Quantum Computation

Finding the period of a function
(Simon's algorithm and its generalisation)

Luís Soares Barbosa & Renato Neves

MSc Physics Engineering

Universidade do Minho, 2024-25

Simon's problem

The problem

Let $f : 2^n \longrightarrow 2^n$ be such that for some $s \in 2^n$,

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

Find s.

Exercise

What characterises f if $s = 0$? And if $s \neq 0$?

Simon's problem

Exercise

- f is bijective if $s = 0$, because $y \oplus 0 = 0$.
- f is two-to-one otherwise ,because, for a given s there is only a pair of values x, y such that $x \oplus y = s$.

Let us assume f to be two-to-one, and rewrite the problem as follows:

Equivalent formulation as a period-finding problem

Determine the period s of a function f periodic under $\oplus$:

$$f(x \oplus s) \;=\; f(x)$$

Simon's problem

Example

Let $f : 2^3 \longrightarrow 2^3$ be defined as

x	$f(x)$
000	101
001	010
010	000
011	110
100	000
101	110
110	101
111	010

Cleary $s = 110$. Indeed, every output of f occurs twice, and the bitwise XOR of the corresponding inputs gives s.

Simon's problem, classically

Compute f for sequence of values until finding a value x_j such that $f(x_j) = f(x_i)$ for a previous x_i, i.e. a colision. Then

$$x_j \oplus x_i \;=\; x_i \oplus (x_i \oplus s) \;=\; s$$

- Since f is two-to-one, after collecting 2^{n-1} evaluations with no collisions, the next evaluation must cause a collision.
- So in the worst case $2^{n-1} + 1$ evaluations are needed.

Simon's problem, classically

Can we do better?

Actually, some problems for which there is a quantum exponential advantage, admit classical probabilisitic interesting solutions, e.g.
To solve Deutsch-Josza with some margin of error evaluate two arbitrary inputs x and y,

- $f(x) = f(y) \implies$ constant
- $f(x) \neq f(y) \implies$ balanced

Probability of giving the right answer?

- f is constant $\implies$ right answer with probability 1
- f is balanced $\implies$ right answer with probability $\frac{2^{n-1}}{2^n} = \frac{1}{2}$

Simon's problem, classically

which can still be improved:

Tackling Deutsch-Josza with Probabilities

To solve the problem with some margin of error evaluate k arbitrary inputs $x_1, \ldots, x_k$,

- output always the same $\Longrightarrow$ constant
- otherwise $\Longrightarrow$ balanced

Probability of giving the right answer?

- f is constant $\Longrightarrow$ right answer with probability 1
- f is balanced $\Longrightarrow$ right answer with probability ...

$$1 - \left(\frac{2^{n-1}}{2^n}\right)^k = 1 - \frac{1}{2^k}$$

Probability of observing the same output in k tries

Simon's problem, classically

Actually, some problems for which there is a quantum exponential advantage, admit classical probabilisitic interesting solutions, e.g.

Deutsch-Joza

- Classical deterministic: requires $2^{n-1} + 1$ queries in the worst case,
- Classical probabilisitic: requires 2 queries with a probability of error at most $\frac{1}{3}$ (i.e. $1\frac{1}{2} + \frac{1}{2} * \frac{1}{2}$)
- Quantum: requires 1 query.

However, for the Simon's problem an exponential number of queries to the oracle accessing f are required by any classical probabilisitic algorithm.

Simon's problem, classically

Compute f for sequence of values until finding a value x_j such that $f(x_j) = f(x_i)$ for a previous x_i, i.e. a colision. Then

$$x_j \oplus x_i \;=\; x_i \oplus (x_i \oplus s) \;=\; s$$

How many evaluations do we need to have a collision with probability p?

To have a collision with probability $p = \frac{1}{k} \leq \frac{1}{2}$ we need

$$\approx \sqrt{(2 \cdot 2^n) \cdot p} = \sqrt{\frac{2}{k} \cdot 2^n} = \sqrt{\frac{2}{k}} \cdot 2^{\frac{n}{2}} \quad \text{evaluations}$$

See the Birthday's problem

But a quantum algorithm solves the problem in polynomial time with probability $\approx \frac{1}{4}$

Note: The birthday problem

Seeks to determine the probability that, in a set of n randomly chosen people, at least two will share a birthday.

