}^{N-1} Prob(x_1, x_2, \ldots, x_{k+1} \text{ ütközésmentes } \mid x_1, x_2, \ldots, x_k \text{ ütközésmentes }) \\ \geq \quad & \prod_{k=1}^{N-1} (1 - \frac{k}{2^{n-1}}) \\ \geq \quad & 1 - \sum_{k=1}^{N-1} \frac{k}{2^{n-1}} \\ = \quad & 1 - \frac{N \cdot (N-1)}{2 \cdot 2^{n-1}} \\ \geq \quad & 1 - \frac{N^2}{2^n} \end{aligned}$$

- tehát, ha $N < \delta \cdot 2^{\frac{n}{2}}$ akkor $N^2 < \delta^2 \cdot 2^n$, ami azt jelenti, hogy $1 - \delta^2$ valószínûséggel ütközésmentes az $(x_1, x_2, \ldots, x_N)$ sorozat, és marad $2^n - 1 - \binom{N}{2} > 2$ lehetõség az u-ra, ami azt jelenti, hogy $\frac{1}{2}$-nél sokkal nagyobb a hiba valószínûsége.

12. fejezet

Faktorizáció visszavezetése perióduskeresésre

A 2011. április 26-i előadás alapján készítette Almási Gábor

Törzstényezős felbontás: könnyen visszavezethető a valódi osztó keresésére. Az egésznek az alapja olyan $(x,\ y)$ párnak a keresése, amelyre teljesül, hogy

$$x^2 \equiv y^2 \ mod\ m,$$

ahol az m számot szeretnénk bontani úgy, hogy

$$x \not\equiv \pm y \ mod\ m.$$

A legtöbb ismert faktorizáló algoritmus (

$$x\ (y \equiv 1),\ x^2 \equiv 1, \text{ de } x \not\equiv\ \pm 1.$$

Ebben az esetben $lnko(m,\ x-1)$ valódi osztója m-nek. Miért igaz ez? Ugyanis

$$x^2 - 1 = (x+1)(x-1) \tag{12.1}$$

osztható m-mel, de annak egyik tényezője sem osztható m-mel, ezért

$$lnko((x-1),\ m) \neq m \tag{12.2}$$

és

$$lnko((x-1),\ m) \neq 1 \tag{12.3}$$

különben az $(x+1)$ osztható lenne m-mel.

A legnagyobb közös osztót az Euklideszi algoritmussal számíthatjuk ki: $O \log_2 m$, tehát jól járunk, ha találunk megfelelő x-et, erre irányul a kvantum-algoritmus (vagy véletlenül mellékesetként talál egy osztót).

Legyen $a \in Z$ és tegyük fel, hogy az a relatív prím az m-hez, azaz $lnko(a,\ m) = 1$. Ekkor létezik egy olyan $r > 0$ egész, hogy az $a^r \equiv 1\ mod\ m$ a legkisebb ilyen r, ezt úgy hívjuk, hogy az a multiplikatív rendje $mod\ m$.
Jelölés: $o_m(a)$.
Az összes ilyen r az $o_m(a)$ többszöröse, sőt

$$x \longmapsto f(x) = a^x mod\ m,$$

$$f(x) = f(y) \Leftrightarrow x - y \text{ osztható } o_m(a)\text{-val}$$

Az x-ből hogyan számolható hatékonyan az f(x)?
Ez $(\lg(x)+lg(n))^{O(1)}$ időben számítható. Az x bitjeinek és n bitjeinek polinomjában számolható gyorshatványozással: $1,\ a,\ a^2\ mod\ m,\ a^4\ mod\ m, \ldots, a^{2k}\ mod\ m$ (mindig redukálunk $mod\ m$-mel), majd az alkalmasakat összeszorozzuk. Melyek az alkalmasak?
Azok, amelyekre teljesül, hogy x megfelelő bináris számjegye 1.

Tehát van egy függvényünk, amelynek periódusa nem más, mint a-nak a multiplikatív rendje, azaz $o_m(a)$.

Definíció: legyen $f : Z \to \{\text{l hosszú 0-1 sorozatok}\}$. Ekkor az f periodikus egy P periódus szerint, ha $f(x) = f(y) \Leftrightarrow P|(x - y)$ $(P > 0$ egész).
Tegyük fel, hogy az $f(x)$ hatékonyan $(l+\lg(x))^{O(1)}$ polinomjában számolható. Ekkor – mint később látni fogjuk – létezik $(l + \lg x)^{O(1)}$ idejű kvantum-algoritmus a P periódus kiszámítására. Tehát a periódus polinom időben számolható, ezt alkalmazzuk a faktorizációnál.

Alkalmazás: $f_a(x) = a^x\ mod\ m$ (látjuk, hogy hatékonyan számolható). $o_m(a)$ (periódus) $\lg(m)^{O(1)}$-ben számolható kvantum-algoritmussal. Ha ez igaz, akkor hogyan tudunk ebből faktorizálni?

Legyen $m \in Z$, $m > 0$. Ekkor

- feltehető, hogy m páratlan
- feltehető, hogy m nem teljes hatvány
- különben az $m = x^2$ vagy $m = x^3$ vagy ... vagy $m = x^k$ $(k < \log_2(m))$

  Szervezhetünk egy ciklust: $k = 2, \ldots, \lg(m)$ $(m = x^k)$. Ha ez igaz, akkor hogyan tudjuk meghatározni x-et?
  Ha $m = x^k$, akkor x megkereshető $O\lg(m)$ lépésben bináris kereséssel.

Állítás: ha m olyan, hogy $m = p_1^{\alpha_1} \cdot \ldots \cdot p_s^{\alpha_s}$, ahol p_i páratlan prímszám, $s > 1$, akkor véve egy egyenletesen véletlen $0 < a < m$ számra $\geq \frac{1}{2}$ valószínűséggel, vagy $lnko(a,\ m) \neq \pm 1$, vagy $o_m(a)$ páros és $a^{\frac{o_m(a)}{2}} \not\equiv \pm 1\ mod\ m$.

Egy ilyen a-ra

- 1. eset: $lnko(a,\ m)$ valódi osztója m-nek
- 2. eset: $x = a^{\frac{o_m(a)}{2}}$-vek:
  $x^2 \equiv 1$ és $x \not\equiv \pm 1\ mod\ m$,
  tehát $lnko((x-1),\ m)$ valódi osztója m-nek; az x gyorshatványozással hatékonyan számolható!

Az eljárás tehát a következő lépésekből áll:

1. veszünk egy véletlen a-t
2. - ha nem relatív prím, akkor nyertünk
   - különben kiszámoljuk a rendjét
     - ha páratlan $\Rightarrow$ passz
     - ha $a^{\frac{o_m(a)}{2}} \equiv \pm 1 \Rightarrow$ passz
     - különben nyertünk

$O(\lg \frac{1}{\epsilon})$ ismétléssel a nyerés valószínűsége $1 - \epsilon$ fölé tehető.

