- Certifying randomness (lect 15)

What can we say from the Bell-CHSH inequality?:

(winning prob. from q-scheme) > 75% = (opt. win prob. from classical scheme)

$$= \max\left(\mathbb{P}[\text{win}] = \mathbb{E}\left[\sum_{x,y}\frac{1}{8}\left(X_x Y_y (-1)^{xy} + 1\right)\right]\right)$$

- treat the computation $(x,y) \rightsquigarrow (a,b)$ as a black box.
- supply random inputs (x_i, y_i) $0 \le i < N$
- then we will "see" that the outputs (a_i, b_i) win the challenge better than any deterministic function $a(x,r)$, $b(y,r)$ r: auxiliary param

$\rightsquigarrow$ we can say that this black box can be use as a random number generator:

$(x_0, y_0), \cdots, (x_{N-1}, y_{N-1}) \rightsquigarrow \underbrace{a_0, \cdots, a_{N-1}}$

has randomness better than any seq. of the form $a(x_i, r)$.

- Quantum query complexity

How we measure complexity of algorithms.

1. Circuit complexity

Given a unitary transform U, estimate the number of basic building blocks V_i (e.g. CNOT, Pauli-X/Y/Z, Hadamard, ...) to $U = V_0 \cdots V_{N-1}$. (this factorization is not unique)

Problem: this is difficult to estimate.

2. Query complexity

Classical model = a "computation" is a function $f: \{0,1\}^n \to \{0,1\}$

(cont.) a "query" to input $x \in \{0,1\}^n$: $0 \le i < n$

$\rightsquigarrow$ "answer" x_i

query complexity = number of queries we need to determine some given prop. of f (on all vals)

i.e. : $f(x_0, x_1, x_2) = (x_0 + x_1)$

(if $x_0 = 0$, return x_1, else return x_2)

$\rightsquigarrow$ two queries are enough to det. all vals.

Quantum model

model "query" and "answer" (oracle) as

a unitary transform U_f for the

oracle func. $f : \{0,1\}^n \to \{0,1\}$

- XOR oracle $U_f : |x\,y\rangle \mapsto |x\,(y \oplus f(x))\rangle$

$x \in \{0,1\}^n$, $y = 0, 1$. ($\oplus$: XOR; $0 \oplus 1 = 1$, $1 \oplus 1 = 0$, ...)

- phase oracle $U'_f : |x\,y\rangle \mapsto (-1)^{f(x)y} |x\,y\rangle$

$\rightsquigarrow$ estimate N s.t. $V_{N-1}\, U_f\, V_{N-2}\, U_f \cdots V_1\, U_f\, V_0$

for fixed V_i that will determine f (some prop. of)

- Deutsch's algorithm (§17.3)

setup : $n = 1$ so $f : \{0,1\} \to \{0,1\}$

we want to determine if $f(0) = f(1)$ or not

classically : we need to query both $f(0)$ and $f(1)$

D-alg. : we use a variant of ph. query

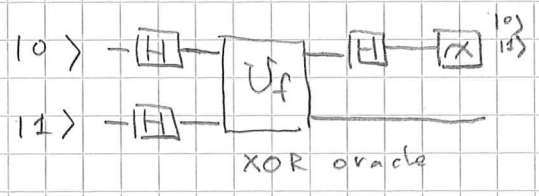
$U'_f : |x\rangle \to (-1)^{f(x)} |x\rangle$

Then $V_0 = H$ (Hadamard gate.) ; $U'_f V_0$ gives

$$|0\rangle \xrightarrow{H} |+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \mapsto \begin{cases} (-1)^{f(0)} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) & f(1) = f(0) \\ (-1)^{f(0)} \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) & f(1) \neq f(0) \end{cases}$$

then measurement in $|+\rangle, |-\rangle$ basis will tell which case

- How to implement this ?



Comparison of XOR query and phase query

$$U_f : |x\,y\rangle = |x\,(y \oplus f(x))\rangle$$

$$U_f' : |x\,y\rangle = (-1)^{f(x)y}\,|x\,y\rangle$$

$$U_f\,|x\rangle \otimes |0\rangle = \begin{cases} |x\rangle \otimes |0\rangle & f(x)=0 \\ |x\rangle \otimes |1\rangle & f(x)=1 \end{cases}$$

$$|x\rangle \otimes |1\rangle = \begin{cases} |x\rangle \otimes |1\rangle & f(x)=0 \\ |x\rangle \otimes |0\rangle & f(x)=1 \end{cases}$$

$$U_f\,|x\rangle \otimes |+\rangle = \begin{cases} |x\rangle \otimes |+\rangle & f(x)=0 \\ |x\rangle \otimes |-\rangle & f(x)=1 \end{cases}$$

$$|x\rangle \otimes |-\rangle = \begin{cases} |x\rangle \otimes |-\rangle & f(x)=0 \\ |x\rangle \otimes |+\rangle & f(x)=1 \end{cases}$$

So $I_{2^N} \otimes H$ conjugates U_f and U_f'

$$(I_{2^N} \otimes H)\,U_f\,(I_{2^N} \otimes H^+) = U_f'$$

$$(I_{2^N} \otimes H^+)\,U_f'\,(I_{2^N} \otimes H) = U_f$$

(note $H^+ = H$)