$n = 23$ leads to $p(n) \approx 0.5$

Let the universe be $U = 365$ (days) and $n = 23$.
U^n is the space of birthdays and $V = \frac{U!}{(U-n)!}$ (n permutations of U) the number of birthdays with no repetitions.
Then,

$$p(n) \; = \; 1 - \frac{V}{U^n} \; \approx \; 1 - 0.493 \; \approx \; 0.507$$

Heuristic for cases leading with $p(n) \leq 0.5$

$$p(n) \; \approx \; \frac{n^2}{U} \quad \Rightarrow \quad n \; \approx \; \sqrt{2U * p(n)}$$

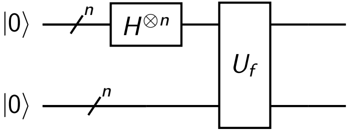
which yields for $p(n) = 0.5$, $n \approx 19$.

Simon's algorithm: The key steps

1. Prepare a superposition $\frac{1}{\sqrt{2}}(|x\rangle + |x \oplus s\rangle)$ for some string x
2. Use interference to find s (indeed, to extract a string y s.t. $y \cdot s = 0$)
3. Repeat previous steps a sufficient number of times to obtain system of equations s.t. $y_k \cdot s = 0$
4. Solve the system for s using Gaussian elimination

Complexity n^3

Simon's algorithm: Preparing the superposition



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

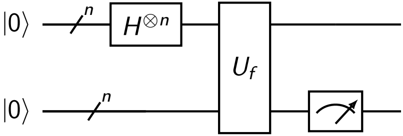
$$U_f(H^{\otimes n} \otimes I)\,|0\rangle\,|0\rangle \;=\; U_f\left(\frac{1}{\sqrt{2^n}} \sum_{x \in 2^n} |x\rangle\,|0\rangle\right) \;=\; \frac{1}{\sqrt{2^n}} \sum_{x \in 2^n} |x\rangle\,|f(x)\rangle$$

The state after the oracle can be rewritten as

$$\frac{1}{\sqrt{2^{n-1}}} \sum_{x \in P} \frac{1}{\sqrt{2}} (|x\rangle + |x \oplus s\rangle)|f(x)\rangle \tag{1}$$

Set P is composed of one representative of each of the 2^{n-1} sets of strings $\{x, x \oplus s\}$, into which 2^n can be partitioned.

Simon's Algorithm: Preparing the superposition



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

If the result of measuring the bottom qubits is $f(x)$, then the top ones will contain superposition

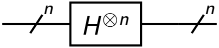
$$\frac{1}{\sqrt{2}}(|x\rangle + |x \oplus s\rangle)$$

as they are the unique values yielding $f(x)$.

i.e. a measurement of the bottom qubits chooses randomly one of the 2^{n-1} possible outcomes of f ...

as f gives the same output for x and $x \oplus s$, to 2^n possible inputs correspond 2^{n-1} possible outcomes.

Simon's Algorithm: Interference to find s



Recall

$$H|x\rangle \;=\; \frac{1}{\sqrt{2}} \sum_{z \in 2} (-1)^{xz} |z\rangle$$

Exercise

Show this extends to a n-qubit as follows

$$\begin{aligned} H^{\otimes n}|x\rangle \;&=\; H|x_1\rangle H|x_2\rangle \cdots H|x_n\rangle \\ &=\; \frac{1}{\sqrt{2^n}} \sum_{z \in 2^n} (-1)^{x \cdot z} |z\rangle \end{aligned}$$

where $x.z$ denotes the bitwise product of x and z, modulo 2, or bitwise conjunction. Conjunction is denoted by juxtaposition.

Simon's Algorithm: Interference to find s

$$
\begin{aligned}
H^{\otimes n}|x\rangle &= H|x_1\rangle H|x_2\rangle \cdots H|x_n\rangle \\
&= \frac{1}{\sqrt{2}} \sum_{z_1 \in 2} (-1)^{x_1 z_1} |z_1\rangle + \frac{1}{\sqrt{2}} \sum_{z_2 \in 2} (-1)^{x_2 z_2} |z\rangle \cdots \frac{1}{\sqrt{2}} \sum_{z_n \in 2} (-1)^{x_n z_n} |z_n\rangle \\
&= \frac{1}{\sqrt{2^n}} \sum_{z_1, z_2, \cdots, z_n \in 2} (-1)^{x_1 z_1 + x_2 z_2 + \cdots + x_n z_n} |z_1 z_2 \cdots z_n\rangle \\
&= \frac{1}{\sqrt{2^n}} \sum_{z \in 2^n} (-1)^{x \cdot z} |z\rangle
\end{aligned}
$$