Az állítást igazoljuk (véletlen a-ra, a 2 esemény bekövetkeztének valószínűsége legalább $\frac{1}{2}$): elég azt igazolni, hogy olyan a-kra, amelyekre $lnko(a,\ m) = 1$, az esetek legalább felében fennáll az $o_m(a)$-s feltétel. Ehhez segítségül hívjuk a kínai maradéktétel egy specializált és erősebb változatát, amely a következőt mondja ki:
legyenek $a_1, \ldots, a_s$ olyanok, hogy $0 \leq a_i < p_i^{\alpha_i}$, és p_i nem osztja a_i-t. Ekkor létezik egyértelműen olyan $0 \leq a < m$, hogy az $a \equiv a_i\ mod\ p_i^{\alpha_i}$ és $lnko(a,\ m) = 1$.
Azaz amit felhasználunk:
véletlen $a\ mod\ m \leftrightarrow$ véletlen $(a_1, \ldots, a_m)(mod\ p_1^{\alpha_1}, \ldots, p_s^{\alpha_s})$.

Tétel: ha p páratlan prím, akkor létezik olyan b, hogy $o_{p^\alpha}(b) = p^\alpha - p^{\alpha-1}$.

Következmény: egy ilyen b-vel a véletlen p^α-hoz relatív prím számok nem mások, mint $b^z\ mod\ p^\alpha$, ahol z egy véletlen $0 \leq z < p^\alpha - p^{\alpha-1}$ (b primitív gyök).

Legyenek p_i-re b_i-k (azaz a primitív gyökök) ilyenek;
ekkor véletlen $a \leftrightarrow$ véletlen $(z_1, \ldots, z_s)$-re $(b_1^{z_1}, \ldots, b_s^{z_s})$-t kapjuk, ahol
$0 \leq z_i < p_i^{\alpha_i} - p_i^{\alpha_i - 1}$. Így az alábbi összefüggést kapjuk:

$$a^x \longleftrightarrow (b_1^{z_1 \cdot x}, \ldots, b_s^{z_s \cdot x}).$$

Ebből az a multiplikatív rendje a következőképpen írható fel:

$$o_m(a) = lkkt(o_{{p_j}^{\alpha_j}}(b_j^{z_j})), \text{ ahol } j = 1, \ldots, s.$$

Legyen $p_i^{\alpha_i} - p_i^{\alpha_i - 1} = 2^{\beta_i} d_i$, ahol d_i páratlan ($i = 1, \ldots, s$). Két esetet különböztetünk meg:

1. eset: a β_i-k nem mind ugyanazok.
Feltehető, hogy β_1 a legnagyobb, és β_2 a legkisebb. Az esetek felében a z_1 páratlan, ilyenkor:

$$o_{p_1{}^{\alpha_1}}(b_1^{z_1}) = 2^{\beta_1} \cdot (\text{páratlan})$$

és

$$o_m(a) = 2^{\beta_1} \cdot (\text{páratlan}).$$

Következésképpen ez egy páros szám lesz és a

$$b_1^{z_1 \frac{o_m(a)}{2}} = b_2^{2^{\beta_1 - 1} \cdot (\text{páratlan})} \equiv -1 \; mod \; p_1^{\alpha_1}$$

illetve a

$$b_2^{\frac{o_m(a)}{2}} = b_2^{2^{\beta_1 - 1} \cdot (\text{páratlan})} \equiv 1 \; mod \; p_2^{\alpha_2}$$

kongruenciákból az alábbi összegzéshez jutunk:

$$a^{\frac{o_m(a)}{2}} \equiv \begin{cases} -1 \; mod \; p_1^{\alpha_1} \\ +1 \; mod \; p_2^{\alpha_2} \end{cases}$$

$\Rightarrow$ Nem lehet $a \equiv \pm 1 \; mod \; m$.
2. eset: minden β_i ugyanaz, $\beta_i > 0$.
Az esetek felében z_1 és z_2 közül pontosan az egyik páros! Ezekben az esetekben az $o_m(a)$ páros és az

$$a^{\frac{o_m(a)}{2}} \equiv \begin{cases} +1 \; mod \; p_1^{\alpha_1} \\ -1 \; mod \; p_2^{\alpha_2} \end{cases}$$

vagy éppen fordítva. $\Rightarrow$ Azaz szintén nem lehet, hogy $a \equiv \pm 1 \; mod \; m$.
Ezzel az állítás helyességét beláttuk.

13. fejezet

Periódus keresése kvantumszámítógéppel

A 2011. április 26-iki előadás alapján készítette Zsigmond Attila

13.1. „Nulladik megközelítés”

Legyen $f : \mathbb{Z} \to \{l \text{ bites sorozatok}\}$ periodikus függvény P periódussal, adott $|x\rangle|0\rangle \overset{U_f}{\longmapsto} |x\rangle|f(x)\rangle$ kvantumorákulummal (unitér művelet).

„Inkorrekt” feltevés: ismert a P-nek egy N többszöröse. A továbbiakban mod$N$ dolgozunk. Elkészítjük a számok uniform szuperpozícióját (alkalmazzuk az U_f orákulumot):

$$\frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle|0\rangle \overset{U_f}{\longmapsto} \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle|f(x)\rangle,$$

majd megmérjük a második regiszter tartalmát (részleges mérés $f(x)$-re). A mérés eredménye valamely $f(x_0)$ érték:

$$\frac{c}{\sqrt{N}} \sum_{x:f(x)=f(x_0)} |x\rangle = \frac{1}{\sqrt{N/P}} \sum_{x \equiv x_0 \bmod P} |x\rangle = \Psi.$$

Legyen $T_z : |x\rangle \longmapsto |x+z\rangle \bmod N$ (z-vel való ciklikus shiftelés). Ekkor $T_P\Psi = \Psi$, azaz Ψ sajátvektora T_P-nek 1 sajátértékkel. Ésszerű az összes szóba jövő z-re mint periódusra a T_z-k sajátvektoraiból való bázisra áttérni.

Az igazán értékes művelet az 1-gyel való shiftelés mod$N$, azaz a T_1. Mátrixa:

$$T_1 = \begin{pmatrix} & & & & 1 \\ 1 & & & & \\ & \ddots & & & \\ & & \ddots & & \\ & & & 1 & \end{pmatrix}$$

$$T_1^2 = \begin{pmatrix} & & 1 & \\ & & & 1 \\ 1 & & & \\ & \ddots & & \\ & & 1 & \end{pmatrix}$$

Látható, hogy $T_z = T_1^z$.

T_1 minimálpolinomja: $x^N - 1$ (mert $T^N = I$). A karakterisztikus polinom ugyanez.

Legyen $\omega = e^{\frac{2\pi i}{N}}$. Ekkor T_1 sajátértékei: ω^j, ahol $j = 0, 1, \ldots, N-1$.

