A quantum algorithm for Subset sum

Sumeet Shirgure

The subset sum problem

Given a collection A of n positive integers {a1, a2 ...}, find a sub-collection with a target sum t.

- Known to be NP complete

Classical methods

Classical methods can be categorized into two kinds :

- Exponential methods : exhaustive search, + some meet in the middle tricks : O(n*sqrt(2^n)), require anywhere between O(n) to O(sqrt(2^n)) memory
- Pseudo-polynomial time dynamic programming : runtime depends on the sum of values in S – related FPTAS

A quantum algorithm via Grover Search

- Reduce to unstructured search : “Given n-bit strings, find the ones that satisfy a predicate f”
- Grover’s algorithm gives us a procedure to find the feasible strings, provided we have an appropriate oracle:
  - $U|x\rangle \longrightarrow (-1)^{f(x)}|x\rangle$

The subset sum oracle

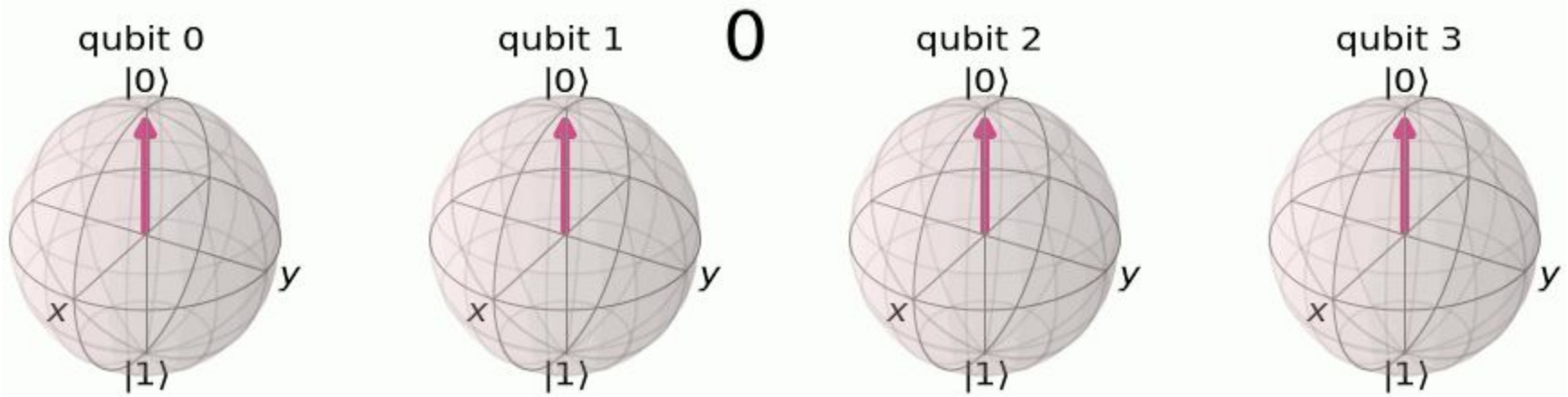
The objective is therefore to construct an oracle that takes a sub-collection σ = <σ1,σ2,...σn>, and negates the phase only when the respective sum s = Σ σi.ai equals t.

To do that, we first construct a circuit with two registers $|\sigma\rangle|\tau\rangle$, and an adder A that acts as

$A|\sigma\rangle|\tau\rangle \longrightarrow |\sigma\rangle|(\tau+\Sigma\ \sigma i.ai) \bmod 2^k\rangle$ for all σ,τ

