1 Grundlagen

Axiome der Quantenmechanik

Ziel: Physik in Quantum computing zu elim.

kontinuierliche Quantenmodelle: formalisiert durch Evolution $U(t)$ von Zustand $\psi(t)$

Ann. •) U -- operator
- stetig $U(\varepsilon) = I - i\varepsilon H$ (H → Hamiltonian; $\varepsilon > 0$ klein)
- umkehrbar $U(t)^T U(t) = I$

$$\Rightarrow U(\varepsilon)\psi(t) = \psi(t+\varepsilon)$$

$$= \psi(t) - i\varepsilon H\psi(t)$$

$$\Leftrightarrow \frac{\psi(t+\varepsilon) - \psi(t)}{\varepsilon} = -iH\psi(t)$$

$$\lim_{\varepsilon \to 0} \Rightarrow \psi'(t) = -iH\psi(t)$$

Schrödinger Gleichung

$\rightsquigarrow$ Lösungsformel

$$\psi(p+t) = e^{-iHt}\psi(p)$$

$U(t)$ ... unitären Operator

$\rightsquigarrow$ motiviert Axiome

Dirac Notation bra/ket Vektoren

V -- komplexer Vektorraum

statt $v \in V \rightsquigarrow |v\rangle$ -- ket Vektor

falls $\dim V < \infty \rightsquigarrow$ wähle ONB

$\dim V = N$: $|0\rangle, |1\rangle, \dots, |N-1\rangle$

Definition Quantenzustand ist Superposition von klass. Zuständen

$$|\phi\rangle = \alpha_0 |0\rangle + \dots + \alpha_{N-1} |N-1\rangle$$

$\alpha_i \in \mathbb{C}$ ... Amplitude von $|i\rangle$ in $|\phi\rangle$

$\rightsquigarrow$ interpretiere $|\phi\rangle = \begin{pmatrix} \alpha_0 \\ | \\ \alpha_{N-1} \end{pmatrix}$ Vektor der Amplituden

$|\phi\rangle^H = (\bar{\alpha}_0, \ldots, \bar{\alpha}_{N-1}) =: \langle\phi| \ldots$ bra Vektor

$\rightsquigarrow$ innerer Produkt

$$\langle \phi, \psi \rangle =: \langle \phi | \psi \rangle = \langle \phi | \cdot | \psi \rangle$$

ab jetzt:

$V = H \ldots$ Hilbert Raum

(QM1) Der Zustand eines (isolierten) quanten Systems ist durch einen Einheitsvektor ($\|\cdot\|_H = 1$) in einem komplexen Hilbertraum gegeben.

Ex. $N=2$ $\quad |\phi\rangle = a|0\rangle + b|1\rangle \quad |a|^2 + |b|^2 = 1$

Def. Ein Quantensystem mit 2 dimensionalem Zustandsraum (ONB $|0\rangle, |1\rangle$) heißt qubit.

(QM2) Zusammensetzung von Systemen:

S_1 -- Zustandsraum V

S_2 -- Zustandsraum W

→ kombiniertes System: $V \otimes W$

tensor Produkt

Exkurs: Tensor Produkt von HR

V, W HR

Ziel: formale Def. von $V \otimes W$ "

•) Definiere „freien Vektorraum"

$\mathcal{F}(V,W)$: alle endl. Linearkomb. in $V \times W$

$$\mathcal{F}(V,W) = \left\{ \sum_{j=1}^{n} \alpha_j (v_j, w_j) : v_j \in V, w_j \in W, \alpha_j \in \mathbb{C} \right\}$$

abgeschlossen bzgl. $+, \cdot$ → VR

Anm. $(v_1, w) + (v_2, w)$ formal verschieden von $(v_1 + v_2, w)$!

• Übertrage VR Struktur von V, W auf $F(V,W)$

Ziel: $(v_1+v_2)\otimes w = v_1\otimes w + v_2\otimes w$

$v\otimes(w_1+w_2) = v\otimes w_1 + v\otimes w_2$

$c(v\otimes w) = (cv)\otimes w + v\otimes(cw)$

Def. Unterraum:

$$U(V,W) = \operatorname{span}\left\{ \sum_{j,k=1}^{n} \alpha_j \beta_k (v_j, w_k) - \left(\sum_{j=1}^{n} \alpha_j v_j, \sum_{k=1}^{n} \beta_k w_k\right)\right\}$$

$\Rightarrow$ ·) $U(V,W) \subseteq F(V,W)$

·) $U(V,W) \neq \emptyset$

·) $U(V,W)$ linear

$\Rightarrow$ Äquivalenz relation

$(v_1, w_1) \sim (v_2, w_2) \Leftrightarrow (v_1, w_1) - (v_2, w_2) \in U$

·) Betrachte $F(V,W)/\sim$

Äquivalent klasse $(v,w) \in F(V,W)/\sim$

$$\| \\ v \otimes w$$

$\rightsquigarrow$ VR Struktur

·) Topologie: $\langle \cdot,\cdot \rangle_V, \langle \cdot,\cdot \rangle_W$ IP auf V, W

definiere

$$\langle v_1 \otimes w_1, v_2 \otimes w_2 \rangle := \langle v_1, v_2 \rangle_V \langle w_1, w_2 \rangle_W$$

ist Sesquilinear form, weil

$b: (v_1, w_1, v_2, w_2) : (V \times W) \times (V \times W)$

$$\hookrightarrow \langle v_1, v_2 \rangle_V \langle w_1, w_2 \rangle_W$$

mult. linear (weil V, W HR)

$\otimes \times \otimes$: $(V \times W) \times (V \times W) \to (V \otimes W) \times V \otimes W)$

bilinear (nach Konstruktion)

$\Rightarrow b = \langle \cdot,\cdot \rangle \circ \otimes \times \otimes$

$\Rightarrow \langle \cdot,\cdot \rangle$ Sesquilinear

.) Definitheit: zeige $\langle \psi, \psi \rangle > 0$ für $\psi \neq 0$

sei $\psi = \sum_i \lambda_i v_i \otimes w_i \in F/\sim$

wähle ONB $\{\phi_i\}$ von $\operatorname{span}\{v_i\}$

$\{\chi_i\}$ von $\operatorname{span}\{w_i\}$

aus Def. der Äquivalenzklassen:

$$\psi = \sum_{j,k} \alpha_{jk} \phi_j \otimes \chi_k$$

$$\alpha_{jk} = \sum_i \lambda_i \langle \phi_j, v_i \rangle_V \langle \chi_k, w_i \rangle_W$$

$$\Rightarrow \langle \psi, \psi \rangle = \sum_{j,k} |\alpha_{jk}|^2 \geq 0$$

$\uparrow$ da sonst $\alpha_{jk} = 0$ $\forall j,k \Rightarrow \psi = 0$

•) Vervollständigung von $F/\sim$ bezüglich $\|\psi\|^2 = \langle \psi, \psi \rangle$

$\rightsquigarrow$ $V \otimes W$

Lemma Sei $\{\varphi_i\}$ ONB von V
$\{\psi_j\}$ ONB von W

$\Rightarrow \varphi_i \otimes \psi_j$ ist ONB von $V \otimes W$

Bem: Orthonormalität von Def. $\langle \cdot, \cdot \rangle$
+ONB von V, W

da φ_i, ψ_j ONBs $\Rightarrow$ span $\{\varphi_i \otimes \psi_j\}$
dicht in $F/\sim$
Gedichtin
$V \otimes W$ □

Bsp: $\mathbb{R}^2 \otimes \mathbb{R}^2 = \text{span}\{\begin{pmatrix}1\\0\end{pmatrix} \otimes \begin{pmatrix}1\\0\end{pmatrix}, \begin{pmatrix}1\\0\end{pmatrix} \otimes \begin{pmatrix}0\\1\end{pmatrix}, \begin{pmatrix}0\\1\end{pmatrix} \otimes \begin{pmatrix}1\\0\end{pmatrix}, \begin{pmatrix}0\\1\end{pmatrix} \otimes \begin{pmatrix}0\\1\end{pmatrix}\}$

$\Rightarrow$ dim = 4 „2-qubit State"

$\mathbb{R}^3 \otimes \mathbb{R}^2$ ... dim = 6

Allgemein: aus obigem Lemma:
dim $V = n$, dim $W = m \Rightarrow$ dim $V \otimes W = n \cdot m$
vgl.: dim $V \times W = n + m$!

Bsp. H HR $\Rightarrow H \otimes \mathbb{C}^n = H^n = H \times \dots \times H$

Bem. Konstruktion von Tensorprodukt ist eindeutig bis auf unitäre Transform.:

falls $\tilde{\otimes}$ -- sesquilineare Alt., verträglich mit $\langle \cdot | \cdot \rangle_V \langle \cdot | \cdot \rangle_W$

$\Rightarrow \exists U$ unitär: $V \otimes W \to V \tilde{\otimes} W$

---

n-qubit System: 2^n Basis Zustände

Schreibweise $|b_1 b_2 \dots b_n\rangle := |b_1\rangle \otimes \dots \otimes |b_n\rangle$

Basis Zustände werden bezeichnet zu

$$|0\rangle, \dots, |2^n - 1\rangle$$

$\rightsquigarrow$ Register von n-qubits: Superposition

$$\alpha_0 |0\rangle + \dots + \alpha_{2^n-1} |2^n - 1\rangle \qquad \sum_{j=0}^{2^n-1} |\alpha_j|^2 = 1$$

Definition Zustand $|\psi\rangle$ heißt

Produkt zustand, wenn

$$|\psi\rangle = |v_1\rangle \otimes \dots \otimes |v_n\rangle \qquad v_i \in H$$

falls nicht möglich: verschränkten Zustand

(auch EPR-Paar)

Bsp $\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$ verschränkt

„Bell state"

da $\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{2}|11\rangle = (a|0\rangle + b|1\rangle)\cdot(c|0\rangle + d|1\rangle)$

$$= ac|00\rangle + ad|01\rangle + bc|10\rangle + bd|11\rangle$$

$$\Rightarrow ac = bd = \frac{1}{\sqrt{2}} \quad \wedge \quad ad = bc = 0$$

$\lightning$ $\qquad a \vee d = 0$

$\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|10\rangle = \left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle\right) \otimes |0\rangle$

Produkt zustand

(QM3) Evolution eines Quantensystems geschieht nur durch unitäre Operatoren agierend am Zustandsraum

---

qubit $|\phi\rangle = \alpha_0|0\rangle + \alpha_1|1\rangle$ $|\alpha_0|^2 + |\alpha_1|^2 = 1$

$\rightarrow U|\phi\rangle = |\psi\rangle = \beta_0|0\rangle + \beta_1|1\rangle$

damit qubit $|\beta_0|^2 + |\beta_1|^2 = 1$

da U unitär ✓ $\rightarrow$ erhält Längen!

(QM4) Messung von Quantenzuständen: gibt nur Wahrscheinlichkeitsverteilung

Born Regel: Man sieht Zustand $|j\rangle$ mit Wahrscheinlichkeit $|\alpha_j|^2$

„Messung in Basis"

Messung ist invasiv (vgl. Schrödingers Katze)

$|\phi\rangle$ kollabiert zu klass. Zustand $|j\rangle$

Projektive Messung : V -- Zustandsraum

$\dim V = n < \infty$

Def. P_i ... orthog. Proj. auf $V_i \subseteq V$

$i = 1, \dots, m$

$P_i P_j = 0 \quad i \neq j$

$P_i P_i = P_i$

mit $\sum_{j=1}^{m} P_j = \mathbb{I}$

$\rightsquigarrow$ m mögliche Ergebnisse

$\rightsquigarrow$ $|\phi\rangle = \sum_{j=1}^{m} |\phi_j\rangle$ mit $|\phi_j\rangle = P_j |\phi\rangle \in V_j$

$\rightsquigarrow$ Messergebnis j mit Wahrscheinlichkeit

$$\| |\phi_j\rangle \|^2 = \langle \phi | P_j P_j | \phi \rangle = \langle \phi | P_j | \phi \rangle$$

und Zustand kollabiert zu

$$\frac{|\phi_j\rangle}{\| |\phi_j\rangle \|}$$

Bem. ·) Man kann nicht a priori auswählen, welche Proj. P_j angewandt wird (nur V_j)
→ nur Wahrsch.!

·) Aber wenn $|\phi\rangle \in V_j \rightsquigarrow$ Messung liefert j mit $P=1$

Bsp $V = \text{span}\{|0\rangle, \dots, |N-1\rangle\}$ $V_j = |j\rangle$

$\rightsquigarrow$ Born's Regel, d.h.

$$P_j |\phi\rangle = \alpha_j |j\rangle$$

Wahrsch. $\| P_j |\phi\rangle \|^2 = \| \alpha_j |j\rangle \|^2$

$= |\alpha_j|^2$

hier $P_j =: |j\rangle\langle j|$ rang 1

„Complete measurement"

Bsp Messung, die nur unterscheidet, ob $|j\rangle$ mit $j < N/2$ oder $j \geq N/2$

Projektoren

$$P_1 = \sum_{j < N/2} |j\rangle\langle j|$$

$$P_2 = \sum_{j \geq N/2} |j\rangle\langle j|$$

für $|\phi\rangle = \frac{1}{\sqrt{3}}|1\rangle + \sqrt{\frac{2}{3}}|N\rangle$

$\Rightarrow$ 1 mit Wahr. $\|P_1|\phi\rangle\|^2 = \frac{1}{3}$

2 mit —"— $\|P_2|\phi\rangle\|^2 = \frac{2}{3}$

„incomplete measurement"

Elementare Operationen auf qubits – gates

Definition Unitäre Operation auf kleiner Anzahl an qubits heißt „gate".

Vgl. AND OR XOR für bits

---

1 qubit gates

Bit flip X („NOT") tauscht $|0\rangle, |1\rangle$

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Phase flip spiegelt $|1\rangle$

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Spezialfall von R_ϕ -- Phasengate

$$R_\phi = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{pmatrix} \qquad (Z = R_\pi)$$

rotiert $|1\rangle$ um $\phi \in [-\pi, \pi]$

$R_{\frac{\pi}{4}}$ -- **T-gate**

Hadamard gate

wichtigste 1 qubit Op.

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

$$\rightsquigarrow H(|0\rangle) = H \cdot \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \simeq \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$$

$\rightsquigarrow$ gl. Wahrsch. für $|0\rangle, |1\rangle$

$$H = H^T = H^{-1} \Rightarrow H\left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle\right) = |0\rangle$$

---

2 qubit gates

Q: Wie implementiert man a XOR b?

a|b	0	1
0	0	1
1	1	0

$(a,b) \mapsto a \text{ XOR } b$

nicht unitär

$\rightsquigarrow$ mache $(a,b) \mapsto (a, a \text{ XOR } b)$

(a,b)	$\|00\rangle$	$\|01\rangle$	$\|10\rangle$	$\|11\rangle$	permutation
$(a, a \text{ XOR } b)$	$\|00\rangle$	$\|01\rangle$	$\|11\rangle$	$\|10\rangle$	$\rightsquigarrow$ unitary

„CNOT" controlled not

$CNOT\,|a\rangle|b\rangle = |a\rangle|b\rangle$

$CNOT\,|1\rangle|b\rangle = |1\rangle|1-b\rangle$

Gegeben 2. qubit („target qubit")

wenn 1. qubit 1 → („control qubit")

$$CNOT = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Circuits

Programmierung mittels quantum circuits

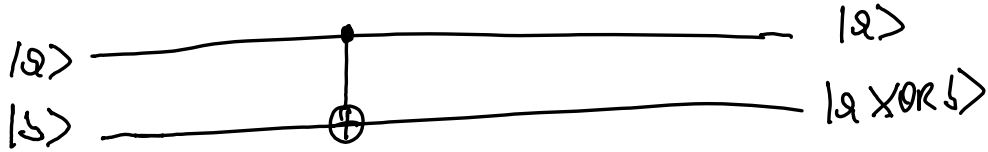
klass. Computer: Boolean circuit:

endlicher gerichteter Graph mit AND, OR, NOT Operationen

input → { AND, OR, NOTs } → output

Qu. Circuit: ersetze AND, OR, NOT durch Gates

Circuit diagram CNOT



---

3 qubit gate

Q: Wie geht a AND b ?

a\b	0	1
0	0	0
1	0	1

.) (a,b) ↦ (a, a AND b) nicht bijektiv
→ nicht unitär

.) → nehme 3 qubits

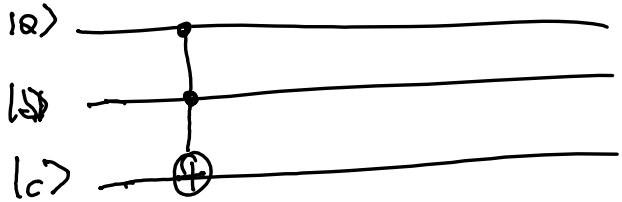
$|a\ b\ c\rangle \mapsto |a\ b\ c\,XOR(a\ AND\ b)\rangle$

|abc>	|000>	|001>	|010>	|011>	|100>	|101>	|110>	|111>
res.	|000>	|001>	|010>	|011>	|100>	|101>	|111>	|110>

für c=|0> → Output |a b a AND b>
unitär da Permutation

extra qubit $|c\rangle$ heißt „ancilla" qubit
= Diener

⇝ CCNOT gate oder Toffoli gate



---

allgemein: U unitär, n-qubit Operation

→ controlled-U

$$\begin{pmatrix} I & 0 \\ 0 & U \end{pmatrix} \in \mathbb{C}^{2^{n+1}} \times \mathbb{C}^{2^{n+1}}$$

I – Identität $\in \mathbb{R}^{2^n} \times \mathbb{R}^{2^n}$

Bem: Toffoli gate kann NOT realisieren $\text{Toffoli}(|11a\rangle) = |11\,1-a\rangle$
fixiere ersten beiden Qubits

da $A \text{ OR } B = \text{NOT}(\text{NOT } A \text{ AND NOT } B)$

→ jeder klass. circuit kann durch circuit von Toffolis implementiert werden

Quanten Information

wissen bisher:

.) 2 Zustände können nur mit Wahrsch. 1 unterschieden werden, wenn sie in orthogonalen VR liegen.

.) 3 mögliche Manipulationen für $|\psi\rangle$

- (Ancilla) keine $|A\rangle$ ~> kombiniere $|\psi\rangle|A\rangle$ (erhöht dim. des Zustandsraums)
- (Unitäre Op.) $U|\psi\rangle$
- (Messung) $|\psi\rangle \to$ kollabiert zu $|j\rangle$

Q: Wie kann Information kopiert/gespeichert werden?

A: „No cloning theorem" – starke Limitierung in Quantentheorie:

Zustände können nicht kopiert werden!

Setting: Zustandsraum H für

- 2 Quanten-Systeme
  - A (enth. $|\psi\rangle$ der kopiert werden soll)
  - B (enth. $|0\rangle$, Ziel der Kopie)
- 1 qu. System
  - M („Kopierer", Zustand $|M_0\rangle$)

Ziel: Operation

$$|\psi\rangle_A |0\rangle_B |M_0\rangle_M \to |\psi\rangle_A |\psi\rangle_B |M_\psi\rangle_M$$

die für alle Zustände in A funktioniert.

Theorem Sei $S \subseteq H$ so dass S zumindest ein paar versch., nicht-orthogonalen Zustände enthält

$\Rightarrow \nexists$ unit. Op. U auf S, die alle Zustände kopieren kann

Beweis Seien $|\xi\rangle, |\eta\rangle \in S$ nicht-orth.

$\rightsquigarrow$ will $U|\xi\rangle_A|0\rangle_B|M_0\rangle_M = |\xi\rangle_A|\xi\rangle_B|M_\xi\rangle_M$

$U|\eta\rangle_A|0\rangle_B|M_0\rangle_M = |\eta\rangle_A|\eta\rangle_B|M_\eta\rangle_M$

U unitär $\Rightarrow$ erhält Skalarprod.

$$\langle |\xi\rangle|\xi\rangle|M_\xi\rangle, |\eta\rangle|\eta\rangle|M_\eta\rangle\rangle = \langle |\xi\rangle|0\rangle|M_0\rangle, |\eta\rangle|0\rangle|M_0\rangle\rangle$$

$$\langle\xi|\eta\rangle\langle\xi|\eta\rangle\langle M_\xi|M_\eta\rangle \qquad \langle\xi|\eta\rangle\underbrace{\langle 0|0\rangle}_{|\cdot|=1}\underbrace{\langle M_0|M_0\rangle}_{|\cdot|=1}$$

$$\Rightarrow |\langle\xi|\eta\rangle|^2 |\langle M_\xi|M_\eta\rangle| = |\langle\xi|\eta\rangle|$$

(kürzen ob $|\langle\xi|\eta\rangle| \neq 0$ wegen nicht-orth.)

Cauchy-Schwarz Ungl.:

$$|\langle M_\xi|M_\eta\rangle| \leq \underbrace{|\langle M_\xi|M_\xi\rangle|}_{=1} \underbrace{|\langle M_\eta|M_\eta\rangle|}_{=1}$$

$$\Rightarrow |\langle\xi|\eta\rangle| = 1$$

$\Rightarrow \xi = e^{i\phi}\eta$ kann nicht für alle Zustände stimmen ↯ □

Zeitumkehr : no deleting theorem

$\exists$ unit. Op sd.

$$U : |\psi\rangle_A |\psi\rangle_B |M_0\rangle_M \to |\psi\rangle_A |0\rangle_B |M_\psi\rangle_M$$

Lösung für Informationstransfer: Verschränkung

Bsp: Quanten Teleportation

Setting : .) Alice → qubit $\alpha_0|0\rangle + \alpha_1|1\rangle =: |\alpha\rangle$

.) Bob → weit entfernt, will Information von Alice qubit haben

.) A+B haben noch ein weiteres verschränktes qubit

$$\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

(Alice, Bob, Alice, Bob)

Q: Wie bekommt Bob die Information ohne physischen Transfer von Alice qubit?