T_1 sajátvektorai:

$$\frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} \omega^{-ij} |i\rangle = \Phi_j$$

$$T_1 \Phi_j = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} \omega^{-ij} |i+1\rangle \bmod N$$

A futó indexet átnevezzük:

$$i + 1 = i'$$

$$i = i' - 1$$

Ekkor:

$$T_1 \Phi_j = \frac{1}{\sqrt{N}} \sum_{i'=0}^{N-1} \omega^{-(i'-1)j} |i'\rangle = \omega^j \frac{1}{\sqrt{N}} \sum_{i'=0}^{N-1} \omega^{-i'j} |i'\rangle = \omega^j \Phi_j$$

QFT_N: Quantum Fourier Transform (Kvantum Fourier-transzformáció)

A transzformáció mátrixának az oszlopai T_1 sajátvektorai, pontosabban:

$$QFT_N |i\rangle = \frac{1}{\sqrt{N}} \sum_j \omega^{ij} |j\rangle$$

Megjegyzés: $QFT_2 = H$ (Hadamard-transzformáció).

Alkalmazzuk Ψ-re QFT_N-t:

$$\frac{1}{\sqrt{N/P}} \sum_{x \equiv x_0} |x\rangle$$

Később tisztázzuk, hogy indokolt-e ezt a lépést megtenni (van-e hatékony implementációja a kvantum Fourier-transzformációnak)?

$$QFT_N \Psi = \frac{1}{\sqrt{N/P}} \frac{1}{\sqrt{N}} \sum_{x \equiv x_0} \sum_y \omega^{xy} |y\rangle = \frac{1}{\sqrt{N/P}} \frac{1}{\sqrt{N}} \sum_y \left(\sum_{x \equiv x_0} \omega^{xy} \right) |y\rangle$$

$$\sum_{x \equiv x_0} \omega^{xy} = \sum_{z=0}^{N/P-1} \omega^{Pz + x_0 y} = \omega^{x_0 y} \sum_{z=0}^{N/P-1} (\omega^{Py})^z$$

Ez pedig egy mértani sor, melynek összegképletét alkalmazva:

$$\begin{cases} \frac{\omega^{PyN}-1}{\omega^{Py}-1} = 0, & \text{ha } \omega^{Py} \neq 1 \\ N/P, & \text{ha } \omega^{Py} = 1, \text{ azaz } Py \equiv 0 \bmod N \end{cases}$$

Nem nehéz belátni, hogy néhány olyan véletlen y, amelyre $Py \equiv 0 \bmod N$, nagy valószínűséggel meghatározza P-t, mégpedig hatékony módon.

14. fejezet

A kvantum Fourier-transzformáció

A május 3-iki előadás alapján írta Varga Péter.

A modulo N vett kvantum Fourier-transzformáció (QFT) a következő Φ_N leképezés:

$$\Phi_N : |i> \mapsto \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \omega^{ij} |j> ,$$

ahol $\omega = \omega_N = e^{2\pi \frac{i}{N}}, (i := \sqrt{-1})$ az N. primitív egységgyök.

14.1. QFT modulo 2-hatvány

Legyen most $N = 2^l$. Vezessük ezt vissza a 2^{l-1} esetre: alkalmazzuk a a $j = 2j'$, illetve a $j = 2j' + 1$ helyettesítést és azt, hogy j bináris ábrázolása ($j = \sum s = 0^{l-1} j_s 2^s$) alapján $|j >= |j_0 > |j' >$ alakú.

$$\Phi_{2^l}|i >= \frac{1}{2^{\frac{l}{2}}} \sum_{j=0}^{2^l-1} \omega_{2^l}^{ij} |j >= \frac{1}{2^{\frac{l}{2}}} \left(\sum_{j'=0}^{2^{l-1}-1} \omega_{2^l}^{2ij'} |0 > |j' > + \sum_{j'=0}^{2^{l-1}} \omega_{2^l}^{2ij'+1} |1 > |j' > \right) =$$

$$= \frac{1}{2^{\frac{l}{2}}} |0 > \otimes \sum_{j'=0}^{2^{l-1}-1} \omega_{2^l}^{2ij'} |j' > + \omega^i |1 > \otimes \sum_{j'=0}^{2^{l-1}-1} \omega_{2^l}^{ij'} |j' >=$$

$$= \frac{1}{\sqrt{2}} (|0 > + \omega^i |1 >) \otimes \frac{1}{2^{\frac{l-1}{2}}} \sum_{j'=0}^{2^{l-1}-1} \omega_{2^{l-1}}^{ij'} |j' >= \frac{1}{\sqrt{2}} (|0 > + \omega^i |1 >) \otimes \Phi_{2^{l-1}} |i \ mod \ 2^{l-1} >$$

Alkalmazzuk a Hadamard transzformációt, majd sorban i bitjeire ($s = 0, 1, \ldots l-1$) végrehajtjuk a következőt:
ha i-nek az s. bitje 1, és a kimenet 1. bitje is 1, akkor az együtthatót szorozzuk be $\omega_{2^l}^{2^s}$-sel.

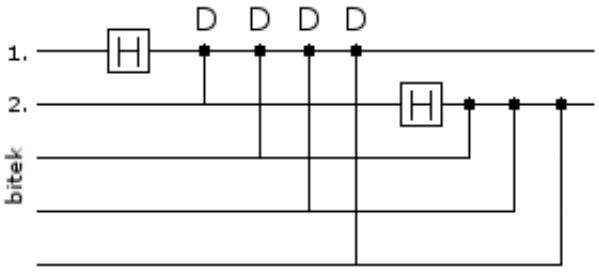
QFT fromális programmal

Célszerű először olyan változatot bemutatni, aminél a bemeneti bitek $|i>$ bitjei magas $\rightarrow$ alacsony helyiértékűek, a kimeneti bitek fordítva.
for $t = 0$ to $l - 1$
H a t. bitre
for $s = t + 1$ to $l - 1$
ha az s. és a t. bit mindkettő 1, akkor szorzás $\omega_{2^{s-t+1}}$-gyel.

Az output fordított sorrendben keletkezik, a végén a biteket meg lehet cserélni, ha zavaró.

QFT hálózattal

14.1. ábra. D: duplán feltételes fázisváltoztatás



14.2. Uniform szuperpozíció létrehozása

Olyan transzformációt szeretnénk konstruálni, ami ezt a szuperpozíciót hozza létre.

$$|0> \mapsto \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} |i>$$

Φ_N ezt $N = 2^l$-re megcsinálja, ezt szeretnénk általánosítani.

1. ötlet - Grover-szerű lépésekkel

$$\frac{1}{\sqrt{2^l}} \sum_{i=0}^{2^l} |i> \text{, ahol } l = \lceil logN \rceil.$$

Grover-lépésekkel csináljuk, N-től függően polilogaritmikus számú lépés kell csak: $(logN)^{O(1)}$.