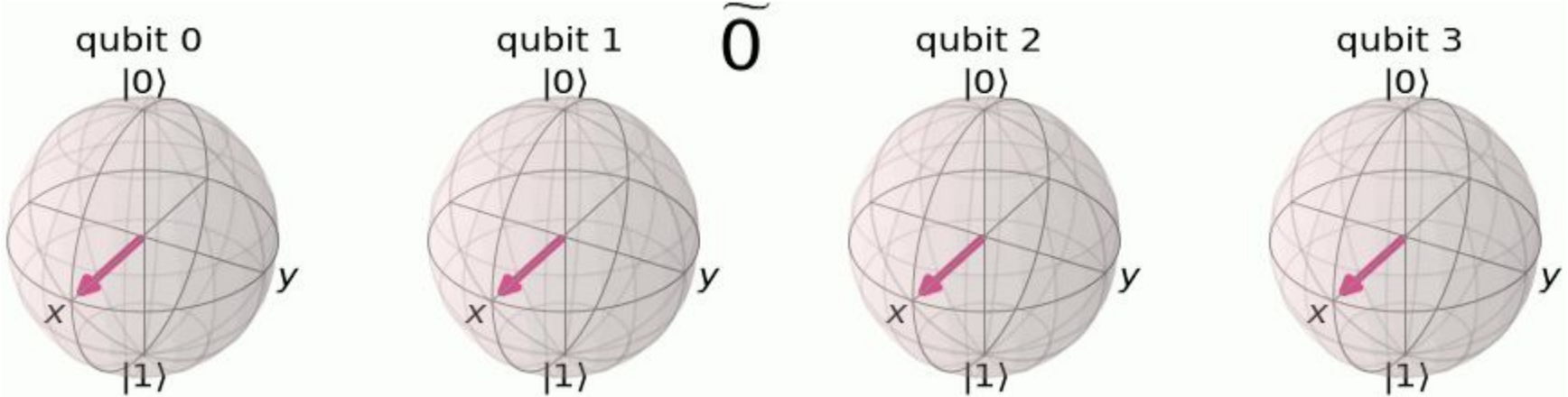
The quantum fourier transform

QFT maps computational (z) basis to the Fourier basis.



The quantum fourier transform

QFT is an eigenbasis transform that maps qubits to the Fourier basis.



QFT based "Multi-adder"

Addition in z-basis is rotation in F-basis.

To add ai, we first take the Fourier transform of 0, then apply the respective rotations in F-basis, and take an inverse transform.

These rotations are exactly the controlled phase gates with successively halved frequencies.

$|\sigma\rangle\,|0\rangle$—[F]—→ $|\sigma\rangle\,|0\sim\rangle$—[CPhase]—→ $|\sigma\rangle\,|s\sim\rangle$—[F†]—→ $|\sigma\rangle\,|s\rangle$

The subset sum oracle

Once we have an adder, we simply apply it on a zero-initialized register, and check if it has the desired value t. If so, we flip an additional qubit.

$|\sigma\rangle|0\rangle|z\rangle\text{—[A]—→}|\sigma\rangle|s\rangle|z\rangle\text{—[C]—→}|\sigma\rangle|s\rangle|z\oplus[s=t]\rangle$

Finally, uncompute to reuse the accumulator register

$\text{—[A†]—→}|\sigma\rangle|0\rangle|z\oplus[s=t]\rangle$

U=A†CA is almost the oracle we need.

One last trick...