Teleportation:

(i) 3 qubits im Spiel 1.qubit $|\alpha\rangle$
2.qubit Alice Teil von verschr. Qubit
3.qubit Bobs Teil von verschr. qubit

komb. System:

$$(\alpha_0|0\rangle + \alpha_1|1\rangle) \otimes \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

(ii) Alice CNOT auf 1., 2. qubit

(iii) Alice Hadamard gate auf 1. qubit

$\Rightarrow$ CNOT: $\frac{\alpha_0}{\sqrt{2}}|0\rangle(|00\rangle + |11\rangle) + \frac{\alpha_1}{\sqrt{2}}|1\rangle(|01\rangle + |10\rangle)$

Hadam.: $\frac{1}{2}\alpha_0(|0\rangle + |1\rangle)(|00\rangle + |11\rangle) +$

$\frac{1}{2}\alpha_1(|0\rangle - |1\rangle)(|10\rangle + |01\rangle)$

$$= \frac{1}{2}|00\rangle(\alpha_0|0\rangle + \alpha_1|1\rangle) +$$
$$\frac{1}{2}|01\rangle(\alpha_0|1\rangle + \alpha_1|0\rangle) +$$
$$\frac{1}{2}|10\rangle(\alpha_0|0\rangle - \alpha_1|1\rangle) +$$
$$\frac{1}{2}|11\rangle(\alpha_0|1\rangle - \alpha_1|0\rangle)$$

Alice qubits

(iv) Alice misst beide qubits
⇒ 4 mögliche Zustände mit $P = \frac{1}{4}$

Ergebnis:

Messung	Zustand nach Messung	
00	$\|00\rangle \|\alpha\rangle$	
01	$\|01\rangle X\|\alpha\rangle$	bitflip
10	$\|10\rangle Z\|\alpha\rangle$	phaseflip
11	$\|11\rangle XZ\|\alpha\rangle$	bit+phaseflip

(v) Alice sendet Messergebnis auf klass. Kanal zu Bob
z.B. 10

(vi) Bob sieht in obiger Tabelle nach und wendet die zugeh. Inverse transf. auf sein Qubit an

für 10 $\rightsquigarrow Z$

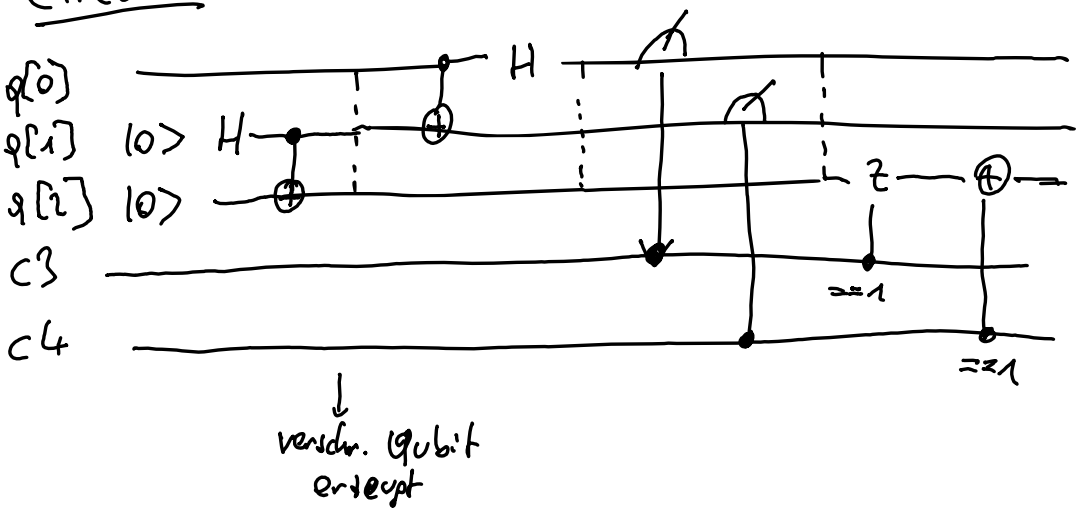
$\Rightarrow$ Sein Qubit ist garantiert im Zustand $|\alpha\rangle$ □

Bem.: .) nicht-lokale Verbindung: physik. Prozesse im Ort zw. A/B beeinflussen nichts!

.) nur Information transportiert, keine Materie → kein Star Trek Beamen!

.) Alice misst ihre Qubits → danach Superposition weg!

Circuit



c3, c4 klass. Zust.

Quantum Gates

How to implement a XOR b?

a \ b	0	1
0	0	1
1	1	0

but $(a, b) \mapsto a \text{ XOR } b$ not unitary

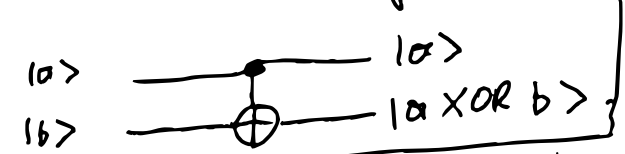
hence $(a, b) \mapsto (a, a \text{ XOR } b)$

(a,b)	$\|00\rangle$	$\|01\rangle$	$\|10\rangle$	$\|11\rangle$
$(a, a \text{ XOR } b)$	$\|00\rangle$	$\|01\rangle$	$\|11\rangle$	$\|10\rangle$

is permutation $\Rightarrow$ unitary

This operation is called „Controlled Not", „CNOT"

In circuit diagrams, this is denoted by



if we identify $|00\rangle, \ldots, |11\rangle$ with $e_1, \ldots, e_4$

we may represent CNOT by the unitary matrix

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

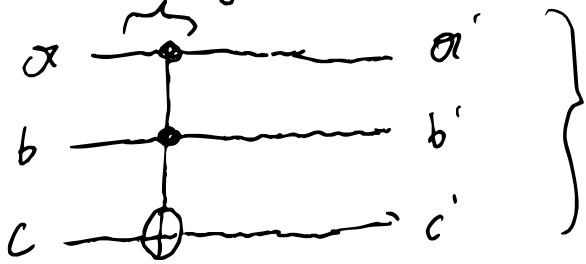
What about „AND"?

a AND b $\Leftrightarrow$

$a \backslash b$	0	1
0	0	0
1	0	1

- $(a,b) \mapsto (a,\ a$ AND $b)$ not bijective
- requires three qbits $|a,b,c\rangle \mapsto |a,b,c$ XOR $(a$ AND $b)\rangle$

Toffoli gate



} unitary as composition of q-gates

$\|abc\rangle$	000	001	010	011	100	101	110	111
	000	001	010	011	100	101	111	110

If Input $c = |0\rangle \Rightarrow$ Output $c' = |a,b,a$ AND $b\rangle$

- extra qbit $|c\rangle$ is called „ancilla" qbit

  ancilla = Diener

Deutsch-Jozsa-Algorithm

"Problem": Given $F(x_1, \ldots, x_n) \in \{0,1\}$ $x_i \in \{0,1\}$

"determine if
- $F = 0$
- $F = 1$
- F is "balanced", i.e. exactly half of $(x_1 \ldots x_n)$ lead to $F(x_1 \ldots x_n) = 1$ the other half $\Rightarrow F(x_1 \ldots x_n) = 0$

- We assume the F satisfies one of the three options
- Not useful in practice, but demonstrates that hard problems can be easier on quantum computers

---

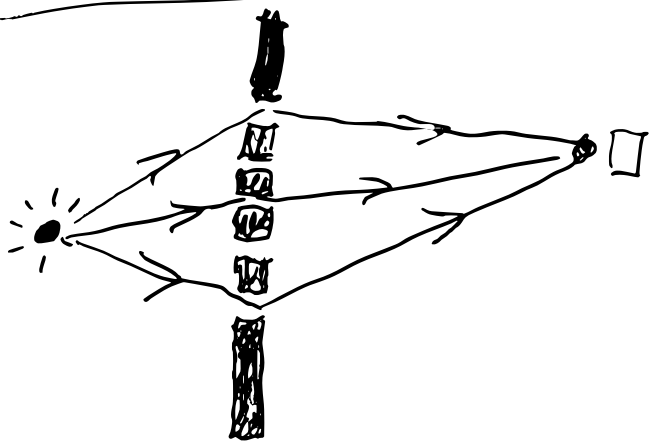
Classical algorithm:

To exclude the "balanced" case, we need to check at least $2^{n-1}+1$ inputs

$(x_1 \ldots x_n) \in \{0,1\}^n$

$\Rightarrow$ Problem size $N = 2^n$, Runtime $N/2+1$

Physical inspiration



Intensity at detector is determined by "phase" difference of light-paths

→ interference

Number the holes with $x \in \{0,1\}^n$ } 2^n holes

Assume the holes are located such that the light amplitude is $\simeq (-1)^{F(y)}$ at detector if light travels through hole y.

Light intensity at detector is given by

$$\simeq \sum_{y \in \{0,1\}^n} (-1)^{F(y)}$$

$= 0$ if F is balanced

$= \pm 1$ if F is const

Quantum Algorithm

We may identify $(x_1,\dots,x_n)\in\{0,1\}^n$ with the basis element $|i\rangle$, where $i = x_1 + x_2 2 + x_3 4 + \dots + x_n 2^{n-1}$

In this sense, define $\overline{f}(|i\rangle) := |f(x_1,\dots,x_n)\rangle$

An essential part of this (and many other algorithms) is the „Query":

Given 1-qbit state $|b\rangle$ and $f(x_1,\dots,x_n)\in\{0,1\}$, define

$$Q_f(|i\rangle\otimes|b\rangle) = |i\rangle\otimes(|b\rangle\oplus f(|i\rangle))$$

Particularly for $|b\rangle = |-\rangle := \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$,

we have

$$Q_f(|i-\rangle) = |i\rangle\otimes\frac{1}{\sqrt{2}}\left[\overline{f}(|i\rangle) - |1\rangle\oplus\overline{f}(|i\rangle)\right]$$

$$= \begin{cases} |i\rangle\otimes\frac{1}{\sqrt{2}}[|0\rangle-|1\rangle] & f(|i\rangle)=|0\rangle \\ |i\rangle\otimes\frac{1}{\sqrt{2}}[|1\rangle-|0\rangle] & f(|i\rangle)=|1\rangle \end{cases}$$

$$= (-1)^{\overline{f}(|i\rangle)}|i-\rangle$$

→ Note that we identify $\overline{f}(|i\rangle)$ with $f(x_1,\dots,x_n)$ instead of $|f(x_1,\dots,x_n)\rangle$, i.e., the last equality only makes sense for basis element $|i\rangle$

Note that Q_f is unitary since

$$|b\rangle \mapsto (|b\rangle \oplus f(|a\rangle))$$

is permutation for any value of $f(a)$

We make extensive use of the Hadamard gate $\boxed{H}$ given by $\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$

we already know:

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

Lemma 1: There holds

$$H^{\otimes n}|0^n\rangle := \bigotimes_{i=1}^{n} H|0\rangle = \bigotimes_{i=1}^{n} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$
$$= \sum_{\substack{(x_1,\dots,x_n) \\ \in \{0,1\}^n}} \frac{1}{\sqrt{2^n}} |x_1 \dots x_n\rangle = \frac{1}{\sqrt{N}} \sum_{i=0}^{2^n-1} |i\rangle$$

as well as for $i = x_1 + 2x_2 + \dots + 2^{n-1}x_n$

$$H^{\otimes n}|i\rangle := \bigotimes_{i=1}^{n} H|x_i\rangle = \bigotimes_{i=1}^{n} \frac{1}{\sqrt{2}}(|0\rangle + (-1)^{x_i}|1\rangle)$$
$$= \sum_{\substack{(y_1\dots y_n) \\ \in \{0,1\}^n}} \frac{1}{\sqrt{N}} (-1)^{x\cdot y} |y_1 \dots y_n\rangle$$
$$= \frac{1}{\sqrt{N}} \sum_{j=0}^{2^n-1} (-1)^{i\cdot j} |j\rangle, \quad \text{where}$$

$$i \cdot j := x \cdot y := \sum_{k=1}^{n} x_k y_k .$$

The Algorithm reads:

1) Initial state $|0^n\rangle := \underbrace{|0\rangle \otimes \cdots \otimes |0\rangle}_{n\text{-times}}$

2) Apply $\boxed{H}$ to each qbit to obtain uniform state

$$\frac{1}{\sqrt{N}} \sum_{i=0}^{2^n-1} |i\rangle$$

3) Tensorize result with $|-\rangle \Rightarrow \frac{1}{\sqrt{N}} \sum |i\rangle -\rangle$

3) Apply query $\boxed{O_f}$ with $|b\rangle = |-\rangle$ to obtain

$$\frac{1}{\sqrt{N}} \sum_{i=0}^{2^n-1} (-1)^{f(|i\rangle)} |i\rangle -\rangle$$

4) Ignore last qbit $|-\rangle$ and apply $\boxed{H}$ to remaining state to obtain (Lemma 1)

$$\frac{1}{N} \sum_{i=0}^{2^n-1} \sum_{j=0}^{2^n-1} (-1)^{i \cdot j} (-1)^{f(|i\rangle)} |j\rangle$$

5) The amplitude of the $j=0 \Leftrightarrow |0^n\rangle$-state is

$$\frac{1}{N} \sum_{i=0}^{2^n-1} \underbrace{(-1)^{i \cdot 0}}_{=1} (-1)^{f(|i\rangle)}$$

- If f is „balanced" $\Rightarrow$

$$\frac{1}{N}\sum_{i=0}^{2^n-1}(-1)^{f(|i\rangle)} = 0$$

- If $f=1 \Rightarrow \frac{1}{N}\sum_{i=0}^{2^n-1}(-1)^{f(|i\rangle)} = -1$
- If $f=0 \Rightarrow \frac{1}{N}\sum_{i=0}^{2^n-1}(-1)^{f(|i\rangle)} = 1$

Complexity of Quantum Alg:

$O(1)$ steps, $O(\log_2(N))$ quantum gates

$O(\log(N))$ qbits $\Rightarrow$ „exponential speedup"

Example for f:

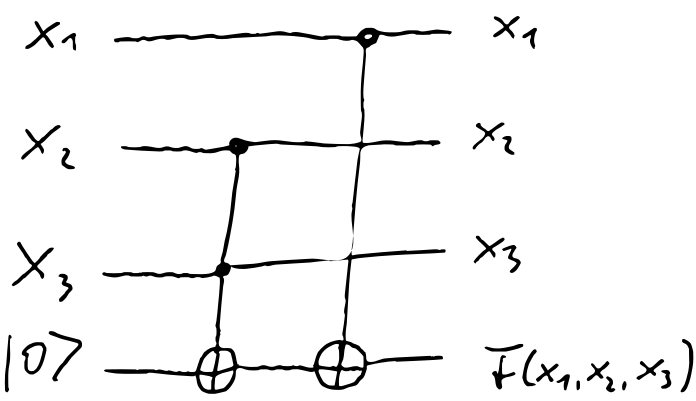
- $f(x_1, x_2, x_3) = \mathrm{mod}(x_1 + x_2 x_3, 2)$

$\Rightarrow f$ is balanced since

$$f(0, x_2, x_3) = 1 - f(1, x_2, x_3)$$

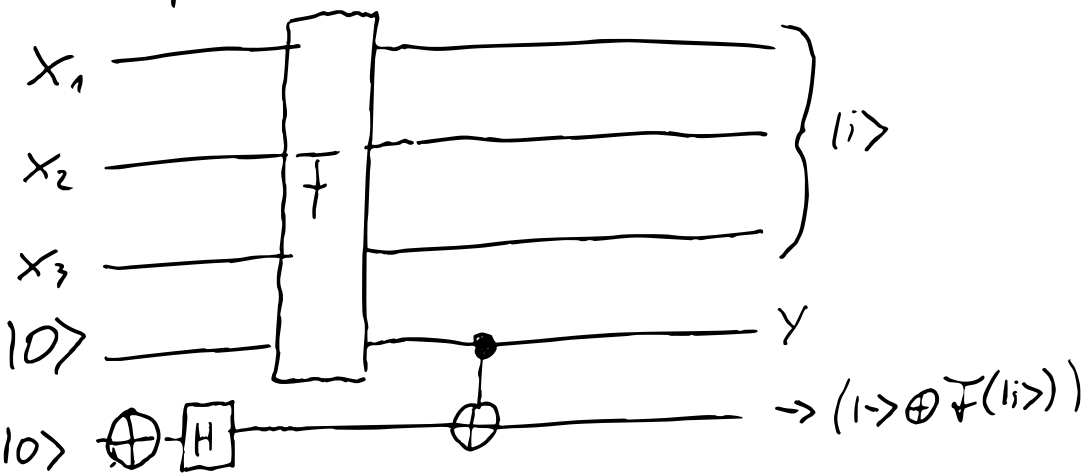
For the algorithm to work, f needs to be a quantum circuit, i.e.

$f(x_1, x_2, x_2) = x_1$ XOR (x_2 AND x_3)

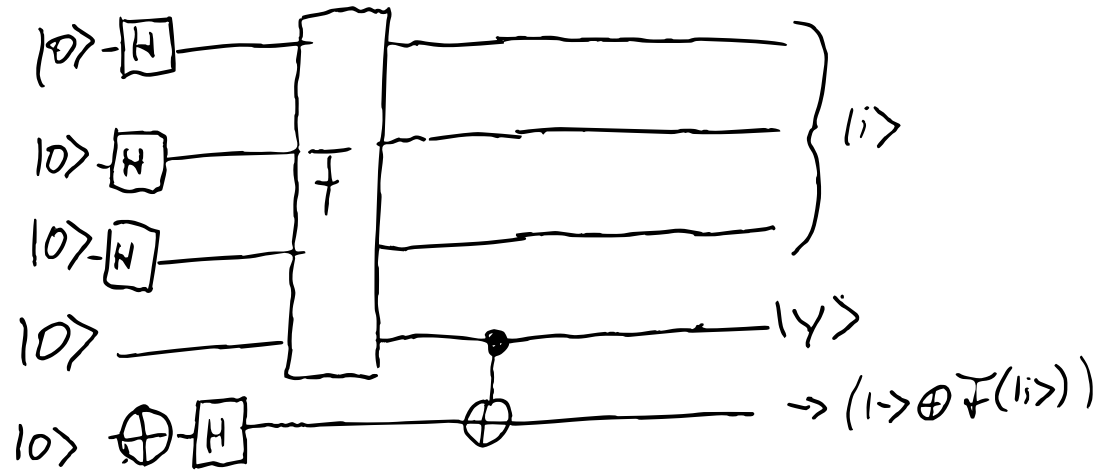


(ancilla qbit

Next, implement Query $Q_f(|i\rangle) = |i\rangle \otimes (|-\rangle \oplus f(|i\rangle))$

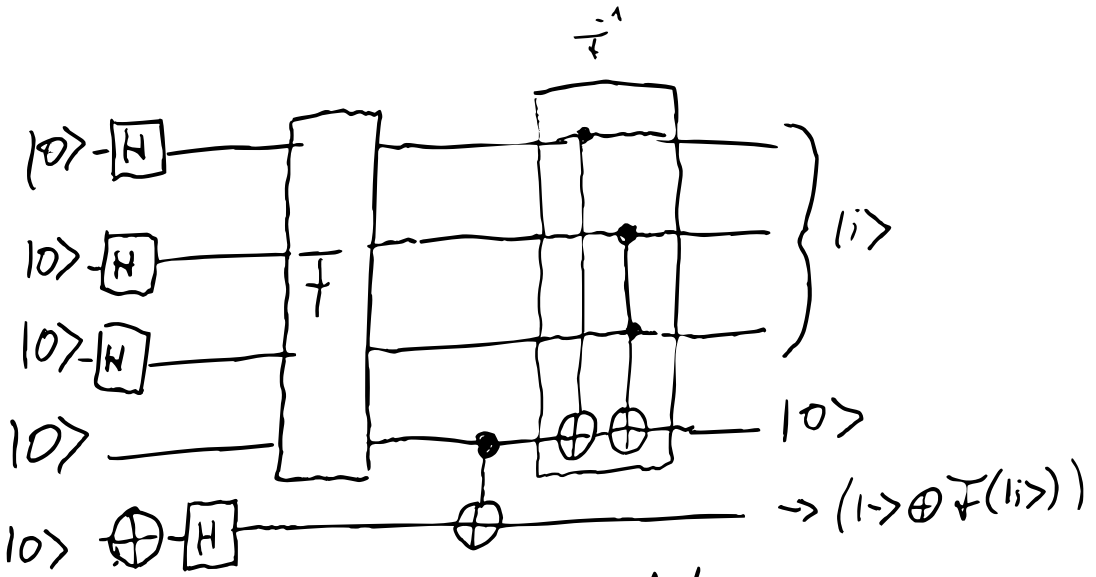


Apply Steps 1-3 of ALG



Problem: $|y\rangle$ and $|i\rangle$ are entangled

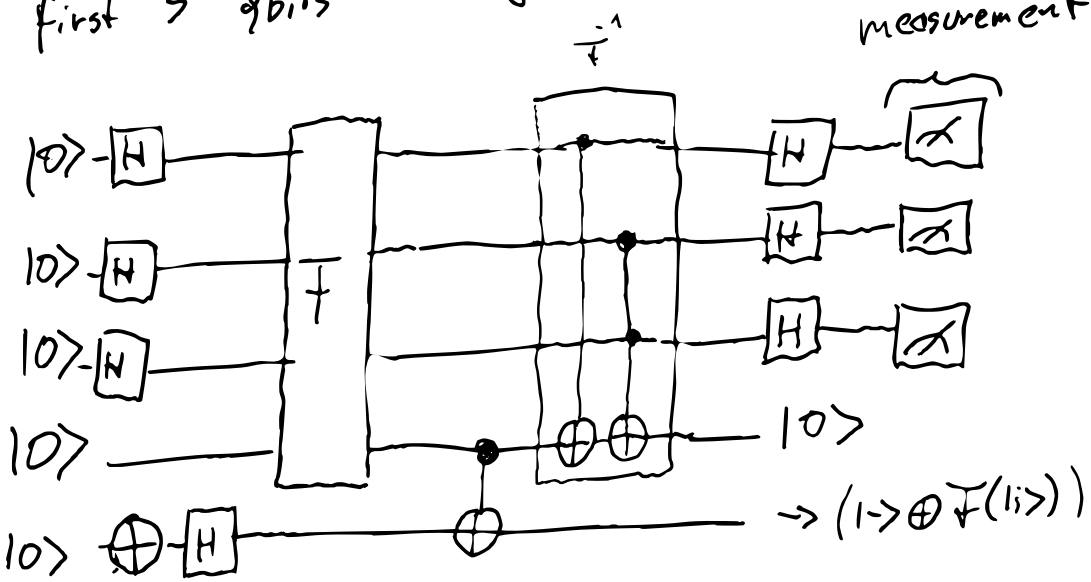
Solution: reverse F such that $|y\rangle = |0\rangle$



Now, we have the desired state

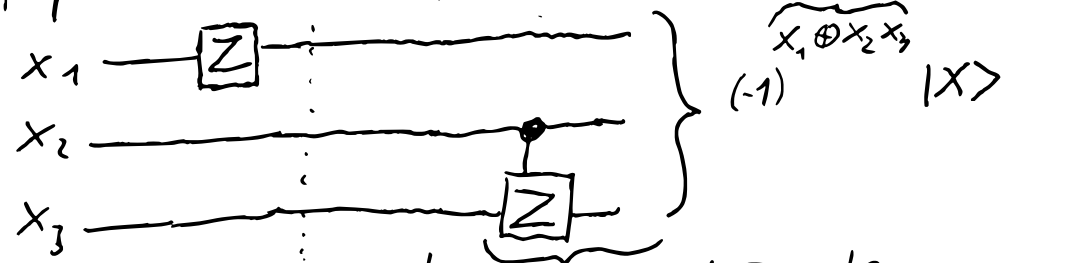
$$|i\rangle \otimes (|-\rangle \oplus F(|i\rangle)) \otimes |0\rangle$$

Apply step 4 of Alg and measure the first 3 qbits



The measurment must contain the projection onto $|000\rangle \otimes \text{span}\{|0\rangle,|1\rangle\}^{\otimes 2}$

Note that this is not the most efficient implementation of Q_F



at this point we have $(-1)^{x_1}|x\rangle$

Controlled Z-gate: Z-gate only active if $x_2 = 1$

$\Rightarrow (-1)^{x_2 x_3}|x\rangle$

$$\boxed{Z} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Remarks on Deutsch-Jozsa:

- First algorithm with exponential speedup over classical computer

- However, consider the following randomized classical algorithm $R(f)$:
  1) generate two random $x, y \in \{0,1\}^n$
  2) Evaluate $f(x)$, $f(y)$
  3) Output: $R(f) = 1$ if $f(x) = f(y) = 1$
     $R(f) = 0$ if $f(x) = f(y) = 0$
     $R(f)$ „balanced" if $f(x) \neq f(y)$

- The algorithm is correct if f is constant and answers correctly with prob. $\frac{1}{2}$ if f is balanced.