2. ötlet - Mérés/eldobás

$$\frac{1}{\sqrt{2^l}}\sum_{i=0}^{2^l-1}|i>|0>\mapsto\frac{1}{\sqrt{2^l}}\sum_{i=0}^{2^l-1}|i>|i\geq N?>$$

Utolsó bit mérése. $\geq\frac{1}{2}$ valószínűséggel kapunk egy tökéletes uniform szuperpozíciót.
Hátránya: Ezt a műveletet egymás után csak konstans sokszor lehet alklamazni, különben túl nagy lesz a hiba valószínűsége.

3. ötlet - 2. ötlet javítása

$$l>>logN$$

$$N_1=2^l$$

$$N_2=N\cdot M,\text{ ahol }2^l\geq NM>2^l-N$$

csináljuk meg az uniform szuperpozíciót(N_1 kettő hatvány):

$$\frac{1}{\sqrt{N_1}}\sum_{i=0}^{N_1-1}|i>\approx\frac{1}{\sqrt{N_2}}\sum_{i=0}^{N_2-1}|i>\mapsto^*\frac{1}{\sqrt{M}}\frac{1}{\sqrt{N}}\sum_{j=0}^{M-1}\sum_{i=0}^{N-1}|i>|j>=$$

Az approximáció azért áll fenn, mert ha N_1,N_2 nagy számok, akkor $\frac{1}{N}_1,\frac{1}{N}_2$ számok közt kicsi a különbség.
* - $|i>$-t kétfelé bontjuk: $|i\%N>$, és $|i/N>$.

$$=\left(\frac{1}{\sqrt{N}}\sum_{i=0}^{N-1}|i>\right)\otimes\left(\frac{1}{\sqrt{M}}\sum_{j=0}^{M-1}|j>\right)$$

Vagy részleges mérést alkalmazunk, vagy visszük tovább a második tagot. Ez utóbbi sem okoz gondot, mert ez nem fonódik össze azzal az állapottal, amit mi szeretnénk.

Az első tag az uniform szuperpozíció, amit kerestünk.

4. ötlet

Hadamard transzformációval előállítjuk az első bitben a 0 és 1 egyforma szuperpozícióját, majd alkalmas $\frac{1}{\sqrt{2}}(|0>+|1>)$ átsúlyozással lecsökkentjük az $|1>$ bit súlyát.
$|0>$ esetén $\Phi_{2^{l-1}}$ megoldja az uniform szuperpozíciót.
$|1>$ esetén N helyett $N-2^{l-1}$-re rekurzió.

15. fejezet

Sajátértékbecslés (fázisbecslés)

A 2011. május 3-iki előadás alapján készítette Pleva Péter

Legyen adott orákulummal egy $\mathcal{U}$ unitér transzformáció, pontosabban unitér transzformációk egy olyan sorozata, melynek tagjai a rögzített $\mathcal{U}$-nak kettőhatványadik hatványai, azaz

$$\mathcal{U},\mathcal{U}^2,\mathcal{U}^4,\dots$$

illetve egy φ kvantumállapot mint $\mathcal{U}$ egy sajátvektora.

A feladat az, hogy becsüljük meg a megfelelő, φ sajátvektorhoz tartozó sajátértéket. Mivel $\mathcal{U}$ unitér, a sajátérték egy 1 abszolút értékű komplex szám, tehát felírható

$$e^{2\pi\iota\cdot\alpha}$$

alakban ($0 \leq \alpha < 1, \iota = \sqrt{-1}$). Ebben az esetben α fázisként értelmezhető, így a feladat a továbbiakban α közelítése.

Ezt többféleképpen megtehetjük $\mathcal{U},\mathcal{U}^2,\mathcal{U}^4,\dots$ transzformációkat használó kvantumáramkörök segítségével:

1. Így például a
$$\varphi \otimes |\,0\,\rangle \longmapsto \varphi \otimes |\,\alpha \text{ első } l \text{ bitje}\,\rangle$$
transzformációval adott l bit pontosságig közelíthetjük α-t. (A becslés bitekben mért l pontossága ez esetben megegyezik a kimenet hosszával.)

2. Egy másik megközelítésben rögzített nevezőjű törtekkel is approximálhatjuk a keresett fázist. Adott N nevező mellett a
$$\varphi \otimes |\,0\,\rangle \longmapsto \varphi \otimes |\,\alpha\text{-hoz legközelebbi } N \text{ nevezőjű tört számlálója}\,\rangle$$
transzformációval írhatjuk le a problémát. Figyeljük meg, hogy ez az eset könnyen visszavezethető a bitenkénti megközelítésre, így hasonlóan jó approximációval szolgál.

15.1. Kvantumos Fourier-transzformáció

A fentiek megoldása előtt azonban tekintsük a sajátértékbecslés egy alkalmazását, az általános, N dimenziós Fourier-transzformációt (QFT mod N).

$$\Phi_N : |\, i \,\rangle \longmapsto \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \omega^{ij} \,|\, j \,\rangle$$

A QFT megvalósításához – hasonlóan más kvantumprogramhoz – segédregisztereket használhatunk, melyeket 0-val inicializálva, majd a kvantumprogram végén visszaállítva ezt a kezdeti értéket ideiglenes tárat hozhatunk létre. Egyetlen regiszter bevezetését a következőképpen jelölhetjük:

$$|\, i \,\rangle\, |\, 0 \,\rangle \longmapsto \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \omega^{ij} \,|\, j \,\rangle\, |\, 0 \,\rangle\,.$$

Elsőként állítsuk elő az uniform szuperpozíciót ($\mathcal{US}$) ideiglenes tárunk felhasználásával:

$$|\, i \,\rangle\, |\, 0 \,\rangle \overset{\mathcal{US}}{\longmapsto} |\, i \,\rangle \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} |\, j \,\rangle = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} |\, i \,\rangle\, |\, j \,\rangle\,.$$

A következő lépésben egyszerűen cseréljük meg a regiszterek tartalmát ($\mathcal{SR}$):

$$\frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} |\, i \,\rangle\, |\, j \,\rangle \overset{\mathcal{SR}}{\longmapsto} \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} |\, j \,\rangle\, |\, i \,\rangle\,.$$

Szintén megtehetjük, hogy a szumma argumentumát egy-egy ij-től függő együtthatóval bővítjük ($\mathcal{CO}$) – így például ω^{ij}-vel, hiszen ez az érték a regisztertartalmakból származtatható. Ehhez előbb kiszámítjuk az ij mod N-t egy újabb regiszterben, majd az eredmény bitjei szerint feltételes fáziseltolásokat alkalmazunk. Végezetül a segédregiszter tartalmának eltüntetéséhez a szorzás algoritmusát fordított irányban hajtjuk végre. A $\mathcal{CO}$ transzformáció által tehát a következő állapothoz jutunk:

$$\frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} |\, j \,\rangle\, |\, i \,\rangle \overset{\mathcal{CO}}{\longmapsto} \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \omega^{ij} \,|\, j \,\rangle\, |\, i \,\rangle\,.$$

Vegyük észre, hogy az elérni kívánt

$$\varphi = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \omega^{ij} \,|\, j \,\rangle$$

