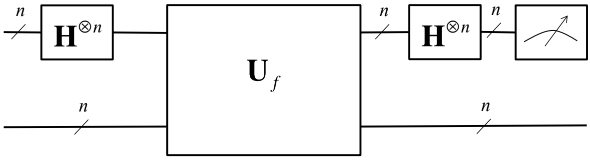
Simon's Periodicity Algorithm



PURPOSE: Detect the Period of a Function

1

Simon's Periodicity Algorithm Problem Overview

- Consider a Function of Interest of the Form:

$$f : \{0,1\}^n \rightarrow \{0,1\}^n$$

- Function is "Encoded" in an Oracle
- Compute a string $\mathbf{c}$,

$$\mathbf{c} = c_0 c_1 c_2 \ldots c_{n-1}$$

Such that for all strings: $\mathbf{x}, \mathbf{y} \in \{0,1\}^n$

$f(\mathbf{x}) = f(\mathbf{y})$ if and only if $\mathbf{x} = \mathbf{y} \oplus \mathbf{c}$

2

Simon's Periodicity Algorithm Problem Overview

$f(\mathbf{x}) = f(\mathbf{y})$ if and only if $\mathbf{x} = \mathbf{y} \oplus \mathbf{c}$

- XOR ($\oplus$) operation is performed bitwise on the strings $\mathbf{y}$ and $\mathbf{c}$
- The values of f repeat themselves with respect to a Bitstring pattern $\mathbf{c}$
- $\mathbf{c}$ is the "Period" of f
- Purpose of Simon's Algorithm is to determine $\mathbf{c}$ (the "Period")

3

Reminder of the XOR Operator

x	y	$x \oplus y$
0	0	0
0	1	1
1	0	1
1	1	0

- $\oplus$, Denotes Addition Modulo-2: $(x + y)(\text{mod } 2) = x +_2 y = x \oplus y$
- $\oplus$, Referred to as the "Exclusive-OR" or XOR is Addition Modulo-2 When the Digit Set is Restricted to $\{0,1\}$

- XOR ($\oplus$) operation is performed bitwise on the n-bit strings $\mathbf{y}$ and $\mathbf{c}$
- EXAMPLES ($n = 3$):

$$x = 011, y = 010: x \oplus y = 001$$
$$x = 101, y = 110: x \oplus y = 011$$
$$x = 111, y = 010: x \oplus y = 101$$ x inverts y
$$x = 000, y = 010: x \oplus y = 010$$ x creates identity for y

4

Periodic Function Example

- number of bits n=3
- Consider $\mathbf{c}$=101

y **c** **x**

$$
\begin{aligned}
000 \oplus 101 &= 101 \Rightarrow f(000) = f(101)\\
001 \oplus 101 &= 100 \Rightarrow f(001) = f(100)\\
010 \oplus 101 &= 111 \Rightarrow f(010) = f(111)\\
011 \oplus 101 &= 110 \Rightarrow f(011) = f(110)\\
100 \oplus 101 &= 001 \Rightarrow f(100) = f(001)\\
101 \oplus 101 &= 000 \Rightarrow f(101) = f(000)\\
110 \oplus 101 &= 011 \Rightarrow f(110) = f(011)\\
111 \oplus 101 &= 010 \Rightarrow f(111) = f(010)
\end{aligned}
$$

this must hold if $f(\mathbf{x})=f(\mathbf{y})$ over $\mathbf{c}$, the period of f

if $\mathbf{c}=0^n$, what does that imply about f

5

Periodic Function Example

- number of bits n=3
- Consider $\mathbf{c}$=101

y **c** **x**

$$
\begin{aligned}
000 \oplus 101 &= 101 \Rightarrow f(000) = f(101)\\
001 \oplus 101 &= 100 \Rightarrow f(001) = f(100)\\
010 \oplus 101 &= 111 \Rightarrow f(010) = f(111)\\
011 \oplus 101 &= 110 \Rightarrow f(011) = f(110)\\
100 \oplus 101 &= 001 \Rightarrow f(100) = f(001)\\
101 \oplus 101 &= 000 \Rightarrow f(101) = f(000)\\
110 \oplus 101 &= 011 \Rightarrow f(110) = f(011)\\
111 \oplus 101 &= 010 \Rightarrow f(111) = f(010)
\end{aligned}
$$

this must hold if $f(\mathbf{x})=f(\mathbf{y})$ over $\mathbf{c}$, the period of f