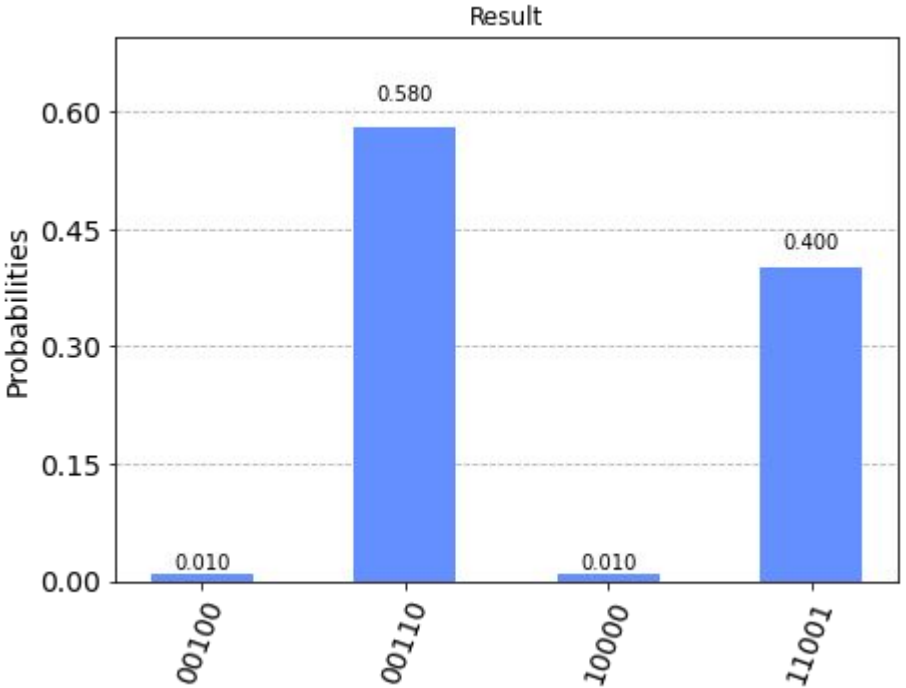
Any binary oracle of the kind $U|x\rangle|z\rangle \longrightarrow |x\rangle|z \oplus h(x)\rangle$ can be converted into a phase flip oracle by initializing the second qubit to the Bell state $(|0\rangle - |1\rangle)/\sqrt{2}$

Simulations

For a random instance :

A = [1,9,8,7,5], t = 15

We get the statistics on the right.

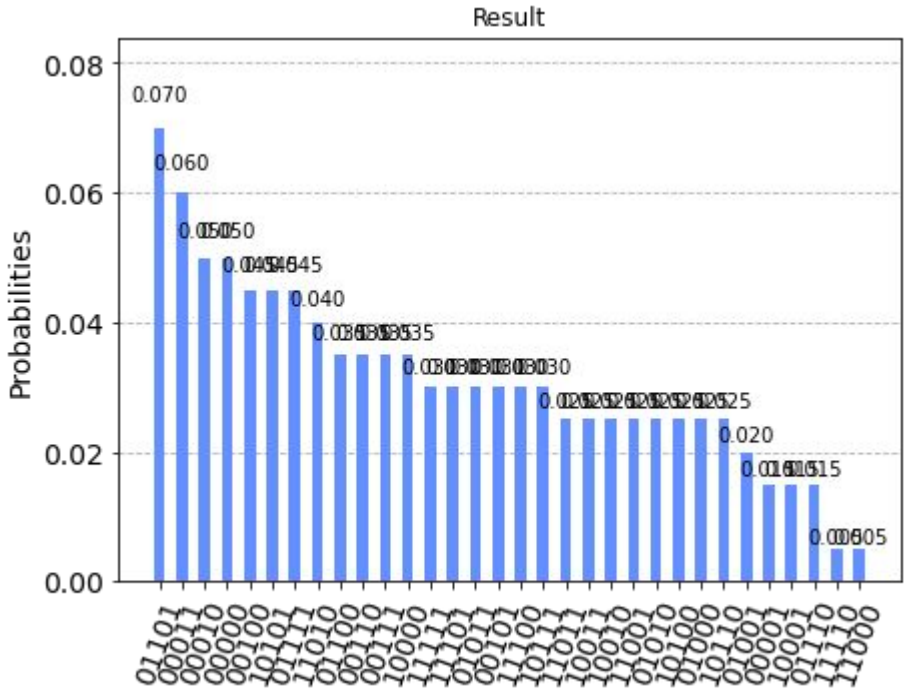


Physical implementation on IONQ

During the hackathon, this circuit was dispatched to IONQ's trapped ion quantum computer.

Here are the results :

- Mode is a feasible set!
- Distribution looks funny



Exercises

How can one construct a qubit-flipping oracle out of a phase flip oracle? (Hint: think phase estimation)

What would change if we’re trying to solve the “subset sum is d-close to t” problem?