állapot a $\mathcal{T}_{-1}$ ciklikus shift-transzformáció sajátvektora

$$\omega^i = e^{2\pi\iota \cdot \frac{i}{N}}$$

sajátértékkel. Így a sajátértékbecslésnél használt jelöléseket alkalmazva az

$$\alpha = \frac{i}{N}$$

hányados nem más, mint φ fázisa. Következésképpen a $\mathcal{CO}$ transzformációval előállt állapotban szereplő i-t tekinthetjük az α-hoz legközelebbi N nevezőjű tört számlálójának. Tehát a fázisbecslés ($\mathcal{PE}$) algoritmusának ismeretében:

$$\varphi \otimes |\, 0 \,\rangle \overset{\mathcal{PE}}{\longmapsto} \varphi \otimes |\, i \,\rangle \,.$$

Tekintve, hogy a $\mathcal{CO}$ végrehajtásával kapott állapotban mi éppen ennek a fordítottját szertnénk elérni – vagyis az i eltűnését –, így a kvantumalgoritmusoknál szokásos módon a $\mathcal{PE}$ transzformáció inverzét kell alkalmaznunk:

$$\frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \omega^{ij} \,|\, j \,\rangle \,|\, i \,\rangle \overset{\mathcal{PE}^{-1}}{\longmapsto} \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \omega^{ij} \,|\, j \,\rangle \,|\, 0 \,\rangle \,.$$

Összefoglalva elmondhatjuk, hogy az $\mathcal{US}$, $\mathcal{SR}$, $\mathcal{CO}$, ill. $\mathcal{PE}^{-1}$ transzformációsorozattal elő tudtuk állítani i kvantumos Fourier-transzformáltját:

$$\Phi_N : |\, i \,\rangle \overset{\mathcal{US}}{\longmapsto} \ldots \overset{\mathcal{SR}}{\longmapsto} \ldots \overset{\mathcal{CO}}{\longmapsto} \ldots \overset{\mathcal{PE}^{-1}}{\longmapsto} \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \omega^{ij} \,|\, j \,\rangle \,.$$

Az implementáció két hibaforrása a két közelítés, hiszen a sajátértékbecslés, ill. az uniform szuperpozíció létrehozása is többnyire közelítő eredményt ad.

15.2. Sajátértékbecslés implementálása

Legyen adott egy φ kvantumállapot, ill. egy megfelelő, l bit méretű segédregiszter az output és az ideiglenes tár számára. A következőkben a φ-hez tartozó sajátértéket szeretnénk közelíteni kvantumáramkörök segítségével. Elsőként hozzuk létre az uniform szuperpozíciót a segédregiszterben:

$$\varphi \otimes |\, 0 \,\rangle \overset{\mathcal{US}}{\longmapsto} \varphi \otimes \frac{1}{\sqrt{2^l}} \sum_{i=0}^{2^l-1} |\, i \,\rangle = \frac{1}{\sqrt{2^l}} \sum_{i=0}^{2^l-1} \varphi \otimes |\, i \,\rangle \,.$$

Ezután alkalmazzuk φ-re $\mathcal{U}$ hatványait a következőképpen: ha $|\, i \,\rangle$ s-edik bitje 1, akkor hajtsuk végre az $\mathcal{U}^{2^s}$-hez tartozó orákulumot ($s = 0, \ldots, l-1$). Például $i = 5$ (binárisan 101) esetén az $|\, i \,\rangle$ vektor nulladik, ill. második komponense 1, így φ-re $\mathcal{U}^{2^0}$, ill. $\mathcal{U}^{2^2}$ orákulumát egyaránt alkalmaznunk kell. Ez viszont egyenértékű az $\mathcal{U}^{2^0+2^2} = \mathcal{U}^5$-höz tartozó orákulum

egyszeri végrehajtásával. Figyeljük meg, hogy mindez általánosságban is elmondható, azaz bármely $|\, i \,\rangle$ az $\mathcal{U}^i$ transzformáció alkalmazását vonja maga után. Tehát:

$$\frac{1}{\sqrt{2^l}} \sum_{i=0}^{2^l-1} \varphi \otimes |\, i \,\rangle \overset{\mathcal{U}^{2^s}}{\longmapsto} \frac{1}{\sqrt{2^l}} \sum_{i=0}^{2^l-1} \mathcal{U}^i \, \varphi \otimes |\, i \,\rangle \,.$$

Tekintve, hogy egy transzformáció hatványaival vett szorzásnak a sajátértékhatványokkal való szorzás felel meg, másrészt φ nem csak $\mathcal{U}$-nak, hanem tetszőleges $\mathcal{U}^i$-nek is sajátvektora, így:

$$\frac{1}{\sqrt{2^l}} \sum_{i=0}^{2^l-1} \mathcal{U}^i \, \varphi \otimes |\, i \,\rangle = \frac{1}{\sqrt{2^l}} \sum_{i=0}^{2^l-1} e^{2\pi\iota\alpha\cdot i} \, \varphi \otimes |\, i \,\rangle \,.$$

Tegyük fel, hogy α felírható valamely $0 \leq j < 2^l$ egész, ill. 2^l hányadosaként, vagyis

$$\alpha = \frac{j}{2^l}$$

alakban (tehát j éppen az n bites kvantumtér egyik kitüntetett bázisvektora). Ekkor az α behelyettesítésével kapott

$$\frac{1}{\sqrt{2^l}} \sum_{i=0}^{2^l-1} e^{\frac{2\pi\iota}{2^l}\cdot j\cdot i} \, \varphi \otimes |\, i \,\rangle$$

állapotban $e^{\frac{2\pi\iota}{2^l}}$ nem más, mint a 2^l-edik primitív egységgyök, azaz ω_{2^l}. Vegyük észre, hogy újbóli behelyettesítéssel, ill. átrendezéssel éppen j (2^l dimenziós) Fourier-transzformáltjához jutunk:

$$\varphi \otimes \frac{1}{\sqrt{2^l}} \sum_{i=0}^{2^l-1} \omega_{2^l}^{j\cdot i} \, |\, i \,\rangle \,.$$

Így a keresett α (pontosabban j) meghatározásához nincs más teendőnk, mint Φ_{2^l} inverzét alkalmazni:

$$\varphi \otimes \frac{1}{\sqrt{2^l}} \sum_{i=0}^{2^l-1} \omega_{2^l}^{j\cdot i} \, |\, i \,\rangle \overset{\Phi^{-1}}{\longmapsto} \varphi \otimes |\, j \,\rangle \,.$$

Összességében a következő transzformációsorozat útján jutottunk el φ fázisának $\frac{j}{2^l}$-es közelítéséhez - feltéve, hogy α felírható 2^l-es tört alakban:

$$\varphi \otimes |\, 0 \,\rangle \overset{\mathcal{US}}{\longmapsto} \varphi \otimes \frac{1}{\sqrt{2^l}} \sum_{i=0}^{2^l-1} |\, i \,\rangle \overset{\mathcal{U}^{2^s}}{\longmapsto} \varphi \otimes \frac{1}{\sqrt{2^l}} \sum_{i=0}^{2^l-1} e^{2\pi\iota\alpha\cdot i} \, |\, i \,\rangle \overset{\Phi^{-1}}{\longmapsto} \varphi \otimes |\, j \,\rangle \,.$$

Figyeljük meg, hogy a tulajdonképpeni számítás a Fourier-bázisban ment végbe, ami tipikusnak mondható kvantumos környezetben.