- Apply the alg k-times with i.i.d random samples to get $(R_1(f), \dots, R_k(f))$. Return
  1 if $R_1(f) = \dots = R_k(f) = 1$
  0 if $R_1(f) = \dots = R_k(f) = 0$
  „balanced" else

$\Rightarrow$ Probability of error 2^{-k}, cost $O(k)$

Remark: Why do you have to "uncompute f" in order to remove entanglement?

After step 3, we have

$$\frac{1}{\sqrt{N}} \sum_{i=0}^{2^n-1} (-1)^{f(|i\rangle)} |i\rangle$$

but in implementation, we actually have

$$\frac{1}{\sqrt{N}} \sum_{i=0}^{2^n-1} (-1)^{f(|i\rangle)} |i\rangle \otimes |0_i\rangle \otimes |-\rangle$$

for some ancilla qbit $|0_i\rangle$. We may ignore last qbit since not entangled.

Step 4 applys $H^{\otimes n}$ to obtain

$$\frac{1}{N} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (-1)^{f(|i\rangle)} (-1)^{i \cdot j} |j\rangle \otimes |0_i\rangle$$

$\Rightarrow$ Amplitude of $|j\rangle = |0\rangle$ $\Rightarrow$

$$\frac{1}{N} \sum_{i=0}^{N-1} (-1)^{f(|i\rangle)} |0\rangle \otimes |0_i\rangle$$

Might be non-zero even for balanced f

Simon's algorithm

First quantum alg. with exp speedup over any (randomized) classical alg.

Given $i,j \in \{0,1\}^n$, define $i \oplus j := (i_1 \oplus j_1, \dots, i_n \oplus j_n)$

Simon's problem: Let $f : \{0,1\}^n \to \{0,1\}^n$ s.t. there exists $s \in \{0,1\}^n$

$$f(x) = f(y) \iff x = y \text{ or } x \oplus s = y$$

$\forall x, y \in \{0,1\}^n$.

Goal: Find s.

Quantum Algorithm

1) Start with $|0^n\rangle|0^n\rangle$ and apply $\boxed{H}$ to the first n qbits to obtain

$$\frac{1}{\sqrt{N}} \sum_{i=0}^{2^n-1} |i\rangle|0^n\rangle$$

2) Apply the query $|i\rangle \otimes |b^n\rangle \mapsto |i\rangle \otimes (|b^n\rangle \oplus f(|i\rangle))$ with $|b^n\rangle = |0^n\rangle$ to obtain

$$\frac{1}{\sqrt{N}} \sum_{i=0}^{2^n-1} |i\rangle|f(|i\rangle)\rangle$$

3) Measure the second n qbits in computational basis, i.e. $\operatorname{span}\{|i\rangle, i=0,\dots 2^n-1\} \otimes \operatorname{span}\{|j\rangle\}$ to $j = 0, \dots, 2^n-1$

obtain some output

$$|a\rangle \otimes |j\rangle = \frac{P_j\left(\frac{1}{\sqrt{N}} \sum |i\rangle F(|i\rangle)\right)}{\| P_j(\cdots) \|}$$

Note that

$$|i\rangle F(|i\rangle) \perp \operatorname{ran} P_j \quad \text{for} \quad F(|i\rangle) \neq |j\rangle$$

by assumption, there exist exactly two inputs $|i_0\rangle$ and $|i_1\rangle = |i_0 \oplus s\rangle$ with $F(|i_k\rangle) = |j\rangle$ $k = 0, 1$.

Hence

$$|a\rangle = \frac{1}{\sqrt{2}} \left(|i_0\rangle + |i_0 \oplus s\rangle \right)$$

4) Ignore second n qbits and apply $\boxed{H}$ to the first n to obtain

$$\frac{1}{\sqrt{2N}} \sum_{j=0}^{2^{n-1}} \left[(-1)^{i_0 \cdot j} + (-1)^{(i_0 \oplus s) \cdot j} \right] |j\rangle$$

5) Measure in computational basis

- $|j\rangle$ has non-zero amplitude if

$$(i_0 \oplus s) \cdot j \underset{\text{mod } 2}{=} i_0 \cdot j \iff s \cdot j = 0 \mod 2$$

$\Rightarrow$ We obtain random element of the set

$$\{ j \mid s \cdot j = 0 \mod 2 \}$$

6) Repeat the procedure to obtain $n-1$ linearly ind. elements $j^{(1)}, \dots, j^{(n-1)}$ with

$$j^{(i)} \cdot s = 0 \mod 2$$

or

$$\begin{pmatrix} j^{(1)} \\ \vdots \\ j^{(n-1)} \end{pmatrix} s = 0 \mod 2$$

7) Solve linear system mod 2 on classical computer in $O(n^3)$

Note that $\#\,\mathrm{span}\{j^{(1)}, \dots, j^{(k)}\} \le 2^k$

Hence, with probability $\frac{2^n - 2^k}{2^n} = 1 - 2^{k-n} \ge \frac{1}{2}$

for $k \le n-1$, we find

a linearly independent vector j^{k+1}

Conclusion: number of qbits $O(n)$, number of gates $O(n)$, number of iterations $O(n)$ with high probability $+ O(n^3)$

Classical algorithms for Simon's problem:

Lemma 2 Any (randomized) classical algorithm with less than $\delta 2^{\frac{n}{2}}$ queries to f fails with probability $\geq e^{-\frac{5}{4}\delta^2}$. (deterministic alg. fail certainly if less than $2^{\frac{n}{2}}$ queries).

Proof Every alg generates a sequence of queries $x_1, \dots, x_k$ with $f(x_1), \dots, f(x_k)$ $= y_1 \quad\quad = y_k$

If all y_i are distinct, the alg can't distinguish between $s=0$ and $s\neq 0$.

Assume all $y_1, \dots, y_k$ are distinct. Then the alg chooses x_{k+1} based on some prob. measure μ on $\{0,1\}^n$ [in the det. case, μ is a delta distribution].

$$\sum_{k=1}^{c} \sum_{s\in\{0,1\}^n} \sum_{i=1}^{k} \mu(x_i \oplus s) = \sum_{k=1}^{c} \sum_{i=1}^{k} \underbrace{\sum_{s} \mu(x_i \oplus s)}_{=1}$$

$$= \frac{c(c+1)}{2}$$

$$\Rightarrow \exists s \in \{0,1\}^n : \sum_{k=1}^{c} \underbrace{\sum_{i=1}^{k} \mu(x_i \oplus s)}_{p_i \leq \frac{1}{2}} \leq \frac{c(c+1)}{2^{n+1}} \leq \frac{1}{2} \quad \text{for } c(c+1) \leq 2^n$$

Hence,

$$\mathbb{P}(y_1, \dots, y_{k+1} \text{ distinct}) = \underbrace{\mathbb{P}(y_{k+1} \notin \{y_1, \dots, y_k\} \mid y_1, \dots, y_k \text{ distinct})}_{\geq 1 - p_i} \cdot \mathbb{P}(y_1, \dots, y_k \text{ distinct})$$

$$\geq 1 - p_i$$

Iterate the argument to obtain

$$P(y_1 \dots y_c \text{ distinct}) \geq \prod_{i=1}^{c} (1-p_i)$$

Taking logarithms shows

$$\log\left(\prod_{i=1}^{c}(1-p_i)\right) = \sum_{i=1}^{c} \log(1-p_i)$$

$$\left(p_i \leq \tfrac{1}{2}\right) \rightarrow \geq -\frac{5}{4}\sum_{i=1}^{c} p_i \geq -\frac{5}{4}\,\frac{c(c+1)}{2^{n+1}}$$

$$\Rightarrow P(y_1 \dots y_c \text{ distinct}) \geq e^{-\frac{5}{4}\frac{c(c+1)}{2^{n+1}}}$$

Choosing $c+1 = \delta\, 2^{\frac{n}{2}}$ satisfies $c(c+1) \leq 2^n$

and $P(y_1, \dots, y_c \text{ distinct}) \geq e^{-\frac{5}{4}\delta^2}$

□

$$\log(1-p) = 0 + \frac{1}{1}(-p) - \frac{1}{2\xi^2}(-p)^2 \quad \text{for some } 1-p \leq \xi \leq p$$

$$\geq -p - \frac{p^2}{2} = -p\left(1 + \frac{p}{2}\right)$$

$$\geq -\frac{5}{4}p$$

The quantum Fourier Transform

Discrete FT: for $x_0, \dots, x_{N-1}$ define

$$\hat{x}_k := \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j \, e^{-\frac{2\pi i}{N} kj}$$

Inverse

$$x_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \hat{x}_j \, e^{\frac{2\pi i}{N} kj}$$

in matrix notation

$$\hat{x} = F_N x \quad \text{with} \quad F_N \in \mathbb{C}^{N \times N}$$

$$\left(F_N\right)_{kj} := \frac{1}{\sqrt{N}} e^{-\frac{2\pi i}{N} kj}$$

F_N is unitary matrix ✓

Fast FT: Assume $N = 2^n$

$$\hat{x}_k = \frac{1}{\sqrt{2}} \left(\frac{1}{\sqrt{N/2}} \sum_{\substack{j=0 \\ j \text{ even}}}^{N-1} x_j \, e^{-\frac{2\pi i}{N/2} k \frac{j}{2}} + e^{-\frac{2\pi i}{N} k} \frac{1}{\sqrt{N/2}} \sum_{j \text{ odd}} e^{-\frac{2\pi i}{N/2} k \frac{j-1}{2}} \right)$$

$\Rightarrow$ split FT into 2 FTs of half size

leads to $O(N \log N)$ complexity.

The QFT maps a state

$$|x\rangle = \sum_{j=0}^{N-1} \hat{x}_j |j\rangle \quad \text{to} \quad |y\rangle = \sum_{j=0}^{N-1} x_j |j\rangle$$

where x_j is given by the classical FT

If $|x\rangle = |j_0\rangle$, then $x_j = \delta_{jj_0}$ and hence

$$\hat{x}_k = \frac{1}{\sqrt{N}} e^{\frac{2\pi i}{N} k j_0}$$

This is a convention. Everything works with FT

$$\Rightarrow |j\rangle \xmapsto{QFT} \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{\frac{2\pi i}{N} kj} |k\rangle$$

$$= F_N |j\rangle$$

To implement F_N efficiently, we observe

$$\frac{k}{N} = \frac{k}{2^n} = \sum_{e=1}^{n} k_e 2^{-e} \quad \text{if } k = k_1 2^{n-1} + k_2 2^{n-2} + \dots + k_n$$

$$F_N |j\rangle = \frac{1}{\sqrt{2^n}} \sum_{k \in \{0,1\}^n} e^{2\pi i j \sum_{e=1}^{n} k_e 2^{-e}} |k_1 \dots k_n\rangle$$

$$= \bigotimes_{e=1}^{n} e^{-2\pi i j k_e 2^{-e}} |k_e\rangle$$

$$= \bigotimes_{e=1}^{n} \left(|0\rangle + e^{2\pi i j / 2^e} |1\rangle \right) \frac{1}{\sqrt{2}}$$

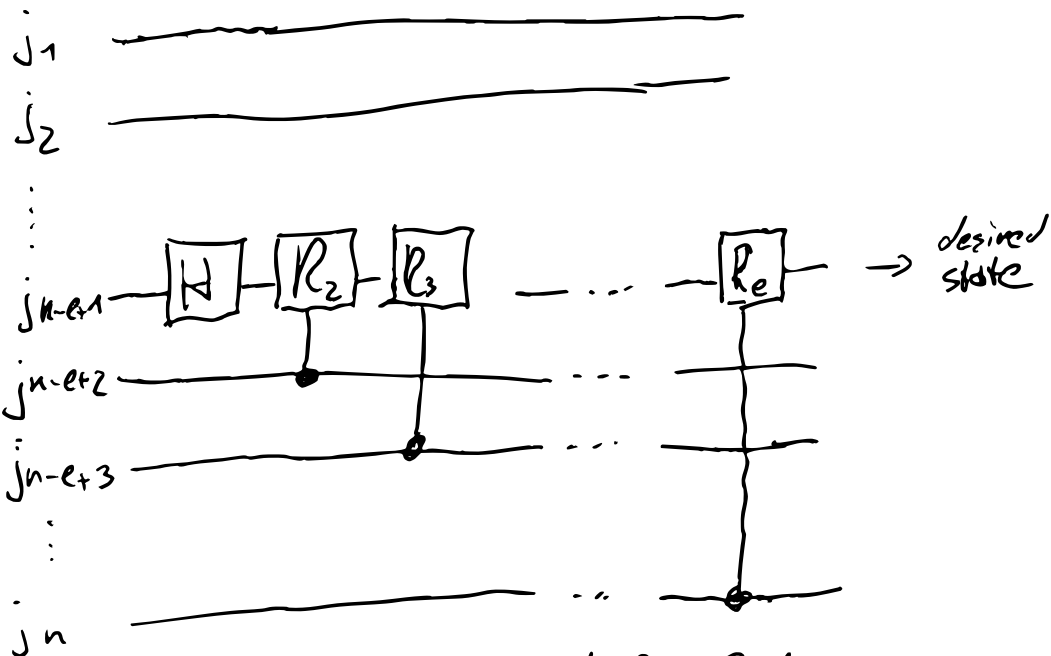
Furthermore note that

$$e^{2\pi i j/2^e} = e^{2\pi i \sum_{m=1}^{n-e} j_m 2^{n-m-e}} \cdot e^{2\pi i \sum_{m=n-e+1}^{n} j_m 2^{n-m-e}}$$

$= 1$ since the exponent $\in \mathbb{N}$

$\Rightarrow$ the first $n-e$ significant bits of j do not matter.

To implement $\frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i \sum_{m=n-e+1}^{n} j_m 2^{n-m-e}} \cdot |1\rangle\right)$ we use the R_s gate or P in IBM q-Composer given by the matrix $\begin{pmatrix} 1 & 0 \\ 0 & e^{\frac{2\pi i}{2^s}} \end{pmatrix}$



need to repeat that for $e = 1, \dots, n$

Cost of QFT:

we need n qbits and $O(n)$ gates per qbit

$\Rightarrow O(n^2)$ gates

$\Rightarrow$ exponential speedup over FFT with $O(n2^n)$ operations.

Remark Strictly speaking, QFT does something different than FT. The state

$$QFT(|x\rangle) = \sum_{j=0}^{N-1} x_j |j\rangle$$

can only be accessed via measurement and hence will collapse to some $|j'\rangle$ with a certain probability. We will see that this is still very useful.

Remark R_s gates don't do very much for large s. One can show that $O(n \log n)$ gates suffice if one accepts a small error probability

Remark reversing the order of the gates and using the adjoint gates gives an efficient implementation of the inverse QFT F_N^{-1}

Application Phase estimation

Suppose we have unitary operator $U : V \to V$, $\dim V = 2^n$

with eigenvector ψ

$\Rightarrow U\psi = \underbrace{e^{2\pi i \phi}}_{\lambda} \psi$ for some $\phi \in [0,1)$

$(|U\psi|^2 = \lambda^2 |\psi|^2 = |\psi|^2 \Rightarrow |\lambda| = 1)$

Assume that $\phi = \sum_{j=1}^{n} \phi_j 2^{-j}$ can be written with n bits

Since U is operator on 2^n-dim Hilbertspace classical computation of $U\psi \cdot \psi$ costs at least $O(2^n)$

Quantum algorithm

1) Start with $|0^n\rangle |\psi\rangle$

2) Apply $H^{\otimes n}$ to the first n qbits to obtain

$$\frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} |j\rangle |\psi\rangle$$ (Applying QFT_N would do the same)

3) Apply $|j\rangle|\psi\rangle \mapsto |j\rangle U^j |\psi\rangle$ to obtain

$$\frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{2\pi i j \phi} |j\rangle |\psi\rangle$$

Note that the first n qbits satisfy

$$\frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{2\pi i j \phi} |j\rangle = \frac{1}{\sqrt{N}} \sum e^{2\pi i j \frac{N\phi}{N}} |j\rangle = F_N(|N\phi\rangle)$$

→ This might be a bit confusing since suddenly $\phi \in \mathbb{R}$ is interpreted as an element of the vector space in which ψ is contained

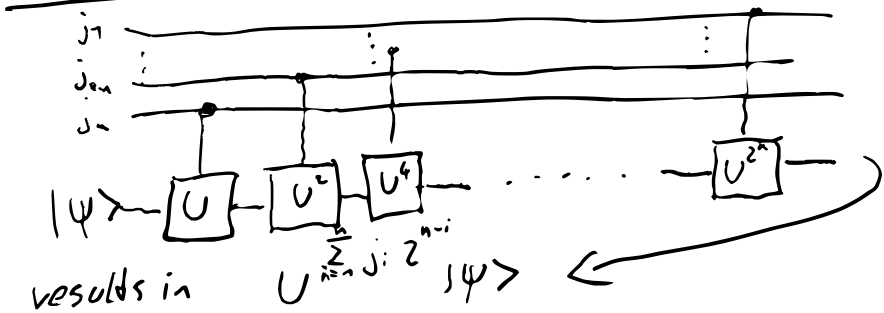
But $N\phi = \sum_{j=1}^{n} \phi_j 2^{n-j}$ is a basis element and hence this makes sense

4) Apply IQFT to the first n qbits to obtain

$$F_N^{-1}\left(\frac{1}{\sqrt{N}} \sum e^{2\pi i j \phi} |j\rangle\right) = |N\phi\rangle$$

5) Measure in computational basis to obtain $N\phi$ and hence ϕ

Remark How to implement step 3 ?



But we will need to assume that

$\boxed{U^{2^k}}$ can be implemented effectively

(This might not be true in general, but is in the applications below)

Note the the input doesn't need to be a single eigenvector. Let $|\psi\rangle, |\psi'\rangle$ denote two EVs with phase ϕ, ϕ'. Linearity implies that input $\frac{1}{\sqrt{2}}(|\psi\rangle + |\psi'\rangle)$ produces output

$\frac{1}{\sqrt{2}}(|N\phi\rangle + |N\phi'\rangle)$

A measurement will produce ϕ, ϕ' will equal probability

---

if ϕ requires more than n bits, the final state is a small perturbation δ of $F_N(|N\phi\rangle)$ with

$$|\delta| = \left| \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \underbrace{\left[e^{2\pi i j\phi} - e^{2\pi i j\tilde{\phi}} \right]}_{O(2^{-n})} |j\rangle \right| \Big/ \left| \frac{1}{\sqrt{N}} \sum e^{2\pi i j\phi} |j\rangle \right|$$

$$\simeq O(2^{-n})$$

Shor's integer factorization

Factoring problem: given $n \in \mathbb{N} \setminus \{\text{primes}\}$
find $1 < k < n$ with $\frac{n}{k} \in \mathbb{N}$.

- $n \in \mathbb{N}$ can be defined using $\log_2(n)$ bits $\Rightarrow$ Rdy. complexity of klps means $\mathcal{O}(\log_2(n)^p)$ operations
- There exist efficient classical algorithms to check whether n is prime or power of prime (see, e.g. AKS-test or BPSW-test)
- Security of many crypto-systems is based on the fact that integer factorization is hard
- Best known classical algorithm is the general number field sieve with runtime $\mathcal{O}\left(e^{(\log_2 n)^{\frac{1}{3}} (\log_2 \log_2 n)^{\frac{2}{3}}}\right)$

Note that the runtime is a conjecture!

- It is not known that classical dlps can not be faster. Latest paper which (falsely) claimed an $O(\log n^p)$-dlp for factorization was by Schnorr (Bonn) in 2021

Reduction to period finding

Want to factor $N \in \mathbb{N}$.

1) Choose random $x \in \{2, \dots, N-1\}$ with $\gcd(x, N) = 1$

$\Rightarrow x \in \left(\mathbb{Z}/N\mathbb{Z}\right)^\times$ multiplicative group mod n

(Lemma 3)

$\Rightarrow x$ has period r with $x^r \bmod N = 1$

$\Rightarrow$ with prob. $\frac{1}{2}$ r is even and

$$x^{\frac{r}{2}} \pm 1 \not\equiv 0 \bmod N$$

$$\Rightarrow (x^{\frac{r}{2}} - 1)(x^{\frac{r}{2}} + 1) \equiv x^r - 1 \equiv 0 \bmod N$$

$\Rightarrow \gcd(x^{\frac{r}{2}} - 1, N)$, $\gcd(x^{\frac{r}{2}} + 1, N)$ are non-trivial factors of N

Lemma Every $x \in \left(\mathbb{Z}/N\mathbb{Z}\right)^\times$ has period r which is minimal $r \in \mathbb{N}$ with $x^r \equiv 1 \bmod N$.