Simon's Algorithm: Interference to find s

$$H^{\otimes n} \otimes I \left(\frac{1}{\sqrt{2}} (|x\rangle + |x \oplus s\rangle) |f(x)\rangle \right)$$

$$= \frac{1}{\sqrt{2^n}} \sum_{z \in 2^n} \frac{1}{\sqrt{2}} ((-1)^{x \cdot z} + (-1)^{(x \oplus s) \cdot z}) |z\rangle |f(x)\rangle$$

$$= \frac{1}{\sqrt{2^n}} \sum_{z \in 2^n} \frac{1}{\sqrt{2}} ((-1)^{x \cdot z} + (-1)^{(x \cdot z) \oplus (s \cdot z}) |z\rangle |f(x)\rangle$$

$$= \frac{1}{\sqrt{2^n}} \sum_{z \in 2^n} \frac{1}{\sqrt{2}} ((-1)^{x \cdot z} + (-1)^{(x \cdot z)} (-1)^{(x \cdot z)} |z\rangle |f(x)\rangle$$

$$= \frac{1}{\sqrt{2^{n+1}}} \sum_{z \in 2^n} \underbrace{(-1)^{x \cdot z} (1 + (-1)^{s \cdot z})}_{(\star)} |z\rangle |f(x)\rangle$$

Simon's Algorithm: Interference to find s

$$\frac{1}{\sqrt{2^{n+1}}} \sum_{z \in 2^n} \underbrace{(-1)^{x \cdot z}(1 + (-1)^{s \cdot z})}_{(\star)} |z\rangle |f(x)\rangle$$

- $s \cdot z = 1 \Rightarrow (\star) = 0$ and the corresponding basis state $|z\rangle$ vanishes
- $s \cdot z = 0 \Rightarrow (\star) \neq 0$: and the corresponding basis state $|z\rangle$ is kept. In this case the probability of geting z upon measurement is $\frac{1}{2^{n-1}}$ (why?)

Simon's Algorithm: Interference to find s

This state can be presented as a uniform superposition as follows:

$$
\begin{aligned}
& \frac{1}{\sqrt{2^{n+1}}} \sum_{z \in 2^n} (-1)^{x \cdot z}(1 + (-1)^{s \cdot z})\, |z\rangle|f(x)\rangle \\
= & \frac{1}{\sqrt{2^{n+1}}} \sum_{z \in S^{\perp}} 2(-1)^{x \cdot z}\, |z\rangle|f(x)\rangle \\
= & \frac{1}{\sqrt{2^{n-1}}} \sum_{z \in S^{\perp}} (-1)^{x \cdot z}\, |z\rangle|f(x)\rangle
\end{aligned}
$$

where $S^{\perp}$, for $S = \{0, s\}$ is the orthogonal complement of subspace S, with $\dim(S^{\perp}) = n - 1$
(because $\dim(S) = 1$, as S is the subspace generated by s)

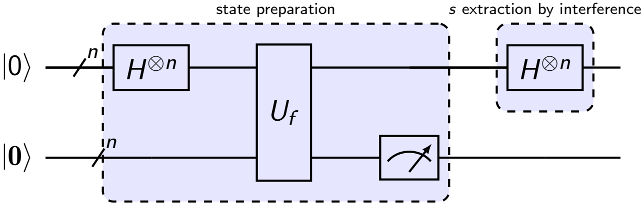
S and $S^{\perp}$

Both are subspaces of the vector space Z_2^n whose vectors are strings of length n over $Z_2 = \{0, 1\}$.

- The dimension of Z_2^n is n; a basis is provided by strings with exactly one 1 in the kth position (for $k = 1, 2, \cdots, n$).
- Two vectors v, u in Z_2^n are orthogonal iff $v \cdot u = 0$ (operation $\cdot$ acts as the internal product).
- Thus, for any subspace F of Z_2^n, $F^{\perp} = \{u \in Z_2^n \mid \forall_{v \in F}.\ \ u \cdot v = 0\}$

Warning: to not confuse with the Hilbert space in which the algorithm is executed and whose basis are labelled by elements of Z_2^n.

Simon's algorithm: The circuit



Simon's Algorithm: Computing s

Running this circuit and measuring the control register results in some z in $(Z_2)^n$ satisfying

$$s \cdot z \;=\; 0\,,$$

the distribution being uniform over all the strings that satisfy this constraint.

Exercise

In the previous discussion we assumed that $s \neq 0$. Show that the conclusion above is still valid if $s = 0$.