Most vizsgáljuk meg azt az esetet, amikor α nem tesz eleget a feltételünknek, vagyis nem írható le egyetlen 2^l nevezőjű törttel. Ekkor a megfelelő számlálót az l bites kvantumtér (kitüntetett) bázisvektorainak komplex együtthatós lineáris kombinációjaként adhatjuk meg. Tehát a keresett fázis szükségképpen felírható

$$\frac{1}{2^l}\sum_{j=0}^{2^l-1} c_j j$$

alakban, ahol $\sum_{j=0}^{2^l-1} |\, c_j \,|^2 = 1$ ($c_j \in \mathbb{C}$). Ezúttal ezt a "szuperpozíciót" írhatjuk be az $\mathcal{U}^{2^0},\ldots,\mathcal{U}^{2^{l-1}}$ transzformációsorozat elvégzésével előállt formulába:

$$\frac{1}{\sqrt{2^l}}\sum_{i=0}^{2^l-1} e^{2\pi\iota\alpha\cdot i}\,\varphi\otimes|\, i\,\rangle = \frac{1}{\sqrt{2^l}}\sum_{i=0}^{2^l-1} e^{\frac{2\pi\iota}{2^l}\sum_{j=0}^{2^l-1} c_j j\cdot i}\,\varphi\otimes|\, i\,\rangle\,.$$

Innen - a korábbiakhoz hasonlóan - az $\omega_{2^l} = e^{\frac{2\pi\iota}{2^l}}$ összefüggés alkalmazásával, majd az inverz Fourier-transzformáció végrehajtásával kapjuk az eredményt szolgáltató

$$\varphi\otimes\sum_{j=0}^{2^l-1} c_j\,|\, j\,\rangle$$

állapotot. Ebből végül méréssel nyerhetjük ki a fázist közelítő értéket.

A következőkben ennek a becslésnek a pontosságát szeretnénk meghatározni – l függvényében. Legyen $0 < \mu < \frac{1}{4}$ a közelítés hibakorlátja, $\epsilon > 0$ tetszőlegesen kicsi valós szám, valamint

$$\frac{1}{2^l} = O\,(\mu\epsilon)\,.$$

Most próbáljunk meg felső becslést adni a hiba valószínűségére, tehát azon c_j együtthatók hossznégyzetösszegére, melyekre $\frac{j}{2^l}$ és α eltérése az adott μ hibahatáron kívül esik, azaz adjunk felső korlátot a

$$\sum_{j\notin J} |\, c_j \,|^2$$

összegre, ahol $J = \left\{j \,:\, \left|\frac{j}{2^l} - \alpha\right| \le \mu\right\}$. Megmutatható, hogy mérés után ilyen hibás eredményt kevesebb mint

$$c_l \cdot \frac{1}{\mu 2^l}$$

valószínűséggel kaphatunk, ahol $c_l \in \mathbb{R}$ egy alkalmas konstans. Felhasználva az l-re vonatkozó megszorítást tovább pontosíthatjuk a becslésünket:

$$\sum_{j\notin J} |\, c_j \,|^2 < c_l \cdot \frac{1}{\mu}\,O\,(\mu\epsilon) = O\,(\epsilon)\,.$$

Vagyis a hibás mérések összvalószínűsége praktikusan elhanyagolható, így a mérés ϵ közelítéssel a kívánt eredményt szolgáltatja:

$$\varphi \otimes \sum_{j=0}^{2^l-1} c_j \,|\, j \,\rangle \overset{\epsilon}{\approx} \varphi \otimes \sum_{j \in J} c_j \,|\, j \,\rangle \,.$$

Mindez azt jelenti, hogy bármilyen fázis tetszőleges pontossággal közelíthető a fenti kvantumalgoritmus segítségével.

16. fejezet

Perióduskeresés - részletes megoldás

A 2011 május 3-iki előadás alapján készítette Glavosits Tamás

Tegyük fel, hogy adott egy $f : \mathbb{Z} \to \{0,1\}^\ell$ függvény, amely periódikus valamilyen P periódussal. Az f függvény orákulummal adott. Ebben a fejezetben ismertetünk egy algoritmust az f függvény ismeretlen P periódusának meghatározására. Ez az algoritmus alkalmazható egy szám faktorizálása során, amikor egy a redukált maradék multiplikatív rendjét kell meghatároznunk.

16.1. Lánctörtek

Mivel a bemutatásra kerülõ algoritmusban használni fogjuk a valós számok lánctörtekkel történõ approximációját, így elõbb röviden bemutatjuk ezt az approximációs módszert.

Elsõ lépésben megmutatjuk, hogy hogy bármely racionális szám euklídeszi algoritmus segítségével véges lánctörtbe fejthetõ. Az euklídeszi algoritmus a maradékos osztás fogalmára épül. Ismert számelméleti tétel alapján bármely a és b pozitív egész számokhoz egyértelmûen léteznek olyan q és r nemnegatív egészek, melyekkel

$$a = bq + r, \qquad \text{továbbá} \qquad 0 \leqslant r < b.$$

Az a-t ostandónak, a b-t osztónak, a q-t hányadosnak, az r-et maradéknak neezzük. A q hányados és az r maradék például az alábbi módon határozható meg:

$$q = \max\{z \in \mathbb{Z} | a - bz \geqslant 0\} \qquad \text{és} \qquad r = a - bq.$$

Az (a, b) pozitív egészekbõl álló elempáron végrehajtott euklídeszi algoritmus maradékos osztásoknak egy olyan sorozata, melynek elsõ tagjában az osztandó az a és az osztó a b, majd az ezt követõ tagokban az osztandó szerepét átveszi a megelõzõ maradéos osztásban kapott osztó, és az osztó szerepét átveszi a megelõzõ maradékos osztásban kapott maradék. Az eljárást mindaddig ismételjük, ameddig zérus maradékot nem kapunk. Nyílvánvaló, hogy az algoritmus véges lépésben véget ér és az utólsó, zérustól különbözõ maradék, a

kiindulásul választott a és b pozitív egészek legnagyobb közös osztója. Ugyanez képletekkel felírva:

$$\begin{aligned}
a =& bq_0 + r_0 && (0 \leqslant r_0 < b)\\
b =& r_0q_1 + r_1 && (0 \leqslant r_1 < r_0)\\
r_0 =& r_1q_2 + r_2 && (0 \leqslant r_2 < r_1)\\
&\vdots && \vdots\\
r_{k-2} =& r_{k-1}q_k + r_k && (0 \leqslant r_k < r_{k-1})\\
&\vdots && \vdots\\
r_{n-2} =& r_{n-1}q_n + r_n && (0 \leqslant r_n < r_{n-1})\\
r_{n-1} =& q_{n+1}r_n
\end{aligned}$$