Proof Consider the set $S = \{1, x, \dots\} \subseteq \left(\mathbb{Z}/N\mathbb{Z}\right)^\times$

Since $\left(\mathbb{Z}/N\mathbb{Z}\right)^\times$ is finite, there exist $j \neq k \in \mathbb{N}$ with $x^j = x^k \bmod N$. W.L.O.G assume $j < k$

$\Rightarrow x^{k-j} = 1 \bmod N$. □

Euler totient func: $\varphi(n) := \#\{1 \leq k \leq n : \gcd(k,n) = 1\}$

$\varphi(n) = n \prod_{\substack{p \mid n \\ p \text{ prime}}} (1 - \frac{1}{p})$ $\Rightarrow \varphi(p^\alpha) = p^\alpha(1 - \frac{1}{p}) = p^{\alpha-1}(p-1)$

Lemma [Chinese remainder Thm]

Let $m_1, \dots, m_n \in \mathbb{N}$ with $\gcd(m_i, m_j) = 1$ $i \neq j$. Then

$$\left[\begin{array}{ll} \text{find} & \\ x \text{ s.t.} & x = a_1 \bmod m_1 \\ & \vdots \quad\quad \vdots \\ & x = a_n \bmod m_n \end{array}\right]$$

is solvable and any two solutions are equal mod $M = m_1 m_2 \cdots m_n$

Proof Define $M_i := \frac{M}{m_i} \Rightarrow \gcd(M_i, m_i) = 1$

$\Rightarrow M_i$ has mult. inverse mod m_i denoted by N_i

$\Rightarrow x = \sum_i a_i M_i N_i$ is solution since

$$M_i N_i \equiv 1 \bmod m_i \text{ and } M_i N_i \equiv 0 \bmod m_j \quad j \neq i$$

Two solutions x, x' satisfy $x - x' \equiv 0 \bmod m_i$

$\Rightarrow M = m_1 m_2 \cdots m_n$ divides $x - x'$ (since m_i are coprime) □

Lemma 4 Let $p > 2$ prime. Let 2^d maximal power of 2 dividing $\varphi(p^\alpha)$. Let x denote randomly chosen element in $\left(\frac{\mathbb{Z}}{p^\alpha \mathbb{Z}}\right)^*$.

$$\Rightarrow \quad P(2^d \text{ divides order of } x) = \frac{1}{2}$$

Proof: $\varphi(p^\alpha) = p^{\alpha-1}(p-1)$ is even $\Rightarrow d \geq 1$.

Let g denote generator of $\left(\frac{\mathbb{Z}}{p^\alpha \mathbb{Z}}\right)^* \Rightarrow x = g^k \bmod p^\alpha$ for some $k \in \{1, \dots, \varphi(p^\alpha)\}$. Let r denote order of x.

1) k is odd: $g^{kr} \equiv 1 \bmod p^\alpha \Rightarrow \varphi(p^\alpha)$ divides kr

since k is odd $\Rightarrow 2^d$ divides r [since $\varphi(p^\alpha)$ is minimal with $g^{\varphi(p^\alpha)} = 1$]

2) k is even: $g^{k\varphi(p^\alpha)/2} = 1 \bmod p^\alpha$

$\Rightarrow r$ divides $\varphi(p^\alpha)/2$ since [$x^r = g^{kr} = 1 \bmod p^\alpha$ — r is minimal with]

$\Rightarrow 2^d$ does not divide r

Hence, for exactly half of $x \in \left(\frac{\mathbb{Z}}{p^\alpha \mathbb{Z}}\right)^*$ we have

$x = g^k$ with k even/odd $\square$

Lemma 5 Let $N = p_1^{\alpha_1} \cdots p_m^{\alpha_m}$ prime factorization of odd N. Let x be random in $\left(\frac{\mathbb{Z}}{N\mathbb{Z}}\right)^*$ with order r mod N.

$$\Rightarrow \mathbb{P}\left(r \text{ is even and } x^{\frac{r}{2}} \not\equiv -1 \bmod N\right) \geq 1 - 2^{-m+1}$$

Proof Chinese remainder theorem $\Rightarrow$ choose x random is equivalent to choose x_j rand in $\left(\frac{\mathbb{Z}}{p_j^{\alpha_j}\mathbb{Z}}\right)^*$ with $x = x_j \bmod p_j^{\alpha_j}$ $\forall j = 1, \ldots, m$

{ This is because $x \leftrightarrow (x_1, \ldots, x_m)$ is one-to-one)

Let r_j be order x_j mod $p_j^{\alpha_j}$ and let 2^{d_j} max power of 2 dividing r_j. (r is odd or $x^{\frac{r}{2}} \equiv -1 \bmod N \Rightarrow$ We will show $d = d_1 = \ldots = d_m$. If this is the case, the following argument concludes the proof:

Let d_j' maximal s.t. $2^{d_j'}$ divides $\varphi(p_j^{\alpha_j})$

Lemma 4 $\Rightarrow \mathbb{P}(d_j = d_j') = \frac{1}{2} \Rightarrow \mathbb{P}(d_j = k) \leq \frac{1}{2}$

$\forall k \in \mathbb{N}$. Since d_j independent

$$\Rightarrow \mathbb{P}(d_1 = \ldots = d_m) \leq 2^{-m+1}$$

- It remains to show $d_1 = \ldots = d_m$.

Case 1: r is odd. $x^r = 1 \bmod N \Rightarrow x^r = 1 \bmod p_j^{\alpha_j}$

$\Rightarrow r_j$ divides r $\Rightarrow r_j$ is odd $\Rightarrow d_j = 0$.

Case 2: r is even and $x^{\frac{r}{2}} = -1 \bmod N$

$\Rightarrow x^{\frac{r}{2}} = -1 \bmod p_j^{\alpha_j}$. If r_j would divide $\frac{r}{2}$, we would have

$$\left.\begin{aligned} x^{\frac{r}{2}} = \underbrace{x^{k r_j}} &= -1 \bmod p_j^{\alpha_j} \\ &= x_j^{k r_j} \bmod p_j^{\alpha_j} \\ &= 1 \bmod p_j^{\alpha_j} \end{aligned}\right\} \Rightarrow r_j \text{ does not divide } \frac{r}{2}.$$

but r_j divides r. Hence largest power of 2 dividing r must be equal to d_j. □

Note that if r is period of $x \bmod N$ and r even, we have $x^{\frac{r}{2}} \neq 1 \bmod N$

$\Rightarrow P(r \text{ even } x^{\frac{r}{2}} \neq \pm 1 \bmod N) \geq 1 - 2^{-m+1} \geq \frac{1}{2}$

if $m \geq 2$, i.e. if N is not a prime power.

Define $U|y\rangle := |x\,y \bmod N\rangle$

Note that $|j\rangle \longmapsto |x^j \bmod N\rangle$ is bijective

since $x^j = x^{j'} \bmod N \quad x^{(j-j')} = 0 \bmod N$

$\Rightarrow \gcd(x,N) \neq 1.$

$\Rightarrow U$ is permutation on basis elements $\Rightarrow$ unitary

Lemma Let r denote the period of x in $\left(\mathbb{Z}/N\mathbb{Z}\right)^*$.

Then, $|v_s\rangle := \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} \exp\left(\frac{-2\pi i s k}{r}\right) |x^k \bmod N\rangle$

are EVs of U with eigenvalues

$\lambda_s := \exp\left(\frac{2\pi i s}{r}\right), \quad s = 0, \ldots, r-1$

Proof

$$U|v_s\rangle = \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} \exp\left(\frac{-2\pi i s k}{r}\right) |x^{k+1} \bmod N\rangle$$

$$= \lambda_s \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} \exp\left(\frac{-2\pi i s (k+1)}{r}\right) |x^{k+1} \bmod N\rangle$$

$x^r = x^0 \bmod N$

$e^{-2\pi i s} = e^0$

$$= \lambda_s |v_s\rangle$$

□

Note that there holds

$$\frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} |u_s\rangle = \frac{1}{r} \sum_{s=0}^{r-1} \sum_{k=0}^{r-1} e^{\frac{-2\pi i k s}{r}} |x^k \bmod N\rangle$$

$$= \frac{1}{r} \sum_{k=0}^{r-1} \underbrace{\sum_{s=0}^{r-1} e^{\frac{-2\pi i k s}{r}}}_{\hookrightarrow \begin{cases} 0 & k \neq 0 \\ r & k = 0 \end{cases}} |x^k \bmod N\rangle$$

$$= |x^0 \bmod N\rangle = |1\rangle$$

$\Rightarrow$ Quantum phase estimation for U with input $|1\rangle$ produces the state

$$\frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} |2^n \tfrac{s}{r}\rangle$$

note that $r \leq N$ hence $\frac{s}{r}$ can be exactly represented with n bits

- To efficiently implement $U^{2^k}|y\rangle = |x^{2^k} y \bmod N\rangle$ we use the fact $y^{2^k} = \underbrace{\left(\left(\left(y^2\right)^2\right)\cdots\right)^2}_{k\text{-times}}$. Hence $|xy^{2^k} \bmod N\rangle$ requires 1 multiplication, k squares mod N

$\Rightarrow$ can be implemented as in classical circuits with XOR & AND gates.

Shor's period finding algorithm

Note that this is the only quantum part of Shor's alg. $N \leq 2^n$

1) Prepare $|\psi\rangle = |1\rangle = |\overbrace{0\cdots01}^{n}\rangle$ and U as before

2) Use quantum phase estimation with U and $|\psi\rangle$ to obtain the state

$$\frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} |2^n \tfrac{s}{r}\rangle$$

3) Measurement gives a random number $\frac{s}{r}$ from the set $\{\frac{1}{r}, \dots, \frac{r-1}{r}\} \subseteq \{\frac{i}{2^n} : i = 0, \dots, 2^n - 1\}$

4) Write $\frac{s}{r} = \frac{s_0}{r_0}$ with $\gcd(s_0, r_0) = 1$. If s, r were coprime already, we have $r = r_0$ and found the period. Otherwise, repeat.

Remark: There are $O(\frac{r}{\varphi\varphi r})$ numbers $1 \leq s < r$ with $\gcd(s, r) = 1$. — Euler's totient function

$\Rightarrow$ Success prob of step 4 is $O(\frac{1}{\varphi\varphi r})$

Since $\frac{1}{\varphi\varphi r} \geq \frac{1}{\varphi\varphi N} \geq \frac{1}{\varphi n}$, step 4 requires on average φn trys.

Remark

Implicitely, we used QFT to find the frequency of

$x^k \bmod N$, i.e. the period of x

Grover's algorithm

Problem: Given $F : \{0,1\}^n \to \{0,1\}$, find $x \in \{0,1\}^n$ with $F(x) = 1$ or determine $F = 0$.

Recall the query

$$Q_F(|i\rangle) = (-1)^{F(i)} |i\rangle$$

(algorithm we identify $x \in \{0,1\}^n$ with $i = 0, \dots, 2^n - 1$)

Define the Grover diffusion operator

$$U_S := 2|S\rangle\langle S| - I,$$

($|S\rangle\langle S|$: projection onto $|S\rangle$)

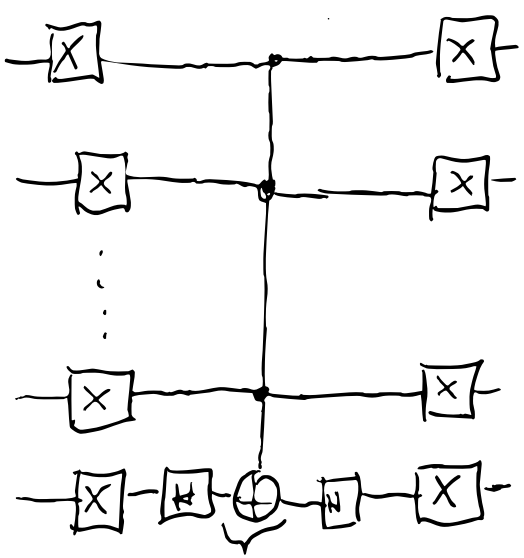
where $|S\rangle$ denotes the uniform state

$$|S\rangle = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} |j\rangle$$

Note that $|S\rangle = H^{\otimes n} |0^n\rangle$ and hence

$$U_S = H^{\otimes n} \underbrace{\left(2|0^n\rangle\langle 0^n| - I\right)}_{\text{corresponds to the matrix}} H^{\otimes n}$$

$$\begin{pmatrix} +1 & & 0 \\ & -1 & \\ & & \ddots \\ 0 & & & -1 \end{pmatrix} \in \mathbb{R}^{2^n \times 2^n}$$



$$-\left(2|0\rangle\langle 0| - \mathbb{I}\right)$$

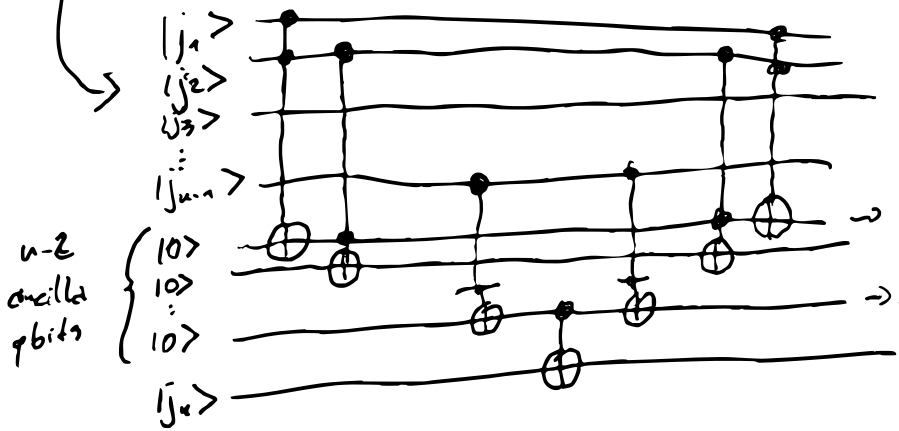
since we can only measure amplitudes, a global phase change (multiplication with $\lambda \in \mathbb{C}, |\lambda| = 1$) is not noticable

alternatively, one can apply

to obtain

$$\begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}$$

generalized Toffoli gate



Algorithm

1) Apply $H^{\otimes n}$ to obtain $|s\rangle = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} |j\rangle$

2) For $k = 1, \dots, r(N)$ do

2a) Apply Q_f to current state $\sum_{j=0}^{N-1} \alpha_j |j\rangle$ to obtain $\sum_{j=0}^{N-1} (-1)^{f(j)} \alpha_j |j\rangle$

2b) Apply U_s to first n qbits

3) Measure in computational basis

Thm: With $r(N) = \frac{\arccos(\sqrt{\frac{t}{N}})}{2 \arcsin(\sqrt{\frac{t}{N}})}$, Grover's alg. finds state $|x\rangle$ with $f(x)=1$ with probability $\geq 1 - \frac{t}{N}$.

Proof:

Define $|B\rangle := \frac{1}{\sqrt{N-t}} \sum_{f(j)=0} |j\rangle$, where $t = \#\, f^{-1}(\{1\})$

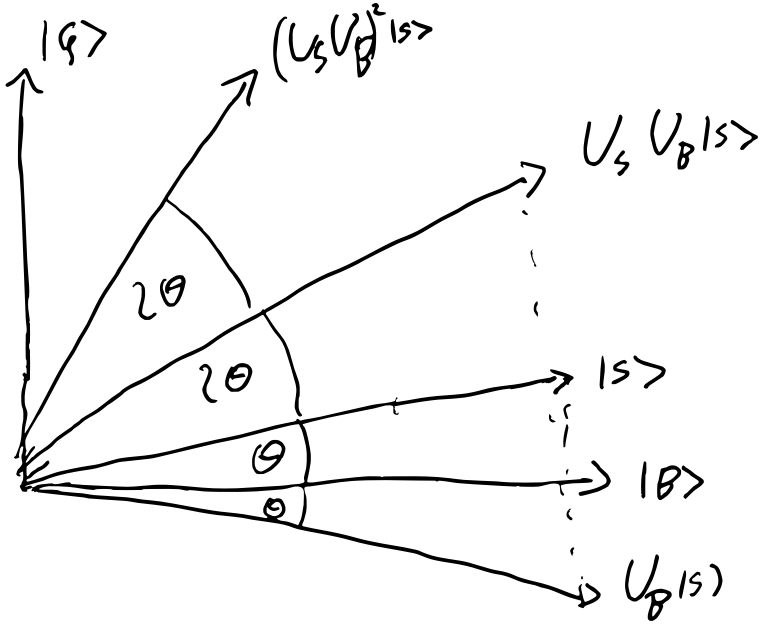
Note that the first n qbits of $|x\rangle \mapsto Q_f(|x\rangle)$ satisfy $|x\rangle \mapsto (2|B\rangle\langle B| - I)|x\rangle = U_B(|x\rangle)$

(Check for basis elements $|x\rangle = |j\rangle$)

Define $|g\rangle := \frac{1}{\sqrt{t}} \sum_{f(j)=1} |j\rangle$ and note

$|s\rangle \in \operatorname{span}\{|g\rangle, |B\rangle\}$. ; $U_s, U_B : \operatorname{span}\{|g\rangle, |B\rangle\} \leftarrow$

hence, the Grover iteration never leaves the plane $\operatorname{span}\{|g\rangle, |B\rangle\}$



Each application of $U_s U_B$ rotates a state $|x\rangle \in \text{span}\{|B\rangle, |\phi\rangle\}$ towards $|\phi\rangle$ by an angle θ given by

$$\cos\theta = \frac{\langle s|B\rangle}{|s||B|} = \frac{1}{\sqrt{N}} \frac{1}{\sqrt{N-t}} \sum_{f(|j\rangle)=0} \langle j|j\rangle = \frac{N-t}{\sqrt{N(N-t)}}$$

$$= \sqrt{1 - \frac{t}{N}}$$

$$\Rightarrow \sin^2\theta = 1 - \cos^2\theta = 1 - 1 + \frac{t}{N} \Rightarrow \sin\theta = \sqrt{\frac{t}{N}}$$

$$\Rightarrow \theta \approx \sqrt{\frac{t}{N}} \quad \text{for large } N.$$

Since the angle between $|\phi\rangle$ and $|s\rangle$ is

$$\cos\alpha = \frac{1}{\sqrt{N}} \frac{1}{\sqrt{t}} \sum_{f(|j\rangle)=1} \langle j|j\rangle = \frac{\sqrt{t}}{\sqrt{N}} \Rightarrow r(N) = \text{round}\left(\frac{\arccos\left(\frac{\sqrt{t}}{\sqrt{N}}\right)}{2\arcsin\frac{\sqrt{t}}{\sqrt{N}}}\right)$$

to obtain a state $|x\rangle$ with $\angle |g|x\rangle \geq \cos(\theta)$

$$\Rightarrow |\langle g|x\rangle|^2 \geq \cos\theta^2 = 1 - \frac{t}{N}$$

□

Remark in practise, we don't know when $|x\rangle$ gets close to $|g\rangle$. But we can measure and check whether $F(x) = 1$. If not, restart the algorithm. If $t \ll N$

$\Rightarrow$

$$r(N) = \frac{\arccos\sqrt{\frac{t}{N}}}{\arcsin\sqrt{\frac{t}{N}}} \approx \frac{\pi}{4}\sqrt{\frac{N}{t}}$$

Remark If $t = 1$, a classical algorithm requires at least N evaluations of F. Grovers algorithm only needs $O(\sqrt{N})$ iterations with $O(n)$ gates

Optimality of Grover's Alg

Lemma Any Quantum Alg based on the Query Q_f requires at least $O(\delta\sqrt{N})$ applications of Q_f to succeed with prob. $\geq \delta^2$.

Proof Any Q-Alg starts with some state $|\psi\rangle$ and applies unitary transformations as well as Q_f. I.e. we may write the state after k applications of Q_f as

$$|\psi_k^f\rangle := \underbrace{U_k\, Q_f\, U_{k-1}\, Q_f \cdots U_1\, Q_f}_{\text{unitary}}\, |\psi\rangle$$

Additionally, we will consider

$$|\psi_k\rangle := U_k\, U_{k-1} \cdots U_1\, |\psi\rangle$$

Let $f_j : \{0,1\}^n \to \{0,1\}$ with $f_j(|i\rangle) = \delta_{ij}$

and define

$$D_k = \sum_{j=0}^{N-1} \| \psi_k^{f_j} - \psi_k \|^2$$

Idea: If D_k is small, the evaluation of $\overline{F}$ doesn't make a big difference and it will be hard to find $\overline{F}(|i\rangle) = 1$.

Step 1: Show that $D_k \leq 4k^2$ by induction.

$k=0$: $D_k = 0$ ✓

$k \to k+1$:

$$D_{k+1} = \sum_{j=0}^{N-1} \| U_{k+1} Q_{\overline{F}_j} \psi_k^{\overline{F}_j} - U_{k+1} \psi_k \|^2$$

$$= \sum_{j=0}^{N-1} \| Q_{\overline{F}_j} \psi_k^{\overline{F}_j} - \psi_k \|^2$$

$$= \sum \| Q_{\overline{F}_j} (\psi_k^{\overline{F}_j} - \psi_k) + (Q_{\overline{F}_j} - I) \psi_k \|^2$$

Note that $Q_{\overline{F}_j} = I - 2|j\rangle\langle j| \Rightarrow (Q_{\overline{F}_j} - I)|\psi_k\rangle$

$$= -2|j\rangle(\langle j|\psi_k\rangle)$$

$\Rightarrow$

$$D_{k+1} \leq \sum_{j=0}^{N-1} \| \psi_k^{\overline{F}_j} - \psi_k \|^2 + 4 \| \psi_k^{\overline{F}_j} - \psi_k \| |\langle j|\psi_k\rangle| + 4 |\langle j|\psi_k\rangle|^2$$

Cauchy - Schwartz shows

$$D_{k+1} \le D_k + 4\Big(\sum_j \|\psi_k^{T_j} - \psi_k\|^2\Big)^{\frac{1}{2}} \underbrace{\Big(\sum_j |\langle j|\psi_k\rangle|^2\Big)^{\frac{1}{2}}}_{=1} + 4\underbrace{\langle\psi_k|\psi_n\rangle^2}_{=1}$$

$$\le D_k + 4\sqrt{D_k} + 4$$

$$\overset{\text{Induction Hyp.}}{\longrightarrow} \le 4k^2 + 8k + 4 = \underline{4(k+1)^2} \quad \checkmark$$

This concludes the induction and shows

$$\underline{D_k \le 4k^2}$$

Step 2: Assume $|\langle j|\psi_k^{T_j}\rangle|^2 \ge \delta^2 > 0 \ \forall j = 0,\dots,N-1$, i.e. the Alg. works with high prob. for each input.