Simon's Algorithm: Computing s

Thus, it is enough to repeat this procedure until $n-1$ linearly independent values $\{z_1, z_2. \cdots, z_{n-1}\}$ are found, and solve the following set of $n-1$ equations in n unknowns (corresponding to the bits of s):

$$
\begin{aligned}
z_1 \cdot s &= 0 \\
z_2 \cdot s &= 0 \\
&\vdots \\
z_{n-1} \cdot s &= 0
\end{aligned}
$$

to determine s. Actually,

$\mathrm{span}\{z_1, z_2, \cdots, z_{n-1}\} = S^\perp$ and $\{z_1, z_2, \cdots, z_{n-1}\}$ forms a base for $S^\perp$

Thus, s is the unique non-zero solution of

$$Z\, s = 0$$

where Z is the matrix whose line i corresponds to vector z_i.

Simon's Algorithm: Computing s

What is the probability of obtaining such a system of equations by running the circuit $n - 1$ times?

Simon's slgorithm: Probability of success

Exercise

If $s \neq 0$ then for half of the inputs y we have $y \cdot s = 0$ and for the other half $y \cdot s = 1$

#	Possibilities of failure at each step	Probability of failure
1	$\{0\}$	$\frac{2^0}{2^{n-1}}$
2	$\{0, y_1\}$	$\frac{2^1}{2^{n-1}}$
3	$\{0, y_1, y_2, y_1 \oplus y_2\}$	$\frac{2^2}{2^{n-1}}$
...	...	...
$n-1$	$\{0, y_1, y_2, y_3 \dots\}$	$\frac{2^{n-2}}{2^{n-1}}$

Simon's slgorithm: Probability of success

#	Possibilities of failure at each step	Probability of failure
1	$\{0\}$	$\frac{2^0}{2^{n-1}}$
2	$\{0, y_1\}$	$\frac{2^1}{2^{n-1}}$
3	$\{0, y_1, y_2, y_1 \oplus y_2\}$	$\frac{2^2}{2^{n-1}}$
...	...	...
$n-1$	$\{0, y_1, y_2, y_3 \dots\}$	$\frac{2^{n-2}}{2^{n-1}}$

Table yields the sequence of probabilities of failure,

$$\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \dots, \frac{1}{2^{n-1}} \qquad \text{(from bottom to top)}$$

Probability of failing in the first $n-2$ steps is thus

$$\frac{1}{4} + \frac{1}{8} + \dots = \frac{1}{4}\left(1 + \frac{1}{2} + \dots\right) \leq \frac{1}{4} \cdot \Big(\underbrace{\textstyle\sum_{i \in \mathbb{N}} \frac{1}{2^i}}_{\text{Geometric series whose sum is equal to two}} \Big) = \frac{1}{2}$$

Simon's algorithm: Probability of success

- Probability of succeeding in the first $n-2$ steps at least $\frac{1}{2}$
- Probability of succeeding in the $(n-1)$-th step is $\frac{1}{2}$
- Thus probability of succeeding in all $n-1$ steps at least $\frac{1}{4}$

More advanced maths tell that the probability is slightly higher (around $0.28878\ldots$)

The algorithm

1. Prepare the initial state $\frac{1}{\sqrt{2^n}} \sum_{x \in 2^n} |x\rangle|0\rangle$ and make $i := 1$
2. Apply the oracle U_f to obtain the state

$$\frac{1}{\sqrt{2^n}} \sum_{x \in 2^n} |x\rangle|f(x)\rangle$$

which can be re-written as

$$\frac{1}{\sqrt{2^{n-1}}} \sum_{x \in P} \frac{1}{\sqrt{2}}(|x\rangle + |x \oplus s\rangle)|f(x)\rangle$$

and measure the bottom qubits not strictly necessary but makes the analysis simpler.

3. Apply $H^{\otimes n}$ to the top qubits yielding a uniform superposition of elements of $S^{\perp}$.

The algorithm

4. Measure the first register and record the value observed z_i, which is a randomly selected element of $S^\perp$.
5. If the dimension of the span of $\{z_1, z_2, \cdots, z_i\}$ is less than $n-1$, increment i and to go step 2; else proceed.
6. Compute s as the unique non-zero solution of

$$Z\,s \;=\; 0$$

The crucial observation is that the set of observed values must form a basis to $S^\perp$.

Going general ...

The problem

Let $f : 2^n \longrightarrow X$, for some X finite, be such that,

$$f(x) = f(y) \text{ iff } x - y \in S$$

for some subspace S of Z_2^n with dimension m.

Find a basis $\{s_1, s_2, \cdots s_m\}$ for S.

In Simon's problem

- $x = y \oplus s$, i.e. $x - y = s$.
- s is a basis for the space S generated by $\{s\}$.