A kapott euklídeszi algoritmusból sorról sorra haladva egyszerû számolással kapjuk az $\frac{a}{b} \in \mathbb{Q}$ lánctört alakját:

$$\frac{a}{b} = q_0 + \frac{r_0}{b} = q_0 + \frac{1}{\frac{b}{r_0}} = q_0 + \frac{1}{q_1 + \frac{r_1}{r_0}} = q_0 + \frac{1}{q_1 + \frac{1}{\frac{r_0}{r_1}}} = \cdots$$

$$\cdots = q_0 + \cfrac{1}{q_1 + \cfrac{1}{\ddots \\ q_n + \cfrac{1}{q_{n+1}}}}.$$

Definiáljuk a $\mathrm{g} : \mathbb{Z}^+ \times \mathbb{Z}^+ \to \mathbb{Q}$ függvényt az alábbi módon:

$$\mathrm{g}(x, y) = x + \frac{1}{y} \qquad (x, y \in \mathbb{Z}^+),$$

és a g függvény segítségével rekurióval definiáljuk az $\mathrm{F}_k : \mathbb{Z}^k \to \mathbb{Q}$ függvényeket az alábbi módon:

$$\begin{aligned}
&\mathrm{F}_0(x) = x\\
&\mathrm{F}_n(x_1, \ldots, x_n, x_{n+1}) = \mathrm{F}_{n-1}(x_1, \ldots, x_{n-1}, \mathrm{g}(x_n, x_{n+1})) \qquad (n = 1, 2, \ldots)
\end{aligned}$$

Ekkor tetszõleges $\frac{a}{b} \in \mathbb{Q}$ számból kiindulva képezhetjük az euklídeszi algoritmus során fellépõ $q_0, q_1, \ldots$ hányadosok illetve a fenti $\mathrm{F}_0, \mathrm{F}_1, \ldots$ függvények segítségével az

$$\mathrm{LT}^{(k)}_{\frac{a}{b}} = \mathrm{F}_k(q_0, q_1, \ldots, q_k)$$

racionális számokat, melyeket a kiindulásul választott $\frac{a}{b}$ racioináli szám k-adik lánctört-konvergensének nevezzük ($k = 0, 1, 2, \ldots$). Világos, hogyha az euklídeszi algoritmus utólsó

0-tól különbözõ maradéka r_n, akkor az $n+1$-edik lánctörtkonvergens visszaadja az eredeti $\frac{a}{b}$ racionális számot.

Érdemes észrevenni, hogy a lánctörtkonvergensek feírásához csak az euklídeszi algoritmus végrehajrása során fellépõ hányadosok (q_n) sorozatára van szükségünk. Ez a sorozat könnyen megkapható maradékos osztások helyett az alsó egészrész függvény ($\lfloor\cdot\rfloor$) illetve a törtrészfüggvény ($\{\cdot\}$) alkalmazásával. Most a (q_n, r_n) sorozat helyett egy (q_n, s_n) sorozatot definiálunk rekurzióval.

$$\begin{aligned}
q_0 &= \max\{q \in \mathbb{Z} | a - bq \geqslant 0\}, \quad r_0 = a - bq_0 \quad \Rightarrow \\
&\Rightarrow \frac{a}{b} = \frac{r_0}{b} + q_0 \quad \Rightarrow \quad q_0 = \left\lfloor \frac{a}{b} \right\rfloor, \quad s_0 = \frac{r_0}{b} = \left\{ \frac{a}{b} \right\}, \\
q_1 &= \max\{q \in \mathbb{Z} | b - r_0 q \geqslant 0\}, \quad r_1 = b - r_0 q_1 \quad \Rightarrow \\
&\Rightarrow \frac{1}{s_0} = \frac{b}{r_0} = \frac{r_1}{r_0} + q_1 \quad \Rightarrow \quad q_1 = \left\lfloor \frac{1}{s_0} \right\rfloor, \quad s_1 = \frac{r_1}{r_0} = \left\{ \frac{1}{s_0} \right\}, \\
q_2 &= \max\{q \in \mathbb{Z} | r_0 - r_1 q \geqslant 0\}, \quad r_2 = r_0 - r_1 q_2 \quad \Rightarrow \\
&\Rightarrow \frac{1}{s_1} = \frac{r_0}{r_1} = \frac{r_2}{r_1} + q_2 \quad \Rightarrow \quad q_2 = \left\lfloor \frac{1}{s_1} \right\rfloor, \quad s_2 = \frac{r_2}{r_1} = \left\{ \frac{1}{s_1} \right\}, \\
&\dots
\end{aligned}$$

Illetve általánosan:

$$\begin{aligned}
q_k &= \max\{q \in \mathbb{Z} | r_{k-2} - r_{k-1} q \geqslant 0\}, \quad r_k = r_{k-2} - r_{k-1} q_k \quad \Rightarrow \\
\Rightarrow &\frac{1}{s_{k-1}} = \frac{r_{k-2}}{r_{k-1}} = \frac{r_k}{r_{k-1}} + q_k \quad \Rightarrow \quad q_k = \left\lfloor \frac{1}{s_{k-1}} \right\rfloor, s_k = \frac{r_k}{r_{k-1}} = \left\{ \frac{1}{s_{k-1}} \right\}.
\end{aligned}$$

Mivel a kapott (q_n) és (s_n) sorozatok az $\frac{a}{b}$ hányados függvényei, így a módszer az $\frac{a}{b}$ pozitív racionális szám helyett tetszõleges x pozitív irracionális szám esetén is alkalmazható.

Röviden összefoglalva, kiindulunk egy x pozitív irracionális számból, majd képezzük a (q_n) illetve (s_n) sorozatokat az alábbi rekurzióval:

$$\begin{aligned}
q_0 &= \lfloor x \rfloor, \quad s_0 = \{x\}, \\
q_k &= \left\lfloor \frac{1}{s_{k-1}} \right\rfloor, \quad s_k = \left\{ \frac{1}{s_{k-1}} \right\} \qquad (k = 1, 2, \dots).
\end{aligned}$$

Ekkor a racionális esettel analóg módon kapott

$$\mathrm{LT}_x^{(k)} = \mathrm{F}_k(q_0, q_1, \dots, q_k)$$

racionális számot az x pozitív valós szám k-adik lánctörtkonvergensének nevezzük ($k = 0, 1, 2, \dots$).

Irracionális számok racionális számokkal történõ approximációja során adott nevezõig a legjobb közelítést a lánctörtkonvergensek szolgáltatják. Felváltva alulról illetve felülrõl közelítik a számot, miközben a különbség (és így a hiba) exponenciálisan csökken.

16.2. Perióduskeresés sajátérték-becsléssel

A most bemutatásra kerülõ algoritmus nem Shor eredeti algoritmusa, hanem egy késõbbi javított változat, amely a sajátértékbecslést is felhasználja.