Replacing $|j\rangle$ with $e^{i\theta_j}|j\rangle$ does not change success prob. $\Rightarrow$ we may assume $|\langle j, \psi_k^{T_j}\rangle| = \langle j, \psi_k^{T_j}\rangle$.

$$\Rightarrow \|\psi_k^{T_j} - j\|^2 = 2 - 2|\langle j, \psi_k^{T_j}\rangle| \le \underline{2(1-\delta)}$$

Define $E_k := \sum_{j=0}^{N-1} \|\psi_k^{T_j} - j\|^2 \Rightarrow E_k \le 2N(1-\delta)$

$$F_k := \sum_{j=0}^{N-1} \|j - \psi_k\|^2$$

$\Rightarrow$

$$D_k = \sum_{j=0}^{N-1} \| (\psi_k^{T_j} - j) + (j - \psi_k) \|^2$$

$$\geq \sum_j \| \psi_k^{T_j} - j \|^2 - 2 \| \psi_k^{T_j} - j \| \| j - \psi_k \| - \| j - \psi_k \|^2$$

$$= E_k + F_k - 2 \left(\sum_j \| \psi_k^{T_j} - j \|^2 \right)^{\frac{1}{2}} \left(\sum_j \| j - \psi_k \|^2 \right)^{\frac{1}{2}}$$

$$= E_k + F_k - 2\sqrt{E_k F_k} = \underline{\left(\sqrt{E_k} - \sqrt{F_k} \right)^2}$$

Note that any state $|\phi\rangle$ satisfies

$$\sum_{j=0}^{N-1} \| \phi - j \|^2 = \sum_j \underbrace{\|\phi\|^2}_{=1} - 2\langle \phi | j \rangle + \underbrace{\|j\|^2}_{=1}$$

$$\geq 2N - 2\sqrt{\sum_j 1} \underbrace{\sqrt{\sum_j |\langle \phi | j \rangle|^2}}_{=1}$$

$$= \underline{2(N - \sqrt{N})}$$

This implies $F_k \geq 2(N - \sqrt{N})$ and hence

$F_k \geq E_k$ for sufficiently large N and

$$\sqrt{F_k} - \sqrt{E_k} \geq \sqrt{2(N-\sqrt{N})} - \sqrt{2(1-\delta)N} = \frac{2\delta N - 2\sqrt{N}}{\sqrt{2(N-\sqrt{N})} + \sqrt{2(1-\delta)N}}$$

$$\geq \frac{\pi\delta N - \pi\sqrt{N}}{\pi\sqrt{2N}} \doteq \frac{\delta}{\sqrt{2}}\sqrt{N} - \frac{\hat{1}}{\sqrt{2}}$$

$$\Rightarrow D_k \geq \left(\sqrt{F_k} - \sqrt{E_n}\right)^2 \simeq \delta^2 N$$

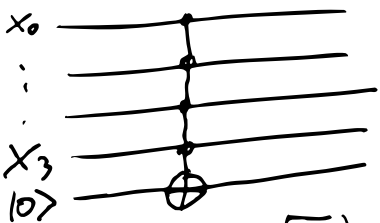
Since $D_k \leq 4k^2 \Rightarrow k \simeq \delta\sqrt{N}$ □

Example in Quantum Composer

Link on webpage, $N = 16$, $n = 4$

$$f(x) = f(x_0, \ldots, x_3) = x_0 \text{ AND } x_1 \text{ AND } x_2 \text{ AND } x_3$$
$$= \delta_{x=(1,1,1,1)} \Rightarrow t = 1$$

$f \longleftrightarrow$



There holds: $\frac{\arccos\left(\sqrt{\frac{1}{16}}\right)}{2\arcsin\left(\sqrt{\frac{1}{16}}\right)} \simeq 2.6083$

$\Rightarrow$ optimal $r(N) = 3$.

Probability of success $\geq 1 - \frac{1}{N} = \frac{15}{16} \simeq 0.9375$

Numerical Quadrature:

Problem: Given $f : \{0,1\}^n \to [-1,1]$, compute $\frac{1}{N}\sum_{i=0}^{N-1} f(i)$

Quantum Super Sampling

Assumption: Oracle $Q_f : |0\rangle \otimes |i\rangle \mapsto \left[\sqrt{1-f(i)}\,|0\rangle + \sqrt{f(i)}\,|1\rangle\right] \otimes |i\rangle$

1) Start with $|0^p\rangle \otimes |0\rangle \otimes |0^n\rangle$

2) Apply $H^{\otimes p} \otimes I \otimes H^{\otimes n} \to$

$$\frac{1}{\sqrt{PN}} \sum_{i=0}^{N-1} \sum_{m=0}^{P-1} |m\rangle \otimes |0\rangle \otimes |i\rangle$$

3) Apply $Q_f \to$

$$\frac{1}{\sqrt{PN}} \sum_i \sum_m |m\rangle \otimes \left[\sqrt{1-f(i)}\,|0\rangle + \sqrt{f(i)}\,|1\rangle\right] \otimes |i\rangle$$

4) Define $|G\rangle := \sqrt{\frac{\sum f(i)}{N}} \sum_i \sqrt{f(i)}\,|1\rangle \otimes |i\rangle$

$$|B\rangle := \sqrt{\frac{\sum 1-f(i)}{N}} \sum_i \sqrt{1-f(i)}\,|0\rangle \otimes |i\rangle$$

$$\bar{f} := \frac{1}{N} \sum f(i)$$

Note

$$U_s := (2|0\rangle\langle 0| - I) \otimes I$$

$$U_s|B\rangle = |B\rangle$$

$$U_s|G\rangle = -|G\rangle$$

$$U_\psi = (2|\psi\rangle\langle\psi| - I)$$

with $|\psi\rangle := \sqrt{1-p}\,|B\rangle + \sqrt{p}\,|G\rangle$

Steps 1 - 3 provide quantum circuit to implement the map $U: |0\rangle \otimes |0^n\rangle \longmapsto |\psi\rangle$

$$\Rightarrow U_\psi = U^{-1}(2|0^{n+1}\rangle\langle 0^{n+1}| - I)\,U$$

as in Grovers Algorithm.

5) State reads

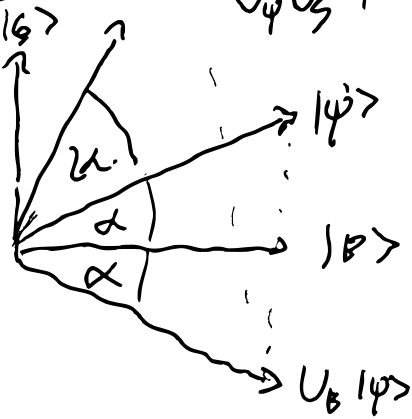
$$\frac{1}{\sqrt{P}} \sum_{m=0}^{P-1} |m\rangle \otimes \left[\sqrt{1-\bar{f}}\,|B\rangle + \sqrt{\bar{f}}\,|G\rangle\right]$$

for $\bar{f} := \frac{1}{N}\sum f(i)$

6) Apply Grover iteration $U_\psi U_S$ m-times to second $n+1$ qbits to obtain

$$\frac{1}{\sqrt{P}} \sum_{m=0}^{P-1} |m\rangle \otimes (U_\psi U_S)^m \left[\sqrt{1-\bar{f}}\,|B\rangle + \sqrt{\bar{f}}\,|G\rangle\right]$$

Explanation: $U_\psi U_S |\psi\rangle$



Let $\sin\theta = \sqrt{\bar{f}}$

$\Rightarrow |\psi\rangle = \cos\theta\,|B\rangle + \sin\theta\,|G\rangle$

$(U_\psi U_S)^m |\psi\rangle = \cos((2m+1)\theta\,|B\rangle + \sin((2m+1)\theta\,|G\rangle$

U_S acts like reflection across $|B\rangle$

Note: m-times application of $U_\psi U_B$ as in „phase estimation" - blg. But: Cost in general $O(P_m)$

7) $|\psi\rangle$ rotates with rate 2θ in $|B\rangle - |g\rangle$ plane. Use QFT to obtain rate.

Measure last $n+1$ qbits in $\{|g\rangle, |B\rangle\}$-Basis to obtain

$$\frac{1}{\sqrt{P}} \sum_{m=0}^{P-1} |m\rangle \otimes \Big[\cos((2m+1)\theta\,|B\rangle + \sin((2m+1)\theta\,|g\rangle \Big]$$

$$\frac{1}{C} \sum_{m=0}^{P-1} \underbrace{\sin\big[(2m+1)\theta\big]}_{x_m} |m\rangle \quad \left(\text{or } \cos(2m+1)\theta\right)$$

where $C = \frac{1}{P} \sum_{m=0}^{P-1} \sin\big[(2m+1)\theta\big]^2$.

→ Apply QFT to obtain [assume $\theta = \frac{\pi \theta_0}{P}$, $\theta_0 \in \mathbb{N}$]

$$\frac{1}{C} \sum_{m=0}^{P-1} x_m |m\rangle \text{, where } x_m \text{ is given}$$

by IFT of $\hat{x}_m$, i.e.

$$x_m = \frac{1}{\sqrt{P}} \sum_{k=0}^{P-1} \hat{x}_k \, e^{\frac{2k\pi i m}{P}}$$

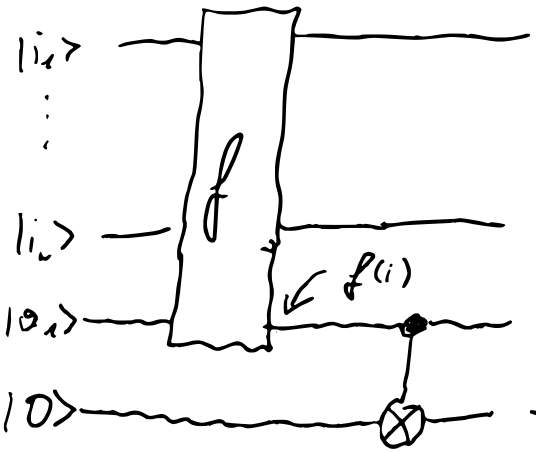
$$= \frac{1}{\sqrt{P}} \sum_{k=0}^{P-1} \frac{1}{2i} \left(e^{i(2k+1)\frac{\theta_0}{P}} - e^{-i(2k+1)\frac{\theta_0}{P}} \right) e^{\frac{2k\pi i m}{P}}$$

$$= e^{i\frac{\theta_0}{P}} \frac{1}{2i} \begin{cases} \sqrt{P} & m = \theta_0 \\ 0 & \text{else} \end{cases}$$

8) Final measurement produces $|m\rangle = |\theta_0\rangle$ and hence $\tilde{f} = \sin\left(\frac{\pi\theta_0}{p}\right)^2$

Remark: implementation of Q_f unclear.

If $f: \{0,1\}^n \to \{0,1\}$, we can use controlled Not-gate to implement



General integrals can always be split into

$$\frac{1}{N}\sum_{i=0}^{N-1} f(i) = \frac{1}{N}\sum_{i=0}^{N-1}\sum_{k=1}^{R} 2^{-k} f_k(i), \quad f_k(i) \in \{0,1\}$$

, i.e., R integration problems for precision 2^{-R}.

Lineare Gleichungssysteme am Quantencomputer

Sei $A \in \mathbb{C}^{N \times N}$, $b \in \mathbb{C}^N$, $N = 2^n$

(invertierbar, hermitesch (komplex symm.))

LGS: finde $x \in \mathbb{C}^N$ s.t. $Ax = b$

für A par.def., sparse $\rightsquigarrow$ klass. Verfahren Standard CG

Aufwand $\mathcal{O}(sN \cdot \ell)$ s .. sparsity, ℓ .. Anzahl Iterationen

Fehler $\|x - x_\ell\|_A \leq 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^\ell \|x - x_0\|_A$

$\kappa = \kappa(A)$ .. Konditionszahl $= \|A\| \cdot \|A^{-1}\|$

für rel. Fehler $\leq \varepsilon$

$$\Rightarrow 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^\ell \leq \varepsilon$$

$$q := \left(1 - \frac{2}{\sqrt{\kappa}+1}\right) \geq e^{-\frac{2}{\sqrt{\kappa}}}$$

$$\Rightarrow \ell \leq \frac{\log \frac{\varepsilon}{2}}{\log q} \leq \frac{1}{2}\sqrt{\kappa} \ln\left(\frac{2}{\varepsilon}\right)$$

$\rightsquigarrow$ Aufwand $\mathcal{O}(sN\sqrt{\kappa} \ln \frac{2}{\varepsilon})$

Q: exponentieller Speedup auf Q-Comp. möglich??

Quantenversion von LGS:

QLGS: $A \in \mathbb{C}^{N \times N}$ hermitesch, $\det A = 1$

$(b_i) = b \in \mathbb{C}^N$, $x \overset{(x_i)}{\in} \mathbb{C}^N$ s.t. $x = A^{-1}b$

gegeben, $N = 2^n$.

Sei weiters $|b\rangle$ ein n-qubit Zustand gegeben als

$$|b\rangle := \sum_i b_i |i\rangle / \| \sum_i b_i |i\rangle \|$$

und

$$|x\rangle := \sum_i x_i |i\rangle / \| \sum_i x_i |i\rangle \|$$

Ziel: finde Zustand $|\tilde{x}\rangle$ s.d.

$\| |x\rangle - |\tilde{x}\rangle \| \leq \varepsilon$ ← resp. Fehlertoleranz

mit Wahrsch. $\Omega \geq \frac{1}{2}$

Schreibweise in Literatur (formal!)

$A|x\rangle = |b\rangle$

Bem.:) Normalisierungen notwendig
da sonst kein Q-Zustand!

.) QLSP ≠ LSP
↳ man erhält nur Zustand $|x\rangle$, nicht
Vektor, der $Ax=b$ löst!

.) Falls A nicht hermitsch betrachte

$$\tilde{A} = \begin{pmatrix} 0 & A^H \\ A & 0 \end{pmatrix} = |1\rangle\langle 0| \otimes A + |0\rangle\langle 1| \otimes A^H$$

⇝ Matrix dim. verdoppelt
≃ 1 zusätzliches Ancilla Qubit

löse $\tilde{A}|0x\rangle = |1b\rangle$

.) $\det A = 1$ keine echte Einschränkung
⇝ schiefe Matrix

.) wollen: effizienten Algor.
↳ poly logarithmisch in N

.) impl. Annahme: Zustand $|b\rangle$ kann
effizient bereitgestellt werden
A kann am Q-Comp. implementiert werden
↳ später!

Anm. Auslesen von Koeff. von $|x\rangle \rightsquigarrow \mathcal{O}(N)$

$\rightarrow$ QLSP nur nützlich wenn Quantenzust. benötigt

$\rightsquigarrow$ Subroutine für andere Probleme

HHL (Harrow, Hassidim, Lloyd) - Algorithmus

Idee: A hermitsch $\rightarrow$ Spektralsatz:

$$A = \sum_{j=0}^{N-1} \lambda_j |u_j\rangle\langle u_j|$$

$\lambda_j \in \mathbb{R}$ -- EW $|u_j\rangle \in \mathbb{R}^N$ -- EV

$$|b\rangle = \sum_{j=0}^{N-1} \beta_j |u_j\rangle = \sum_{j=0}^{N-1} \langle u_j|b\rangle |u_j\rangle$$

Spektralsatz (oder direkt nachrechnen):

$$A^{-1} = \sum_{j=0}^{N-1} \frac{1}{\lambda_j} |u_j\rangle\langle u_j|$$

$$\Rightarrow |x\rangle = A^{-1}|b\rangle = \sum_{j=0}^{N-1} \frac{\beta_j}{\lambda_j} |u_j\rangle$$

nur formal zu verstehen $\rightsquigarrow$ als korrekte Quantenoperation später!

Problem: A, A^{-1} i.A. nicht unitär
$\Rightarrow A^{-1}|b\rangle$ nicht erlaubt

Lösung: Matrix $U = e^{iA}$ unitär,
hat selbe EV wie A da

$$A = XDX^{-1} \Rightarrow e^{iA} = \sum_{n=0}^{\infty} \frac{(iA)^n}{n!}$$

$$= \sum_{n=0}^{\infty} \frac{(iXDX^{-1})^n}{n!}$$

$$= X \sum_{n=0}^{\infty} \frac{(iD)^n}{n!} X^{-1}$$

$$= X e^{iD} X^{-1}$$

$$= \sum_j e^{i\lambda_j} |v_j\rangle\langle v_j|$$

---

1) berechne EW von U
   $\Rightarrow$ EW von A
   mittels Quantum Phase Estimation
2) Invertiere EW $\lambda_j \mapsto \frac{1}{\lambda_j}$
   wie ?? $\rightsquigarrow$ „controlled rotation"
3) Reversiere QPE

Hamiltonian Simulation

direkt $O(N^3)$ ↯

Q: Wie kann man $e^{iAt} = U$ bzw. U^{2^k} effizient implementieren?

hier: A ist Hamiltonian für U

obige Frage ist eine fundamentale Fragestellung in Quanten computing!

Schrödinger glg.s

$$i\frac{d}{dt}|\psi(t)\rangle = H|\psi(t)\rangle$$

beschreiben jedes Q-System

$$\hookrightarrow |\psi(t)\rangle = e^{-iHt}|\psi(0)\rangle$$

zur eff. Simulation $\rightsquigarrow$ muss e^{-iHt} eff. impl.!

„Hamiltonian Simulation"

als circuit bis auf Fehler $\leq \varepsilon$

Definition Ein Hamiltonian H, der auf n qubits agiert kann effizient simuliert werden, wenn

$\forall t > 0, \varepsilon > 0$ $\exists$ q-circuit U_H bestehend aus $poly(n, t, \frac{1}{\varepsilon})$ gates s.d.

$$\|U_H - e^{-iHt}\| < \varepsilon$$

Bem. Zeitabh. wichtig
„no-fast forwarding theorem"
generell: min. Zeit zur Simulation von H an t
$\rightsquigarrow \Theta(t)$

---

generelles approx. Problem $\rightsquigarrow$ NP schwer Gate Zerl. zu finden

$\rightarrow$ brauchen Annahmen

Ann. 1: $H = \sum_{j=1}^{m} H_j$ $\quad m \sim \text{poly}(n)$

& alle H_j agieren auf $k = \Theta(1)$ qubits

„k-local Hamiltonians"

Trotter-Suzuki splitting

Für k-local Ham. Implementierung von

$e^{iH_j t}$ einfacher/effizient ob H_j nur auf k qubits agiert

falls z.B. H diagon. $\Rightarrow H = T D T^H$

$\Rightarrow e^{iHt} = T e^{iDt} T^H$

falls Dee effizient bestimmbar

$\Rightarrow |i 0\rangle \rightarrow |i \text{Dee}\rangle \rightarrow e^{i \text{Dee} t} |i \text{Dee}\rangle$

(Coord entry; Phase gate)

$\rightarrow e^{i \text{Dee} t} |i 0\rangle$

(de Coord entry)

für k-local $\rightsquigarrow$ effiziente Diagonalisierung ✓ der H_j

Aber: $e^{iHt} \neq \prod_{j=1}^{m} e^{iH_j t}$

gilt nur, wenn alle H_j kommutieren 0 0

da $e^{A+B} = \sum_{n=0}^{\infty} \frac{(A+B)^n}{n!} = \sum_{n=0}^{\infty} \sum_{k=0}^{n} \binom{n}{k} \frac{A^k B^{n-k}}{n!}$, $\binom{n}{k} = \frac{1}{k!(n-k)!}$

Cauchy prod. $= \sum_{k=0}^{\infty} \frac{A^k}{k!} \cdot \sum_{n=0}^{\infty} \frac{B^n}{n!} = e^A \cdot e^B$

nur für Komm 0 0

Lösung: Trotter / Lie-Produkt Formel

Theorem Sei $H = H_1 + H_2$, alle hermitsch

$$\Rightarrow \lim_{\ell \to \infty} \left(e^{iH_1 t/\ell} \cdot e^{iH_2 t/\ell} \right)^{\ell} = e^{iHt}$$

Es gilt sogar

$$\left\| e^{iHt} - \left(e^{iH_1 t/L} \cdot e^{iH_2 t/L} \right)^{L} \right\| \leq C \cdot \frac{t^2}{L}$$

mit $C = C(\|H_1\|, \|H_2\|)$

Beweis Taylor: zu verstehen als Terme in H_1, t, L beschr. durch

$$e^{iH_1 t/L} = I + iH_1 \frac{t}{L} + O\left(\|H_1\|^2 \frac{t^2}{L^2} \right)$$

$$\Rightarrow e^{iH_1 t/L} \cdot e^{iH_2 t/L} = \left(I + iH_1 \frac{t}{L} + O\left(\|H_1\|^2 \frac{t^2}{L^2} \right) \right) \cdot \left(I + iH_2 \frac{t}{L} + O\left(\|H_2\|^2 \frac{t^2}{L^2} \right) \right)^{L}$$

$$= \left(I + i \cdot (H_1 + H_2) \frac{t}{L} + O\left(\max(\|H_1\|, \|H_2\|)^2 \cdot t^2/L^2 \right) \right)^{L}$$

Taylor

$$\Rightarrow \quad (*) \qquad = \Big(\underbrace{e^{i(H_1+H_2)\frac{t}{L}}}_{=:A} + \underbrace{O\left(\max(\|H_1\|, \|H_2\|)^2 \frac{t^2}{L^2} \right)}_{=:B} \Big)^{L}$$

Da $(A+B)^L = A^L + \sum_{j=0}^{L-1} A^{L-j-1} B A^j + \dots + B^L$

$\underbrace{\qquad\qquad}_{\mathcal{O}(\|B\|^2)}$

$\Rightarrow (*) = e^{i(H_1+H_2)t} + L \cdot \mathcal{O}\left(\max(\|H_1\|, \|H_2\|)^2 \frac{t^2}{L^2}\right)$

$\Rightarrow \left\| e^{i(H_1+H_2)t} - \left(e^{iH_1 t/L} \cdot e^{iH_2 t/L}\right)^L \right\| \leq C \frac{t^2}{L}$

$\downarrow L \to \infty$

0

□

Bem. ·) Für eff. Simulation:

$\max(\|H_1\|, \|H\|_2) = \mathcal{O}(\text{poly}(n))$

·) Fehler $\leq \varepsilon$ vorgeg.