Most ismertetjük az algoritmust. Legyen N' a sejtett periódushoz képest nagy pozitív egész, N pedig egy olyan 2 hatvány, amelyre

$$\frac{1}{\sqrt{N}} \sim \frac{1}{\sqrt{N'}}.$$

(A módszer szépséghibája éppen ebben rejlik, mivel jelenleg csak 3, 4, 5, (egyes hírek szerint 7, sőt 9) bites gépeket tudunk kezelni, azonban titkosírás megfejtéséhez az alkalmazott prímszámok nagyságrendje miatt 500–1000 bites gépre lenne szükség.) Kiszámoljuk az alábbi uniform szuperpozíciót,majd alkalmazunk egy becslést:

$$\frac{1}{\sqrt{N\prime}} \sum_{0 \leqslant x < N\prime} |x> |f(x)> \approx \frac{1}{\sqrt{N}} \sum_{0 \leqslant x < N\prime}^{\square} |x> |f(x)>$$

A $\sum$ jel tetején látható kis $\square$ azt jelenti, hogy az összegzés az ismeretlen P szám N'-nél kisebb többszörösei közül a legnagyobbnál véget ér. A kapott állapottal dolgozunk tovább. A két állapot távolsága (azaz a hiba) mindvégig megõrzõdik. Megmérjük az f-et és azt kapjuk, hogy az eredmény:

$$\frac{1}{\sqrt{\frac{N}{P}}} \sum_{x=0}^{\frac{N}{P}} |x_0 + Px> .$$

ahol x_0 olyan, hogy $f(x_0)$ éppen a mért érték. A mért értékkel a továbbiakban nem foglalkozunk. Gondolatban alkalmazzuk a Fourier transzformációt, melynek segítségével felírjuk ezt az állapotot a T_1 ciklikus shift páronként ortogonális sajátvektorainak lineáris kombinációjaként, azaz

$$\frac{1}{\sqrt{\frac{N}{P}}} \sum_{x=0}^{\frac{N}{P}} |x_0 + Px> = \sum c_y U_y,$$

ahol

$$U_y = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \omega^{-jy} |j> .$$

Érdemes észrevenni, hogy ekkor U_y a T_1 ciklikus shift sajátvektora ω^y sajátértékkel, ugyanis

$$\mathrm{T}_1 U_y = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \omega^{-jy} \mathrm{T}_1 |j> = \omega^y \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \omega^{-(j+1)y} |j+1> = \omega^y U_y.$$

A c_y együtthatók Fourier transzformáció segítségével határozhatók meg az alábbi módon:

$$c_y = \frac{1}{\sqrt{\frac{N}{P}}} \frac{1}{\sqrt{N}} \sum_{x=0}^{\frac{N}{P}} \sum_{x=0}^{\frac{N}{P}} \omega^{y(x_0+Px)} = \frac{\omega^{yx_0}}{\sqrt{\frac{N}{P}}} \frac{1}{\sqrt{N}} \sum_{x=0}^{\frac{N}{P}} \omega^{yPx},$$

azaz c_y egy mértani sor konstansszorosának $\frac{N}{P}$-edik részletösszegeként határozható meg. A részletösszeg kiszámítása közben figyelni kell, hogy a mértani sor quotiense ω^{yP} nem egyenlõ, illetve egyenlõ 1-gyel. Ennek a két esetnek megfelelõen kapjuk, hogy a részletösszeg

$$\sum_{x=0}^{\frac{N}{P}} \omega^{yPx} = \begin{cases} \frac{\omega^{P\frac{N}{P}} - 1}{\omega^{yP} - 1} = 0 & \text{ha}, \omega^{yP} \neq 1 \\ \\ \frac{N}{P} & \text{egyébként.} \end{cases}$$

Amibõl kapjuk, hogy

$$|c_y| = \begin{cases} 0 & \text{ha}, \omega^{yP} \neq 1 \\ \\ \frac{|\ \omega^{yx_0}|}{\sqrt{\frac{N}{P}}} \frac{1}{\sqrt{N}} \frac{N}{P} = \frac{1}{\sqrt{P}} & \text{egyébként,} \end{cases}$$

ami öszhangban van azzal a ténnyel, hogy $\sum |c_y|^2 = 1$, mivel c_y pontosan azokra az y-okra nem tûnik el, amelyek $\frac{N}{P}$-nek többszörösei, és pontsan P számú ilyen y van. Tehát

$$|c_y| = \begin{cases} \frac{1}{\sqrt{P}} & \text{ha, } y\text{többszöröse } \frac{N}{P}\text{-nek} \\ \\ 0 & \text{egyébként,} \end{cases}$$

Az U_y-ok sajátvektorok, így alkalmazhatjuk a sajátértékbecslést. Az algoritmus következõ lépése legyen a következõ:

$$\sum c_y U_y |0> \longmapsto \sum c_y U_y |z_y>,$$

ahol z_y az $\frac{y}{N}$ tört egy jó bináris közelítése. Ugyanis az U_y sajátvektor sajátértéke

$$\omega^y = e^{\frac{2\pi i}{N} y} = e^{2\pi i \frac{y}{N}}.$$

Ekkor bevetünk még egy trükköt a z_y számítására. (Ez egy klasszikus számítás.)

$$\longmapsto \sum c_y U_y |\mathrm{LT}_y>,$$

ahol LT_y jelöli a z_y legjobb lánctörtkonvergensét legfeljebb N' nevezõvel. (Ha z_y nagyon közel van $\frac{y}{N}$-hez, akkor $\mathcal{O}(\log(N))$ lépésben megkapjuk $\frac{y}{N}$-et.)

Alkalmazunk egy mérést. Az $\frac{N}{P}$ többszöröseire egyenletes valószínûséggel $\frac{y}{N}$ redukált alakot kapjuk, azaz

$$y = a\frac{N}{P} \quad \Rightarrow \quad \frac{y}{N} = \frac{a}{P}.$$

Tehát valójában $\frac{a}{P}$ $(a = 0, 1, \ldots, P-1)$ alakú törteket kapunk egyforma valószínûséggel. Ebbõl $\frac{\phi(P)}{P}$ valószínûséggel a redukált alak nevezõje megmarad P-nek. Itt most ϕ az Euler-féle ϕ függvényt jelöli, mely definíció szerint a $0, 1, \ldots P-1$ sorozatban a P-hez relatív prímek számát jelöli, továbbá

$$\phi(n) = n\left(1 - \frac{1}{p_1}\right)\cdots\left(1 - \frac{1}{p_k}\right)$$

tetszõleges pozitív n esetén, ahol $p_1, \ldots p_k$ az n pozitív egész összes különbözõ prímosztói. Ismert számelméleti tétel alapján

$$\frac{\phi(P)}{P} > \frac{c}{\log\log P}$$

azaz $\mathcal{O}(\log\log P)$lépést elvégezve legalább 1-szer a nevezõ tényleg P.