$\Rightarrow L = \mathcal{O}\left(C(H_1, H_2), \frac{t^2}{\varepsilon}\right)$

·) hier: Splitting 1. Ordnung

höhere Ordnung auch möglich, z.B. Strang splitting

$\left\| e^{iHt} - \left(e^{iH_2 t/(2L)} e^{iH_1 t/L} e^{iH_2 t/(2L)}\right)^L \right\| \leq C \frac{t^3}{L^2}$

$\Rightarrow L = \mathcal{O}\left(C \frac{t^{\frac{3}{2}}}{\sqrt{\varepsilon}}\right)$

Strang splitting: 2. Ordnung

Methoden Ordnung p möglich

$\rightsquigarrow L = O\left(t^{\frac{p+1}{p}} \varepsilon^{-\frac{1}{p}}\right)$ $\in$ fast linear in t !!

·) Falls k-local Ham. mit m-Termen:

$$\left\| e^{iHt} - \left(e^{iH_1 t/L} \cdot \ldots \cdot e^{iH_m t/L}\right)^L \right\| \leq C \frac{m^2 t^2}{L}$$

·) Fehler Abschätzung rel. pessimistisch, in Anw. besser

Vorteil: einfach, braucht keine zus. Qubits

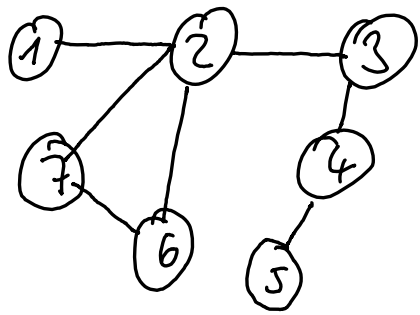
Essentielle Frage: Wie findet man Zerlegung

$H = \sum H_i$??

z.B. mittels Graphentheorie „graph coloring methods"

Def. Ein ungerichteter Graph $G=(V,E)$ ist def. durch Knotenmenge V und Kanten E, die ungeordnete Paare von Knoten sind.

Bsp.



$V=\{1,\dots,7\}$

$E=\{(1,2),(2,3),(2,6),(2,7),(6,7),(3,4),(4,5)\}$

Def. $v_1,v_2 \in V$ heißen verbunden, wenn $\exists e \in E$ s.d. $e=(v_1,v_2)$.

Der Grad eines Knotens $v \in V$ ist $\#\{v_i : (v,v_i) \in E\}$.

Sei $|V|=n$. Dann kann ein Graph mittels seiner $n \times n$-adjecency matrix A beschrieben werden:

$$A_{ij} = \begin{cases} 1 & \text{falls } i,j \in V : (i,j) \in E \\ 0 & \text{sonst} \end{cases}$$

obiges Bsp:

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

graph coloring problem: gegeben k Farben, können alle Kanten E von G so eingefärbt werden, dass zwei Kanten, die einen Endpunkt teilen nie die gleiche Farbe haben?

Antwort: Vizings Theorem:

Theorem Sei $G=(V,E)$ ein Graph mit maximalem Grad der Knoten d

$\Rightarrow$ $\exists$ Lösg. für das gc-problem mit $k \leq d+1$

Anwendung auf Ham. Kreis.:

1. Identifiziere H mit adj. matrix A:

$$A_{ij} = \begin{cases} 1 & \text{falls } H_{ij} \neq 0 \\ 0 & \text{falls } H_{ij} = 0 \end{cases}$$

$\rightsquigarrow$ liefert zugehörigen Graph G

2. Bestimme eine Einfärbung von G mit k Farben

3. Zerlege G bzw. A anhand der Farben in Sub-Graphen

$$A = \sum_{c=1}^{k} A_c$$

Für allg. A → unmöglich effizient realisierbar!

→ verlange sparsity

Annahme $A \in \mathbb{C}^{2^n \times 2^n}$ Hermitesch, $\|A\| \leq 1$ sei s-sparse und wir haben sparse access auf die Matrix-Einträge. D.h.:

-) jede Zeile/Spalte von A hat max. s nicht-0 Einträge
-) Wir haben query

$$O_A : |i\,j\rangle|0\rangle \mapsto |i\,j\rangle|A_{ij}\rangle$$

hier: hinteres Register groß genug, dass $A_{ij} \in \mathbb{C}$ exakt/hinreichend genau als Binärdarst. geschr.

-) Wir haben weitere query

$$O_C : |j\,\ell\rangle \mapsto |j\,\nu(j,\ell)\rangle$$

wobei $\nu(j,\ell) \in \{0,\dots,N-1\}$ die Position der ℓ-ten nicht-0 Einträge in der j-ten Spalte von A ist.

-) Wir können O_A^{-1}, O_C^{-1} ausführen.

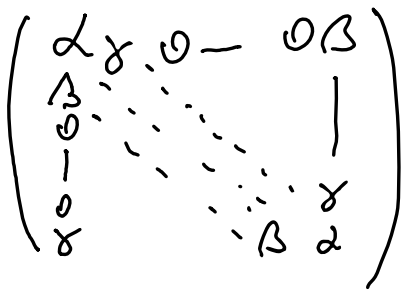
- Sparsity $\rightsquigarrow$ max. Grad der Graphen $\leq s$
- Vizing Thm $\rightsquigarrow$ brauche max. $s+1$ Farben
- Effizient berechenbare Einfärbung mit s^2 Farben
- Matrizen A_C sind symmetrisch, 1-sparse
  $\Rightarrow$ zugeh. H_C effizient simulierbar da Matrizen in $O(1)$ diagonalisierbar

  o.B.d.A. Diagonaleinträge da $H = \underbrace{\text{diag}(H)}_{\text{effizient } \checkmark} + \text{Rest}$
- Gesamtaufwand (naive Realisierung hier)

$$O(s^2 t^2 \text{poly}(n)/\varepsilon)$$

Q: Wie wird „sparse access" implementiert?

Bsp. circulant matrix



3-sparse

$\cdot) \; r(j,l) = j+l-1 \bmod N \quad l = 0,1,2$

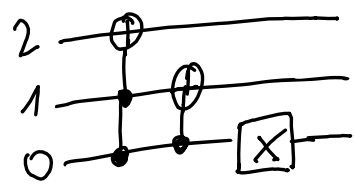
(hier Matrizen von 0 weg indiziert)

$$\rightarrow O_C: \; |j\rangle \mapsto \begin{cases} |\mathrm{mod}(j-1,N)\rangle & l=0 \\ |j\rangle & l=1 \\ |\mathrm{mod}(j+1,N)\rangle & l=2 \end{cases}$$

$\mathrm{mod}(j \pm 1, N)$ können durch shift Permutationen realisiert werden

$$R = \begin{pmatrix} 0 & 1 & & 0 \\ & & 1 & \\ 1 & & & 0 \end{pmatrix} \qquad L = \begin{pmatrix} 0 & & & 1 \\ & & & 0 \\ 1 & & & \\ & 0 & & \end{pmatrix}$$

Circuits:

L: 

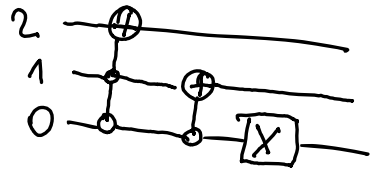
(für 3 qubits)

($X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ bitflip)

zu verstehen als Stellen in Binärdarst.

z.B.: $|0\,1\,0\rangle \mapsto |0\,1\,1\rangle$ also $2 \mapsto 3$

$|1\,1\,1\rangle \mapsto |0\,0\,0\rangle$ $\quad 7 \mapsto 0$

R: 

hier ⊕ controlled NOT, aktiv wenn Kontrolle = 0

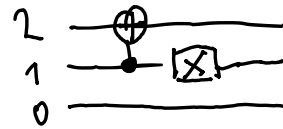
⊕ analog CCNOT, aktiv wenn beide = 0

---

brauchen auch L^2, R^2

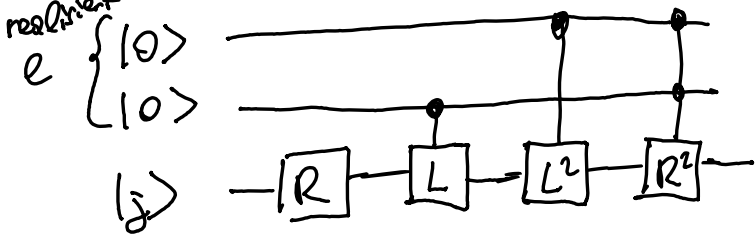
es gilt: 

$=$ 

$=$ 

analog für R^2_0

O_C circuit:



-) R-gate für alle ℓ $\quad j \mapsto \mathrm{mod}(j-1, N)$
  passt für $\ell = 0$
  also $|00\rangle$
-) für $\ell = 1 \simeq |01\rangle$ $\quad j \mapsto j$
  $\rightsquigarrow$ mache R-gate mittels L-gate rückgängig
-) $\ell = 2 \simeq |10\rangle$ $\quad j \mapsto \mathrm{mod}(j+1, N)$
  mache L^2-shift $\Rightarrow RL^2 = L$-shift
-) $\ell = 3 \simeq |11\rangle$ – entspricht 0-Eintrag
  $\rightsquigarrow$ mache R^2-shift um L^2-shift umkehren

O_A Circuit

generell: controlled rotations: unitäre Op.

$$U_\theta : |\theta\rangle|0\rangle \mapsto |\theta\rangle\left(\cos(\pi\theta)|0\rangle + \sin(\pi\theta)|1\rangle\right)$$

↑ Kontrolle ↑ Ziel

hier: $\theta \in [-1,1)$, Ann. $\theta = \theta_0 \cdot 2^{-1} + \theta_1 \cdot 2^{-2} + \dots + \theta_{d-1} \cdot 2$ exakte Binärdarst.

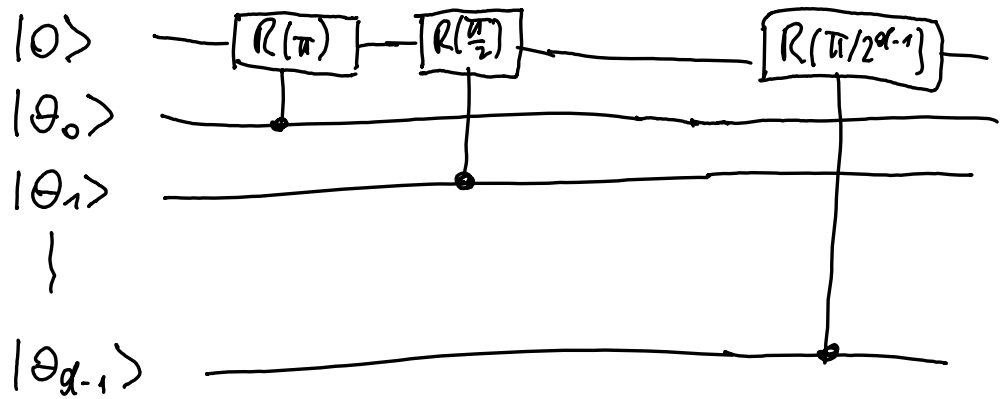
für $|\theta\rangle = |0\rangle \rightsquigarrow$ Output $|0\rangle|0\rangle$

$|\theta\rangle = |1\rangle \rightsquigarrow$ Drehung um $\pi\theta$ $\rightarrow$ Matrix $\begin{pmatrix} \cos J & -\sin J \\ \sin J & \cos J \end{pmatrix} =: R(2J)$

Schreibweise: 

$\simeq$ 1-qubit Rotation um y-Achse

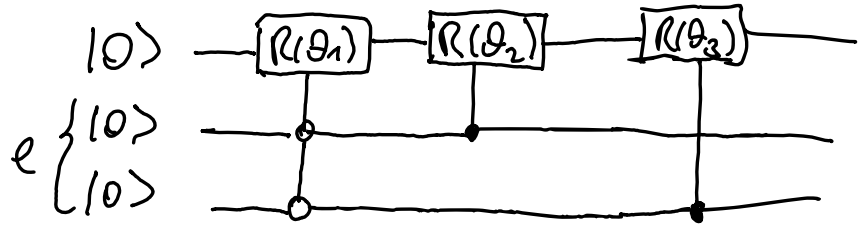
Implementierung



$\rightsquigarrow$ $\Theta(d)$-gates

$\rightsquigarrow$ generiere Wert $\alpha \in [0,1]$ mittels Rotation um Winkel $\theta_1 = \arccos \alpha \cdot \frac{1}{\pi}$

analog $\rightsquigarrow \theta_2 = \frac{1}{\pi} \arccos(\beta)$, $\theta_3 = \frac{1}{\pi} \arccos(\gamma)$



$e = 0 \rightsquigarrow \alpha$, $e = 1 \rightsquigarrow \beta$, $e = 2 \rightsquigarrow \gamma$

Modifikation für 3-dig Matrix:

füge 2 zusätzliche Rotationen $R(\theta_4), R(\theta_5)$ hinter die Elemente A_{1n} bzw A_{n1} auf 0 zurückrotieren

$\rightsquigarrow$ brauche 1 control Qubit mehr

Einlesen der rechten Seite

gegeben $b \in \mathbb{R}^N$, Ziel: $|b\rangle := \frac{\sum b_i |i\rangle}{\|b\|_2}$

in $\mathcal{O}(\log N)$ (bzw $|\tilde{b}\rangle$ mit $\tilde{b} \approx b$ bis auf machine prec.)

mögliche Lösg: <u>quantum RAM</u> (qRAM)

klassische Datenstruktur auf die mit Überlagerungen von q-Zuständen zugegriffen werden kann.

---

oBdA $\|b\|_2 = 1$ da das feste Zahl

$\rightsquigarrow$ kann mittels Binärdarst. eingelesen werden

---

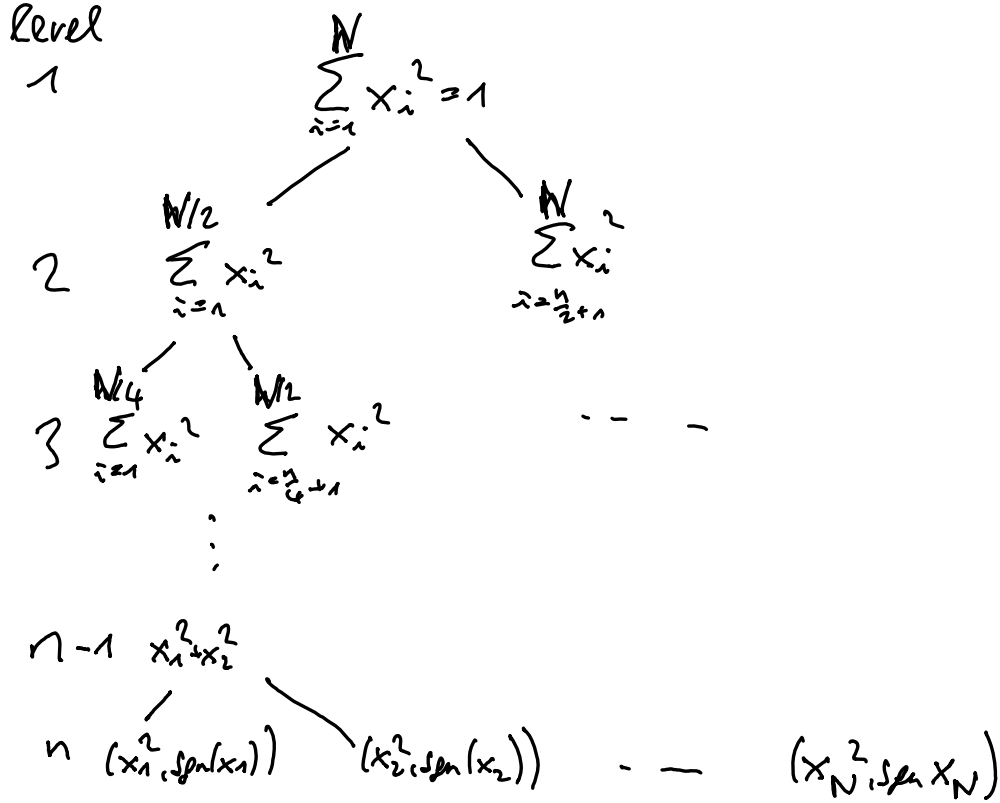
divide and conquer – Idee $\rightsquigarrow$ speichere

x in Binärbaum B_x

Wurzel $\sum_{i=1}^{N} x_i^2 = 1$

Tiefe $\log N = n$

Blätter $(x_i^2 ; \operatorname{sgn} x_i)$



Anm.:) Jeder Knoten ist Summe der Kinder
(zusätzliche sgn. Information im Blatt gespeichert)

) Vektor wird geladen: Läuft durch Baum von Wurzel weg, fügt wenn nötig neue Register hinzu, verwendet controlled rotation, um Werte in den jeweiligen Nodes zu realisieren

.) B_x hat $\Theta(N)$ Knoten

Ann.: ist pre-computed (klassisch)
↳ binär in Arbeitsmed...

Algorithmus (lade Vektor aus qRAM)

```
% input:  x∈R^N gesp. im Baum B_x
% output: |x>
Initialisiere n-qubits |0...0>, bez. mit |q_1>...|q_n>
r = root(B_x), call processNode(r)
processNode (vertex v)
        v_l, v_r := kinder(v)
        θ = arccos sqrt(v_l / v)
        Controlled Rotation |q_k> = cos θ |0> + sin θ |1>
          falls qubits |q_1>...|q_{k-1}> ≃ Binärdarst. von v
        if v_l, v_r Blätter
            |q_k> = processSign(q_k, v_l, v_r)
            return
        else
            processNode(v_l)
            processNode(v_r)
        end
end
```

```
process sign ( q_k, v_l, v_r )
    if sgn(v_l) = sgn(v_r) = +1    return  q_k
    if  —   "   = -1               return -q_k
    if sgn(v_l) = 1, sgn(v_r) = -1 return  i q_k
    if sgn(v_l) = -1, sgn(v_r) = 1 return -i q_k
end
```

i-phase flip

Lemma Falls $x \in \mathbb{R}^N$ in B_x vorgespeichert

$\Rightarrow$ Alg. erzeugt Zustand $|x\rangle = \sum_{i=1}^{N} x_i |i\rangle$

in $\Theta(n) = \Theta(\log N)$.

Beweis.) Zeige Alg. erzeugt tatsächlich $|x\rangle$

Alg. $\rightsquigarrow$ wandere durch Baum, multipliziere Knoten v_k mit $\sqrt{\frac{v_k}{v_{k-1}}}$, Blatt i nach mit signum

$k \simeq$ level

$$\Rightarrow \prod_{k=2}^{n} \sqrt{\frac{v_k}{v_{k-1}}}\, \mathrm{sgn}(x_i) = \sqrt{\underbrace{\frac{v_n}{v_1}}_{\frac{x_i^2}{1}}}\, \mathrm{sgn}(x_i) = x_i$$

✓

Laufzeit: 2^k Rotationen (in $\mathcal{O}(1)$) auf Level k

passiert parallel $\rightsquigarrow$ kontrollierte Op. auf selben Qubit !

(Kontrolle checkt binär-Reg. von Vorfahre)

$\rightsquigarrow$ # Levels $= n = \log N$ beschr. Aufwand

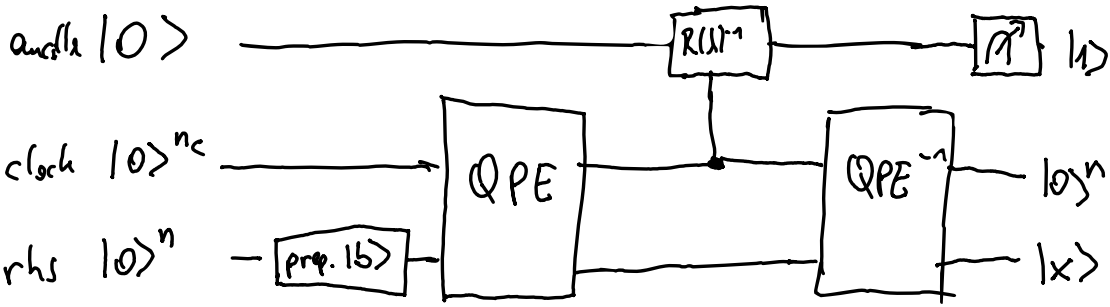
Aber: Annahme B_x precomputed signifikant!

kann Speedup ruinieren!

Zumindest: Lemma zeigt: kann dann $|b\rangle$ und auch Kopien von $|b\rangle$ $\binom{0}{0}$ schnell erzeugen

HHL revisited

Circuit



1. berf. $|b\rangle$ (z.B QRAM)
   mit $n = \log N$ qubits um $|b\rangle$ darzustellen

2. Wende Quantum Phase Estimation an auf

$$|0\rangle\,|b\rangle = \sum_{j=0}^{N-1} \beta_j\,|0\rangle|u_j\rangle \qquad (\text{hier } |0\rangle = |0\rangle^{n_c})$$

   mit $U = e^{iA}$

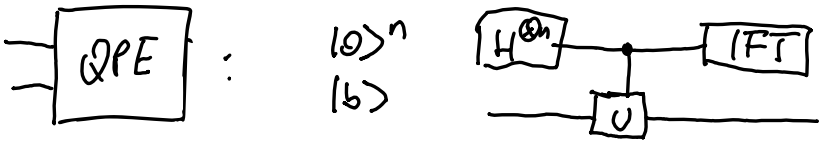
   benötigt: $H^{\otimes n_c}$ ✓

   U^{2^j} $j = 0, \dots, n_c - 1$ Ham. Sim. ✓

   IQFT $\rightarrow \mathcal{O}(n^2)$-gates ✓

   liefert: Approx. zu EWen von A (nicht von U selbst)

Approx., da in QPE angenommen: λ_j hat exakte Binärdarst. mit n_c-qubits



U realisiert $|j\rangle|\psi\rangle \mapsto \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{2\pi i \lambda j} |j\rangle|\psi\rangle$

mit λ EW zu EV ψ

$\Rightarrow$ Anwendung auf $|b\rangle$ von QPE: Zustand

$$|0\rangle|b\rangle \mapsto \sum \beta_j |\tilde{\lambda}_j\rangle |v_j\rangle$$

($\tilde{\lambda}_j$: binär-Rep. von λ_j)

3. Controlled rotation: füge Ancilla qubit hinzu und drehe um Winkel (Dreh. mit $R_y(2\cdot)$ diagonalisiert)

$$\frac{\theta}{2} = \arcsin\left(\frac{C}{\tilde{\lambda}_j}\right)$$

$$\Rightarrow \sum_j \beta_j |\tilde{\lambda}_j\rangle |v_j\rangle \left(\sqrt{1 - \frac{C^2}{\tilde{\lambda}_j^2}}\,|0\rangle + \frac{C}{\tilde{\lambda}_j}|1\rangle\right)$$

hier: C .. Konstante s.d.

$$C \leq \min |\lambda_j| = \Theta\left(\frac{1}{\kappa}\right)$$

Bem.: $\arcsin(x)$ kann mit $O(\text{poly}_k(n))$ elementaren Gates (approx.) realisiert werden
→ Literatur

4. Inverse QPE „uncomputing": Zustand

$$\sum_{j=0}^{N-1} \beta_j |0\rangle |v_j\rangle \left(\sqrt{1 - \frac{C^2}{\lambda_j^2}} |0\rangle + \frac{C}{\lambda_j} |1\rangle \right)$$

5. Messen des letzten Qubits
Falls Ergebnis $|1\rangle$ ⇒ Zustand

$$C \sum_{j=0}^{N-1} \frac{\beta_j}{\lambda_j} |0\rangle |v_j\rangle$$

(proport. zu $|\tilde{x}\rangle$)

Wahrsch. $|1\rangle$ zu messen: $\dfrac{1}{\sqrt{\sum_{j=0}^{N-1} C^2 \frac{|\beta_j|^2}{|\lambda_j|^2}}}$

Bem. Normalisierungskonstante kürzt C
⇒ tritt nicht in Lsg. auf, aber sehr wohl in der Erfolgswahrsch.

Quantum composer Bsp

$A = \begin{pmatrix} 1 & -\frac{1}{3} \\ -\frac{1}{3} & 1 \end{pmatrix}$ $b = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ 1-qubit System

Lsg. von $Ax = b$ $\rightsquigarrow x = \frac{1}{8}\begin{pmatrix} 3 \\ 9 \end{pmatrix}$

EW/EV von A: $\lambda_0 = \frac{2}{3}$, $\lambda_1 = \frac{4}{3}$

$v_0 = \frac{1}{\sqrt{2}}\begin{pmatrix} -1 \\ -1 \end{pmatrix}$ $v_1 = \frac{1}{\sqrt{2}}\begin{pmatrix} -1 \\ 1 \end{pmatrix}$

brauche $U = e^{iAt}$

$U^2 = e^{i2At}$

wähle $t = \frac{3\pi}{4}$

$\rightsquigarrow$ exakte Binärdarst. für EWe in QPE mit 2 qubits

$$U = \frac{1}{\sqrt{2}}\begin{pmatrix} -1 & -1 \\ -1 & 1 \end{pmatrix} \underbrace{\begin{pmatrix} e^{i\lambda_0 t} & 0 \\ 0 & e^{i\lambda_1 t} \end{pmatrix}}_{\begin{pmatrix} i & 0 \\ 0 & -1 \end{pmatrix}} \frac{1}{\sqrt{2}}\begin{pmatrix} -1 & -1 \\ -1 & 1 \end{pmatrix}$$

$$= \frac{1}{2}\begin{pmatrix} -1+i & 1+i \\ 1+i & -1+i \end{pmatrix}$$

analog $U^2 = \begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix}$

Implementiert mit 4-Parameter unitary

Gate in IBM-Q:

$$U = \begin{pmatrix} e^{i\delta}\cos(\theta/2) & -e^{i(\gamma+\lambda)}\sin(\theta/2) \\ e^{i(\gamma+\varphi)}\sin(\theta/2) & e^{i(\gamma+\varphi+\lambda)}\cos(\theta/2) \end{pmatrix}$$

$\rightsquigarrow \theta = \frac{\pi}{2}, \varphi = -\frac{\pi}{2}, \lambda = \frac{\pi}{2}, \gamma = \frac{3\pi}{4}$ für U

$\theta = \pi, \varphi = \pi, \lambda = 0, \gamma = 0$ für U^2

Controlled rotation

chain 2 qubits → 4

EWe aus QPE $\tilde{\lambda}_j = N\lambda_j t/2\pi$

$\rightarrow \tilde{\lambda}_0 = 1 \quad \tilde{\lambda}_1 = 2$

→ kann Konstante $C = 1$ wählen

$\rightsquigarrow \theta_1 = 2\arcsin\left(\frac{1}{\tilde{\lambda}_1}\right) = \pi$

$\theta_2 = 2\arcsin\left(\frac{1}{\tilde{\lambda}_2}\right) = \frac{\pi}{3}$

Resultat: bekomme normierte Wahrscheinlichkeiten

Zust.	P
$\|00\rangle$	0.1875
$\|01\rangle$	0.0625
$\|10\rangle$	0.1875
$\|11\rangle$	0.5625

↑ ancilla

↑ 1D-qubit

— nur relevant, da ancilla $|1\rangle$ gemessen werden muss

Wahrsch.: Verhältnis 1:9

norm. Lösg: $|x\rangle = \frac{3}{\sqrt{90}}|0\rangle + \frac{9}{\sqrt{90}}|1\rangle$

cond. Messung ↳ von Ancilla $|0\rangle$ mit $P = \frac{1}{10}$ $|1\rangle$ mit $P = \frac{9}{10}$

ebenfalls 0

Fehler- und Komplexitätsanalyse

bisherige Ann.: alle Größen haben exakte Binärdarst.
i.A. nicht machbar für EW von A

→ Phase estimation produziert EWe $\tilde{\lambda}_j$ mit

$$|\tilde{\lambda}_j - \lambda_j| \leq \delta$$

mit Wahrsch. $1 - \frac{1}{\text{poly}(n)}$ mit

Laufzeit $O(T_U \text{poly}(n)/\delta)$ wobei

T_U – Rechenzeit für Impl. von $U = e^{iA}$

Problem: numerische Stabilität von $\lambda_j \mapsto 1/\lambda_j$
mit contr. Rotation

Falls $\lambda_j \approx 0$ → kleine Störung $\tilde{\lambda}_j = \lambda_j + \varepsilon$

→ großen Effekt auf Fehler:

$$\left|\frac{1}{\lambda_j} - \frac{1}{\tilde{\lambda}_j}\right| = \left|\frac{\varepsilon}{\lambda_j(\lambda_j+\varepsilon)}\right| \approx \frac{\varepsilon}{\lambda_j^2}$$

Sing. größer als ε !

→ Problem mit schlechter Konditionszahl κ von A !

Lösg: Filter Funktionen

Invertiere EWe nur sofern $\lambda \geq \frac{1}{K}$ ← Kond.-Zahl

$\rightsquigarrow f(\lambda) = 0$ für $\lambda < \frac{1}{2K}$

für große EW $\sim \frac{1}{\lambda}$

$\Rightarrow f(\lambda) = \frac{1}{2K\lambda} \quad \lambda \geq \frac{1}{K}$

dazwischen interpolierend (für num. Stbs.)

$f(\lambda) = \frac{1}{2} \sin\left(\frac{\pi}{2} \cdot (2K\lambda - 1)\right) \quad \frac{1}{2K} \leq \lambda < \frac{1}{K}$

$\Rightarrow f$ stetig

analog umgek. Filter

$$g(\lambda) = \begin{cases} 0 & \lambda \geq \frac{1}{K} \\ \frac{1}{2}\cos\left[\frac{\pi}{2}(2K\lambda - 1)\right] & \frac{1}{K} > \lambda > \frac{1}{2K} \\ \frac{1}{2} & \frac{1}{2K} \geq \lambda \end{cases}$$

$\Rightarrow g$ stetig und $f^2(\lambda) + g^2(\lambda) \leq 1 \quad \forall \lambda$

Anm: nur eine Wahl, nicht eindeutig,
auch Abschneiden bei $\frac{1}{2K}$ kann woanders sein

statt controlled rotation füge 3-qubit register hinzu

also keine Inversion durchgef.

$$|h(\tilde{\lambda}_j)\rangle := \sqrt{1 - f(\tilde{\lambda}_j)^2 - g(\tilde{\lambda}_j)^2}\,|\text{nothing}\rangle + f(\tilde{\lambda}_j)\,|\text{well}\rangle + g(\tilde{\lambda}_j)\,|\text{ill}\rangle$$

$|\text{well}\rangle$: hier EWe invertiert

$|\text{ill}\rangle$: Teile von $|b\rangle$ im schlecht kond. TR von A

danach $(QPE)^{-1}$ und Messen von $|\text{well}\rangle$

---

prob: Endzustand $\simeq$

$$\sum_{j:\lambda_j \geq \frac{1}{\kappa}} \lambda_j^{-1} \beta_j |u_j\rangle |\text{well}\rangle + \sum_{j:\lambda_j < \frac{1}{\kappa}} \beta_j |u_j\rangle |\text{ill}\rangle$$

Lemma Die Abb. $\lambda \mapsto |h(\lambda)\rangle$ ist Lipschitz stetig mit $L = O(\kappa)$, i.e.:

$$\| |h(\lambda_i)\rangle - |h(\lambda_j)\rangle \|_2 \leq C \kappa |\lambda_i - \lambda_j|$$

Bew: Elementar durch Abschätzen der Abl. von f, g (expl. berechnen bar)

□

Ziel: Fehlerabsch. für inexakte QPE nach gefilterter Inversion der EW

exakt: $|\psi\rangle := \sum \beta_i |v_i\rangle |h(\lambda_i)\rangle$

approx.: $|\tilde{\psi}\rangle := \sum \beta_i |v_i\rangle |h(\tilde{\lambda}_i)\rangle$

$$\Rightarrow \| |\psi\rangle - |\tilde{\psi}\rangle \|_2^2 = \| |\psi\rangle \|^2 + \| |\tilde{\psi}\rangle \|^2 - 2\,\mathrm{Re}\langle\psi|\tilde{\psi}\rangle$$

$$= 2\left(1 - \underbrace{\mathrm{Re}\langle\psi|\tilde{\psi}\rangle}_{\in [0,1] \text{ (C.S.)}}\right)$$

$$\Rightarrow \mathrm{Re}\langle\psi|\tilde{\psi}\rangle \underset{\uparrow v_i \text{ ONB}}{=} \sum_{i=1}^{N} |\beta_i|^2 \,\mathrm{Re}\langle h(\lambda_i)|h(\tilde{\lambda}_i)\rangle$$

Lemma $\Rightarrow \mathrm{Re}\langle h(\lambda_i)|h(\tilde{\lambda}_i)\rangle \geq 1 - \frac{C^2K^2}{2}|\lambda_i - \tilde{\lambda}_i|^2$

$$\geq 1 - \frac{C^2K^2\delta^2}{2}$$

$\uparrow$ Fehler pro EW $\leq \delta$

$$\leq \underbrace{\sum_{i=1}^{N} |\beta_i|^2}_{=1} C^2K^2\delta^2$$

$$\Rightarrow \| |\psi\rangle - |\tilde{\psi}\rangle \|_2 \leq CK\delta$$

Für Fehler $\mathcal{O}(\varepsilon)$

↳ Phase estimation Fehler $\delta = \mathcal{O}\left(\frac{\varepsilon}{K}\right)$

↳ Runtime $\mathcal{O}(K \cdot \text{poly}(n)/\varepsilon)$

Messung: möchte $\text{Anc}\langle 1 | \rangle$ (wohlkond. A)
$\text{Anc}\langle 1 | \text{well} \rangle$ (generelleres A)

$\rightsquigarrow$ Erfolgswahrsch

$$p \geq \sum_{i:\lambda_i \geq 1/K} |\beta_i|^2 \left|\frac{1}{\lambda_i K}\right|^2 = \mathcal{O}\left(\frac{1}{K^2}\right)$$

↑ note: $\sum \frac{|\beta_i|^2}{|\lambda_i|^2} \approx \|A^{-1}\|$

kann mittels „amplitude amplification"
(vgl. Grover-Algorithmus) auf $\mathcal{O}\left(\frac{1}{K}\right)$
verbessert werden

$\Rightarrow \mathcal{O}(K)$ von Prozedur nötig um
$|\text{well}\rangle$ mit bel. hoher Wahrsch. zu
erhalten

Gesamtaufwand:

-) state prep.: $\mathcal{O}(n)$ (falls QRAM precomputed)
-) Hamiltonian simulation:

  e^{iAt} wenn A s-sparse bis auf Fehler ε in

  $$\mathcal{O}\left(nst \cdot \text{poly}\left(\log \tfrac{st}{\varepsilon}\right)\right)$$

  (bestes Resultat in Literatur, einfache Methoden in VO vorgestellt $\mathcal{O}(ns^2t^2/\varepsilon)$)
-) QPE $\mathcal{O}(\text{poly}(n)\, K/\varepsilon \cdot T_U)$
-) Ampl. amp. $\mathcal{O}(K)$

$$\Rightarrow \mathcal{O}\left(\text{poly}(n)\, K^2 s/\varepsilon \cdot \text{poly}\left(\log \tfrac{s}{\varepsilon}\right)\right)$$

vgl. CG:

$$\mathcal{O}\left(\exp(n)\, s\sqrt{K}\, \ln\left(\tfrac{2}{\varepsilon}\right)\right)$$

besser in n aber schlechter in K, ε

↳ exp. speedup!

↳ Verbesserbar??

Verbesserung des HHL-Algorithmus

CG: Komplexität vs. Genauigkeit $\mathcal{O}(\log \frac{1}{\varepsilon})$

HHL: (wegen QPE) $\longrightarrow \mathcal{O}(\frac{1}{\varepsilon})$

Ziel: q-LGS Algorithmus mit ebenfalls log. Abh. von ε^{-1}, immer noch exp. speedup in N

Idee: Approximiere A^{-1} direkt

vgl. Cayley-Hamilton Theorem:

p - char. Polynom von $A \Rightarrow p(A) = 0$

$\in \mathcal{P}_N \Rightarrow A^N + \alpha_{N-1} A^{N-1} + \dots + \alpha_0 I = 0 \qquad \alpha_i \in \mathbb{C}$

$\Leftrightarrow A^{-1} = -\frac{1}{\alpha_0} \cdot \left(\alpha_1 I + \dots + \alpha_{N-1} A^{N-2} + A^{N-1} \right)$

$= p_{N-1}(A)$

Berechnung von α_i zu teuer, aber

ev. $\exists$ Polynom $q_m \in \mathcal{P}_m$ mit $m \ll N-1$

s.d. $A^{-1} \simeq q_m(A)$

und $q_m(A)$ ist eff. implementierbar

2 Mögl. 1. trigonometrische Polynome $\rightsquigarrow$ Fourier Approx. $\rightarrow$ unitär

2. Chebyshev Polynome $\hookrightarrow$ minimieren $\|q_m(A)\|_2$

Q: Wie wirkt sich approx. der Matrix auf Endzustand aus?

Lemma Sei B hermitsch mit $\|B^{-1}\| \leq 1$ und D so dass $\|B-D\| \leq \varepsilon < \frac{1}{2}$

$\Rightarrow$ Zustände $|x\rangle := \frac{B|\psi\rangle}{\|B|\psi\rangle\|}$ und $|\tilde{x}\rangle := \frac{D|\psi\rangle}{\|D|\psi\rangle\|}$ erfüllen $\| |x\rangle - |\tilde{x}\rangle \| \leq 4\varepsilon$

Beweis: Δ-Ungl.!

$$\| |x\rangle - |\tilde{x}\rangle \| = \left\| \frac{B|\psi\rangle}{\|B|\psi\rangle\|} - \frac{D|\psi\rangle}{\|D|\psi\rangle\|} \right\|$$

$$\leq \underbrace{\left\| \frac{B|\psi\rangle}{\|B|\psi\rangle\|} - \frac{B|\psi\rangle}{\|D|\psi\rangle\|} \right\|} + \frac{1}{\|D|\psi\rangle\|} \| B|\psi\rangle - D|\psi\rangle \|$$

$$\leq \frac{\big|\|D|\psi\rangle\| - \|B|\psi\rangle\|\big|}{\|D|\psi\rangle\|}$$

nochmal Δ-Ungl.:

$1 \leq \|B|\psi\rangle\| \leq \|D|\psi\rangle\| + \|(B-D)|\psi\rangle\| \leq \|D|\psi\rangle\| + \varepsilon$

(1 $\leq$ bzw.)

$\Rightarrow$ beide Terme $\leq \frac{\varepsilon}{\|D|\psi\rangle\|} + \frac{\varepsilon}{\|D|\psi\rangle\|} \leq 2\frac{\varepsilon}{1-\varepsilon} \leq 4\varepsilon$

□

$\Rightarrow$ wende Lemma mit $B = A^{-1}$ und $D \approx A^{-1}$ an

---

Fourier approach

Approximiere $A^{-1} \approx \sum_j \alpha_j e^{-iAt_j}$ $\quad \alpha_j, t_j \in \mathbb{R}$

also durch Linearkomb. von unitären Op. (LCU)

2 Fragen: .) Wie genau sieht die Approx. aus?

.) Ist diese effizient implementierbar?

Implementierung von LCU

o.B.d.A. $\alpha_j > 0$ da Phase sowieso nicht messbar

Ziel: Impl. von $M = \sum_j \alpha_j U_j$, wobei U_j unitär

M nicht unbedingt!

Lemma Sei $M = \sum_j \alpha_j U_j$ mit $\alpha_j > 0$, U_j unitär und V Abb. $V|0^m\rangle := \frac{1}{\sqrt{\alpha}} \sum_j \sqrt{\alpha_j}|j\rangle$ mit $\alpha = \sum_j \alpha_j$

sowie $U := \sum_j |j\rangle\langle j| \otimes U_j$

$\Rightarrow V^H U V =: W$ erfüllt

$$W|0^m\rangle|\psi\rangle = \frac{1}{\alpha}|0^m\rangle M|\psi\rangle + |\phi^\perp\rangle$$

$\forall$ zust. $|\psi\rangle$, wobei $(|0^m\rangle\langle 0^m| \otimes I)|\phi^\perp\rangle = 0$ $\quad =: \Pi|\phi^\perp\rangle$

Also: 1. Term realisiert M (nachher A^{-1})

2. Term orthogonal zu $|0^m\rangle$ im 1. Register

$\Rightarrow$ Messen von $|0^m\rangle$ im 1. Reg. $\rightsquigarrow$ Impl. von M

Beweis: $W|0^m\rangle|\psi\rangle = V^H U \left(\frac{1}{\sqrt{2}} \sum_j \sqrt{\alpha_j}|j\rangle|\psi\rangle \right)$

$\overset{\text{select } U_j}{=} V^H \left(\frac{1}{\sqrt{2}} \sum_j \sqrt{\alpha_j}|j\rangle U_j|\psi\rangle \right)$

$= \underbrace{\Pi V^H(\quad)}_{\sim} + \underbrace{(I-\Pi)V^H(\quad)}$

$(|0^m\rangle\otimes I)\left(\frac{1}{\sqrt{2}}\sum\sqrt{\alpha_j}\langle j|\otimes I\right)|\phi^\perp\rangle$ da $\Pi(I-\Pi)=0$

da Π 1. Reg. auf $|0^m\rangle$ Proj. und aus Def. von V^H

$\Rightarrow$

$= \frac{1}{2}|0^m\rangle \sum_j \alpha_j U_j|\psi\rangle + |\phi^\perp\rangle$

□

·) U -- „select U", wählt U_j anhand von Kontrollregister aus

·) Erfolgswahrsch. $\frac{\| M|\psi\rangle \|^2}{\alpha^2}$

unsere Anwendung $\rightarrow M = A^{-1}$, $|\psi\rangle = |b\rangle$

mit QRAM: multiple $|b\rangle$-preparation möglich

↳ Amplitude amplification möglich

(Drehung um $|b\rangle$) $\Rightarrow$ quadratischen Speedup

hohe Wahrsch. in $\mathcal{O}\left(\frac{\alpha}{\| M|\psi\rangle \|}\right)$ Wiederholungen

·) V -- unitär, einfach impl.

·) falls unitäre Op. U_j einfach implementierbar und Zerlegung $\mathcal{O}(\log N)$-Terme

↳ eff. Impl. von M

·) Query-Komplexität von $U \simeq$ Query-Kompl. des teuersten U_j

für Gate-Komplexität i. A. nicht so gut!

aber: für spez. Anw. hier schon, da $U_j = (e^{-iA})^{t_j}$ ↖ überall gleich

Diagonalisierung $A = T^H D T$

$\Rightarrow T^H D^{-1} T = A^{-1} \overset{!}{\approx} \sum_j \alpha_j e^{-iAt_j}$

$= T^H \sum_j \alpha_j e^{-iDt_j} T$

da $D = \mathrm{diag}(\lambda_j) \Rightarrow$ brauche nur Entwicklung von

$f(x) = \frac{1}{x} \overset{!}{\approx} \sum_j \alpha_j e^{-ixt_j}$

für $x \in \sigma(A)$ bzw. $x \in \Omega \supseteq \sigma(A)$

---

wie bei HHL $\rightsquigarrow$ betrachte nur gut konditionierte EWe ($\geq \frac{1}{\kappa}$), verlange auch $\lambda_{max} = 1$ (skalieren)

$\rightsquigarrow$ suche Approx. an f auf $D_\kappa := [-1,1] \setminus [-\frac{1}{\kappa}, \frac{1}{\kappa}]$

Idee: 1) glätte f um 0

2) Fourier trafo (Integral)

3) Ersetze Integral durch endl. Summe

$\rightarrow$ Funktion h der Form $\sum \alpha_j e^{-ixt_j}$ mit

$\sup_{x \in D_\kappa} |f(x) - h(x)| \leq C\varepsilon$

$\rightarrow$ Vorherige Lemmata $\rightarrow h(A)|b\rangle$ (+Normierung) ist Approx. an $|x\rangle$

Lemma Die Funktion

$$h(x) := \frac{\tilde{\wedge}}{\sqrt{2\pi}} \sum_{j=0}^{J-1} \sum_{k=-K}^{K} \Delta_y \Delta_z \, z_k \, e^{-\frac{z_k^2}{2}} \, e^{-i x y_j z_k}$$

mit $y_j := j\Delta_y$, $z_k := k\Delta_z$ und $J = \Theta\left(\frac{K}{\varepsilon} \log \frac{K}{\varepsilon}\right)$,
$K = \Theta\left(K \log \frac{K}{\varepsilon}\right)$, $\Delta_y = \Theta\left(\frac{\varepsilon}{\sqrt{\log K/\varepsilon}}\right)$, $\Delta_z = \Theta\left(\frac{1}{K\sqrt{\log \frac{K}{\varepsilon}}}\right)$
erfüllt

$$\sup_{x \in D_K} \left| h(x) - \frac{1}{x} \right| \leq c\varepsilon$$

Bew: Sei $g(y) = y e^{-\frac{y^2}{2}}$

Substitution $\Rightarrow \frac{1}{x} = \int_0^\infty g(xy)\,dy$ da $\int_0^\infty g(y)\,dy = 1$

Wahl von $g \Rightarrow$ klingt schnell ab, glatt und

$F(g) = -i g$ also Eigenfkt. von Fourier trafo zu EW $-i$

$$\Rightarrow g = i F(g) = \frac{\tilde{\wedge}}{\sqrt{2\pi}} \int_{\mathbb{R}} z e^{-\frac{z^2}{2}} e^{-iyz}\,dz$$

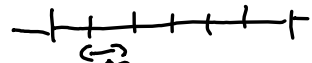
$$\Rightarrow \boxed{\frac{1}{x} = \frac{\tilde{\wedge}}{\sqrt{2\pi}} \int_0^\infty \int_{-\infty}^\infty z e^{-\frac{z^2}{2}} e^{-ixyz}\,dz\,dy}$$

$\rightsquigarrow h(x)$ ist Riemann Summe zu abgeschnittenen Integralen in Formel für $\frac{1}{x}$

1. Schritt: Summation / Integr. in y: in h $\rightsquigarrow$ gesam. Reihe

$$h(x) = \frac{i\Delta_y}{\sqrt{2\pi}} \sum_{k=-K}^{K} \Delta_z \, z_k e^{-\frac{z_k^2}{2}} \frac{1 - e^{-ix\Delta_y z_k}}{1 - e^{-ix\Delta_y z_k}}$$

$$\frac{1}{x} = \frac{1}{\sqrt{2\pi}} \cdot \frac{1}{x} \int_{-\infty}^{\infty} e^{-\frac{z^2}{2}} dz$$



2. Schritt: Approx von $\int$ durch Reihe (∞-Riemann Summe)

$$\left| \frac{1}{x} \left(1 - \underbrace{\frac{1}{\sqrt{2\pi}} \sum_{k=-\infty}^{\infty} \Delta_z e^{-\frac{z_k^2}{2}}}_{\text{Gitterweite}}\right) \right| = \sum_{k=-\infty}^{\infty} e^{-\left(\frac{2\pi k}{\Delta_z}\right)^2/2} = 1 + \sum_{|k|\geq 1} e^{-\left(\frac{2\pi k}{\Delta_z}\right)^2/2}$$

Poisson'sche Summenformel

$$\sum_{k=-\infty}^{\infty} f(k) = \sqrt{2\pi} \sum_{k=-\infty}^{\infty} \hat{f}(2\pi k)$$

$$\Rightarrow \leq \frac{1}{|x|} \cdot 2 \sum_{k=1}^{\infty} e^{-\frac{2\pi^2 k}{\Delta_z^2}} \overset{\text{geom. Reihe}}{=} \frac{1}{|x|} \frac{2}{e^{2\pi^2/\Delta_z^2} - 1}$$

$$\leq C K \cdot \varepsilon e^{-K^2 \log K} \quad \text{nach Wahl von } \Delta_z \text{ und mit } \frac{1}{|x|} \leq K$$

$$\leq \tilde{C} \varepsilon$$

3. Schritt: Reihe abschneiden

$$\Rightarrow \left| \frac{1}{\sqrt{2\pi}x} \cdot \left(\sum_{k=-\infty}^{\infty} \Delta_z \cdot e^{-\frac{z_k^2}{2}} - \sum_{k=-K}^{K} \Delta_z \cdot e^{-\frac{z_k^2}{2}} \left(1 - e^{-ix\gamma_j z_k}\right) \right) \right|$$

$$\leq C\varepsilon$$

folgt aus Dreiecksungl. (für Term $(1-e^{-ix\gamma_j z_k})$ hinzu)

Poisson'sche Summenformel

Abschätzung Restsumme mittels Wahl von K

Langwierig, aber einfach $\rightsquigarrow$ Literatur

4. Schritt:

$$\left| h(x) - \frac{1}{\sqrt{2\pi}x} \sum_{k=-K}^{K} \Delta_z e^{-\frac{z_k^2}{2}} \left(1 - e^{-ix\gamma_j z_k}\right) \right|$$

Formel für h aus 1. Schritt:

$$\hookrightarrow \leq \frac{2}{\sqrt{2\pi}} \left| \sum_{k=-K}^{K} \underbrace{\left(\frac{i\Delta_\gamma z_k}{1 - e^{-ix\Delta_\gamma z_k}} - \frac{1}{x} \right)}_{\leq \Delta_\gamma |z_k| \text{ da } \left| \frac{1}{1-e^{-ix}} - \frac{1}{x} \right| < 1 \text{ für } x \in [-1,1]} \cdot \Delta_z e^{-\frac{z_k^2}{2}} \right|$$

$|z_k| \leq e^{\frac{|z_k|^2}{4}}$

$$\leq \sqrt{\frac{2}{\pi}}\, \Delta_\gamma \underbrace{\sum_{k=-K}^{K} \Delta_z e^{-\frac{z_k^2}{4}}}_{\leq \int_0^\infty e^{-\frac{z^2}{4}} dz}$$

$$= C\,\Delta_\gamma \leq C\varepsilon \qquad \text{up to log-terms}$$

5-Schritt: Kombiniere Schritt 2-4 mit Dreiecksungl.

$$\Rightarrow |h(x) - \tfrac{1}{x}| \leq C \cdot \varepsilon$$ □

Komplexität

Theorem Das q-LGS kann mit $\mathcal{O}(K\sqrt{\log \frac{K}{\varepsilon}})$ Anwendungen von Hamiltonian Sim. für e^{-iHt} mit $t = \mathcal{O}(K \log \frac{K}{\varepsilon})$ mit Genauigkeit $\mathcal{O}\left(\frac{\varepsilon}{K\sqrt{\log \frac{K}{\varepsilon}}}\right)$ gelöst werden.

Die Gate-Komplexität ist

$$\mathcal{O}\left(sK^2 \log^{2.5}\left(\tfrac{K}{\varepsilon}\right)\left(\log N + \log^{2.5}\left(\tfrac{K}{\varepsilon}\right)\right)\right)$$

Beweis → Literatur muss V, U impl.

V in $\mathcal{O}(K \log \frac{K}{\varepsilon})$

U bzw. select $U_{\tilde{A}}$ mit Ham. Simulation

für sparse $\tilde{A}$ $\Leftarrow$ $\mathcal{O}\left(st\left(\log N + \log^{2.5}\left(\tfrac{t}{\varepsilon}\right)\right)\log \tfrac{t}{\varepsilon}\right)$

Wahl von t → gew. Genauigkeit

Ham. Sim. muss $\mathcal{O}(K\sqrt{\log \frac{K}{\varepsilon}})$ durchgef. werden

(α in Lemma vorne) □

Chebyshev approach

Idee: ersetze Fourier-Entwicklung durch Chebyshev Polynome

Problem: sind nicht unitär

Lösg. „Block-encoding" (n+1-qubit block enc.)

Sei: $A \in \mathbb{C}^{2^n \times 2^n}$, $\|A\| \leq 1$ und angenommen, dass

$\exists$ unitäre Matrix $U \in \mathbb{C}^{2^{n+1} \times 2^{n+1}}$ s.d.

$$U = \begin{pmatrix} A & * \\ * & * \end{pmatrix}$$

hier $* \in \mathbb{C}^{2^n \times 2^n}$

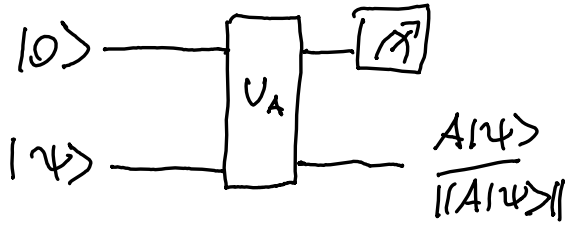
Einträge egal, nur wichtig, dass U unitär

$\rightsquigarrow \forall$ Zustände $|\psi\rangle \rightsquigarrow$ betrachte $|0\rangle|\psi\rangle \simeq \begin{pmatrix} \psi \\ 0 \end{pmatrix}$

$$\Rightarrow U(|0\rangle|\psi\rangle) = |0\rangle A|\psi\rangle + \underbrace{|1\rangle|\phi^\perp\rangle}_{\text{unwichtig}}$$

$\Rightarrow$ messe $|0\rangle$ in 1. qubit $\rightarrow$ Realisierung von $A|\psi\rangle$

Circuit:



Bem. ·) LCU (ohne amplitude amp.) ist Spezialfall von block encoding

·) für generelles A → teuer
für sparse A → effizient durchführbar

⇒ Ziel: block encoding von A^{-1}

Theorem Sei $p \in \mathbb{P}_d$ mit $\|p\|_{\infty,[-1,1]} \leq \frac{1}{4}$ polynom von Grad $\leq d$. Sei U block enc. von A mit $(n+a)$-qubits.
⇒ Ein block encoding von $P(A)$ mit $(n+\mathcal{O}(a))$-qubits kann mittels d-Anwendungen von U, U^{-1}, einer Anwendung von controlled-U und $\mathcal{O}(ad)$ elem. Gates (2-qubit) realisiert werden.

Anwendung von Thm.:

Approx. $f(x) = \frac{1}{x}$ durch $p(x)$:

für $\tilde{d} = \mathcal{O}(\kappa^2 \log \frac{\kappa}{\varepsilon})$

$$\Rightarrow \sup_{x \in D_\kappa} \Big| \underbrace{\frac{1-(1-x^2)^{\tilde{d}}}{x}}_{\in \mathbb{P}_{2\tilde{d}-1}} - \frac{1}{x} \Big| \leq \frac{\varepsilon}{2}$$

-) schreibe $\frac{1-(1-x^2)^{\tilde{d}}}{x} = \sum_{j=0}^{2\tilde{d}-1} \alpha_j T_j$

↳ Cheby. polynome

-) schneide Entwicklung bei $d = \mathcal{O}(K \log \frac{K}{\varepsilon})$ ab
→ Fehler $\leq \frac{\varepsilon}{2}$

⇒ Theorem gibt eff. Implementierung von $P_d(A)$

⇒ Anwendung von block encoding → Zustand $|\tilde{x}\rangle$ mit
$\| |\tilde{x}\rangle - |x\rangle \| \leq C\varepsilon$

gesamt: Gate-komplexität:

$$\mathcal{O}\left(sK^2 \log^2\left(\frac{sK}{\varepsilon}\right)\left(\log N + \log^{2.5}\left(\frac{sK}{\varepsilon}\right)\right)\right)$$

besser als Fourier approach, braucht aber direkt sparse access (funktioniert nicht für allg. Matrizen)

Fourier approach → strukturlosen Hamiltonian simulieren
↳ allgemeiner

Solving linear differential equations

Any linear diff. eq. can be reduced to first order system

$$x'(t) = A(t)x(t) + b(t) \in \mathbb{R}^{N_x}$$

$$x(0) = x_0$$

$A(t) \in \mathbb{R}^{N_x \times N_x}$ is s-sparse, $s \in \mathbb{N}$.

Idea 1: Lie-Trotter approach.

$\Rightarrow$ exponential cost in t

$\Rightarrow$ no inhomogeneous equations

Idea 2: Space-time approach

Feynman's clock: Encode time in basis states $|j\rangle$ and produce output state

$$|\psi\rangle := \sum_{j=0}^{N_t} |j\rangle |x_j\rangle$$

where $x_j \approx x(t_j)$

arXiv: 1010.2745
High-order q. alg. for solving ODEs

Here $N_t = T/\tau$ ... number of timesteps

T ... Final time

τ ... size of timestep.

Problem: Probability of measuring $x(T) \approx x_{N_t}$ is small. Idea: extend ODE beyond T with

$$A(t) := I, \quad b(t) = 0 \quad \forall\, t \in (T, 2T].$$

$\Rightarrow x(t) = x(T) \quad \forall\, t \in [T, 2T]$

$\Rightarrow$ increased probability.

Example: Forward Euler

$$\frac{x_{j+1} - x_j}{\tau} = A(t_j) x_j + b(t_j) \qquad \forall\, t \in [0, 2T]$$

define $\vec{x} := \begin{pmatrix} x_0 \\ x_1 \\ \vdots \\ x_{2N_t} \end{pmatrix}$, $\vec{b} := \begin{pmatrix} x_0 \\ b_0 \tau \\ b_1 \tau \\ \vdots \\ b_{N_t} \tau \\ 0 \\ \vdots \\ 0 \end{pmatrix}$

$$\underline{A} = \begin{pmatrix} I & & & & \\ -(I + A\tau) & I & & 0 & \\ & \ddots & \ddots & & \\ & -(I + A\tau) & I & & \\ & & -I & I & \\ 0 & & & \ddots & \ddots \\ & & & & -I \; I \end{pmatrix} \in \mathbb{R}^{2N_t N_x \times 2N_t N_x}$$

$$\sqrt{\sum_{i=1}^{n} i^2} \simeq \sqrt{n^3} = n^{\frac{3}{2}}$$

$\Rightarrow$ Euler $\Leftrightarrow$ $\underline{A\vec{x} = \vec{b}}$

Informal Analysis: local Euler error $\simeq \tau^2$

$\Rightarrow$ Error at end-time $2N_t \tau^2 \simeq \frac{T^2}{N_t}$

to achieve error $\simeq \varepsilon$, we need $\underline{N_t \simeq \frac{T^2}{\varepsilon}}$

Use HHL Algorithm to solve linear system

$\Rightarrow$ require bounded cond. number

$$\|A^{-1}\| \geq \frac{\|x\|}{\|b\|}$$

Consider $\begin{pmatrix} 1 & & 0 \\ -1 & \ddots & \\ 0 & \ddots & 1 \end{pmatrix} \underbrace{\begin{pmatrix} 1 \\ 2 \\ 3 \\ \vdots \\ n \end{pmatrix}}_{x} = \underbrace{\begin{pmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}}_{b} \Rightarrow = \frac{\sqrt{\sum_{i=1}^{n} i^2}}{\sqrt{\sum_{i=1}^{n} 1}} \simeq \frac{n^{\frac{3}{2}}}{n^{\frac{1}{2}}} \simeq n$

Hence $\kappa \simeq 2 M_+ M_- \simeq T^2$

$\uparrow$ cond. number of $\underline{A}$

$\Rightarrow$ HHL Alg requires at least $\kappa^2 \simeq T^4$ operations

possible improvement:

Multistep methods

$$\sum_{e=0}^{k} \alpha_e x_{j+e} = \tau \sum_{e=0}^{k} \beta_e \left(A(t_{j+e}) x_{j+e} + b(t_{j+e}) \right)$$

Stability of method given by

$$\rho(\xi) = \sum_{j=0}^{k} \alpha_j \xi^j \,, \quad \sigma(\xi) = \sum_{j=0}^{k} \beta_j \xi^j$$

Let $R_j(\mu)$ denote the roots of

$$\rho(\xi) - \mu \sigma(\xi) = 0$$

and define

$$S := \left\{ \mu \in \mathbb{C} \;\middle|\; \begin{array}{l} \text{all roots } R_j(\mu) \text{ satisfy } |R_j(\mu)| \leq 1 \\ \text{multiple roots } R_j(\mu) \text{ satisfy } |R_j(\mu)| < 1 \end{array} \right\}$$

if all roots of σ satisfy $\nearrow$ $\Rightarrow$ method is stable at infinity.

In matrix form, the method reads

$$A_{j,j} = I \qquad 0 \le j < k, \quad N_t < j \le 2N_t$$

$$A_{j,j-1} = -(I + A\tau) \qquad 1 \le j < k$$

$$A_{j,j-k+l} = \alpha_l I - \beta_l A\tau \qquad k \le j \le N_t, \quad 0 \le l \le k$$

$$A_{j,j-1} = -I \qquad N_t < j \le 2N_t$$

$$b_0 = x_0$$

$$b_j = bh \qquad 1 \le j < k$$

$$b_j = \sum_{l=0}^{k} \beta_l bh \qquad k \le j \le N_t$$

$$b_j = 0 \qquad N_t < j \le N_t$$

Assume Oracles: $O_A |j,l\rangle|z\rangle = |j,l\rangle\, z \oplus A_{jl}$ (binary rep.)

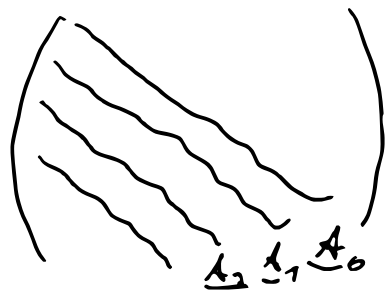
$O_f |j,l\rangle = |j, f(j,l)\rangle$ ↪ lth nonzero in column j

Similar Oracle for ℓ-th nonzero in row j needed.

Lemma There holds $\|\underline{A}\| \leq 1$ if $\tau \leq \frac{1}{\|A\|}$.

Proof We write $\underline{A}$ as sum of block-diags

$$\underline{A} = \sum_{k=0}^{\ell} \underline{A}_k$$



$$\Rightarrow \|\underline{A}_0\| \leq \max\{1, |\alpha_k| + |\beta_k| h \|A\|\}$$

$$\|\underline{A}_1\| \leq \max\{1 + h\|A\|, |\alpha_{k-1}| + |\beta_{k-1}| h \|A\|\}$$

$$\|\underline{A}_\ell\| \leq |\alpha_\ell| + |\beta_\ell| h \|A\| \quad \forall 2 \leq \ell \leq k$$

$$\Rightarrow \|\underline{A}\| \leq \sum_{\ell=0}^{k} \|\underline{A}_k\| \leq k \leq 1. \quad \square$$

Lemma Assume that $A = VDV^{-1}$ with eigenvalues λ_i s.t. $|\arg(-\lambda_i)| \le \alpha$.
Assume the multistep method is $A(\alpha)$-stable
$(S \supseteq \{\lambda \in \mathbb{C} \mid |\arg(-\lambda)| < \alpha, \lambda \neq 0\})$.
Then, $\|\underline{A}^{-1}\| \le N_t K_V$, where $K_V := \|V\|\|V^{-1}\|$

Proof Let $\underline{V}$ denote the block-diag matrix $\underline{V} = \begin{pmatrix} V & & \\ & V & \\ & & \ddots \\ & & & V \end{pmatrix}$ and $\underline{D}$ the matrix $\underline{A}$ where we replace A with D. Then

$$\underline{A} = \underline{V}\,\underline{D}\,\underline{V}^{-1} \quad \text{and} \quad \|\underline{A}^{-1}\| \le K_V \|\underline{D}^{-1}\|.$$

It remains to estimate $\|\underline{D}^{-1}\|$.

$$\underline{D} = \sum_{\ell=0}^{k} \underline{D}_\ell$$ with (off-diagonal) block-matrices $\underline{D}_\ell$ $\begin{pmatrix} \ddots & & \\ & \ddots & \\ & & 0 \end{pmatrix}$

Case $\ell = 1$:

$$\underline{D}^{-1} = (\underline{D}_0 + \underline{D}_1)^{-1} = \underline{D}_0^{-1}(I + \underline{D}_1 \underline{D}_0^{-1})^{-1}$$

$$= \underline{D}_0^{-1} \sum_{k=0}^{\infty} (-\underline{D}_1 \underline{D}_0^{-1})^k$$

$\underline{D}_1 \underline{D}_0^{-1}$ is of form $\begin{pmatrix} 0 & & * \\ & \ddots & \\ * & & 0 \end{pmatrix}$

$$\Rightarrow (\underline{D}_1 \underline{D}_0^{-1})^k = 0 \quad \forall k \geq 2N_t$$

The off-diagonal entries have the form

$$(1 + C\tau)^k \leq 1 \quad \forall k \leq \frac{1}{\tau} = N_t$$

$$\Rightarrow \underline{\|\underline{D}^{-1}\| \leq N_t}$$

Case $\ell > 1$: (sketch) $\underline{D}\, y = v$ corresponds to the discretization of the system

$$y^{(l)}(t) = \lambda_j\, y^{(l)}(t) + y^{(l-1)}(t)$$

Since the method is α-stable, the numerical approximations $y_i^{(j)}$ can not grow unless forced by $r^{(j)}$

Let $\left(y_i^{(j,k)}\right)_{i=1}^{N_t}$ denote the solution with rhs $\left(r_i^{(j)} \delta_{ik}\right)_{i=1}^{N_t}$ and initial cond $x_0 \delta_{k0}$

$$\Rightarrow \quad y_i^{(j)} = \sum_{k=0}^{N_t} y_i^{(j,k)}$$

Stability shows $|y_i^{(j,k)}| \lesssim |r_k^{(j)}| \quad \forall i \geq k$

$$\Rightarrow \|y^{(j)}\| \lesssim \sum_{k=0}^{N_t} \sqrt{(N_t - k)}\, |r_k^{(j)}|$$

$$\lesssim N_t \underbrace{\sqrt{\sum_{k=0}^{N_t} |r_k^{(j)}|^2}}_{\|r^{(j)}\|}$$

$$\Rightarrow \|y\| \lesssim N_t \|r\|$$

$\square$

Theorem Under the above assumptions the HHL Alg produces a state proportional to

$$|\psi\rangle = \sum_{j=0}^{N_x} |j\rangle |x_j\rangle$$

with error ε in

$$O\left(\log N_x \, s^{9/2} (\|A\| T)^{2+\frac{2}{p}} \kappa_V^5 \left(\|x_0\| + \frac{\|b\|}{\|A\|} \right) \varepsilon^{-2} \right)$$

calls to the oracles for A, b, and x_0

---

arXiv:1701.03684 (Berry, Childs, Ostrander, Wang)

Theorem Suppose $A = V D V^{-1}$ with $\mathrm{Re}\, D \le 0$. Assume $A(t) = A$ and $b(t) = b$. There exists q-alg which produces

$$\frac{x(T)}{\|x(T)\|}$$

up to error ε with

$$O\left(\kappa_V \, s \, q \, T \|A\| \, \mathrm{poly}\left(\log\left(\kappa_V \, s \, q \, \beta T \frac{\|A\|}{\varepsilon} \right) \right) \right)$$

- $q \uparrow$: $\max_{t \in [0,T]} \frac{\|x(t)\|}{\|x(T)\|}$
- $\beta \uparrow$: $\frac{\|x_0\| + T\|b\|}{\|x(T)\|}$

Query calls.