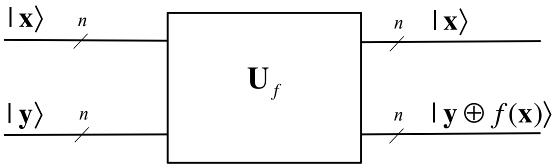
if only possible $\mathbf{c}=0^n$, what does that imply about f

f is not periodic

6

Function Represented as Oracle

- Function of Interest Specified as a Unitary Operation (Oracle) of the Form:



- Setting $\mathbf{y}=0^n$ Provides a Convenient way to evaluate $f(\mathbf{x})$ When it is an Oracle
- Setting $\mathbf{y}=1^n$ Provides a Convenient way to evaluate inverse of $f(\mathbf{x})$ When it is an Oracle

7

Classical Solution to Problem

- Evaluate $f(\mathbf{x})$ on different binary strings
- After Each Evaluation, Check if Function Response has Already Been Found
- If two strings $\mathbf{x}_1$ and $\mathbf{x}_2$ are found such that $f(\mathbf{x}_1)=f(\mathbf{x}_2)$ then it is assured that:

$$\mathbf{x}_1 = \mathbf{x}_2 \oplus \mathbf{c}$$

- How do we find $\mathbf{c}$?

8

Classical Solution to Problem

- Evaluate $f(\mathbf{x})$ on different binary strings
- After Each Evaluation, Check if Function Response has Already Been Found
- If two strings $\mathbf{x}_1$ and $\mathbf{x}_2$ are found such that $f(\mathbf{x}_1)=f(\mathbf{x}_2)$ then: $\mathbf{x}_1 = \mathbf{x}_2 \oplus \mathbf{c}$
- This Must Hold Consistently for All $\mathbf{x}_i$ Pairs
- If it Does, How do we find $\mathbf{c}$?

$$\mathbf{x}_2 \oplus \mathbf{x}_1 = \mathbf{x}_2 \oplus \mathbf{x}_2 \oplus \mathbf{c} = \left(\mathbf{x}_2 \oplus \mathbf{x}_2\right) \oplus \mathbf{c} = 0 \oplus \mathbf{c} = \mathbf{c}$$

$$\mathbf{x}_1 \oplus \mathbf{x}_2 = \mathbf{c}$$

9

Classical Solution Complexity

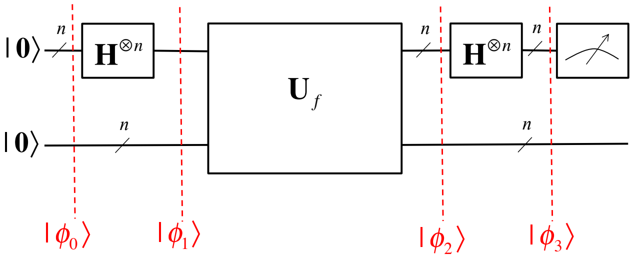
- If $f(\mathbf{x})$ is periodic (it is two-to-one) then a repeat will be found before half of inputs are evaluated
- If more than half of inputs checked with no match, then $f(\mathbf{x})$ is not periodic and $\mathbf{c}=0^n$
- Worst case number of evaluations:

$$\frac{2^n}{2} + 1 = 2^{n-1} + 1$$

- Exponential Complexity

10

Periodicity Quantum Algorithm



$$|\phi_0\rangle = |\mathbf{00}\rangle \qquad |\phi_2\rangle = \mathbf{U}_f\,|\phi_1\rangle$$

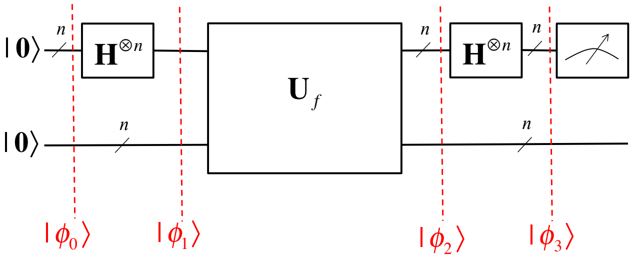
$$|\phi_1\rangle = \left(\mathbf{H}^{\otimes n} \otimes \mathbf{I}_n\right)|\phi_0\rangle \qquad |\phi_3\rangle = \left(\mathbf{H}^{\otimes n} \otimes \mathbf{I}_n\right)|\phi_2\rangle$$

$$|\phi_3\rangle = \mathbf{G}_{simon}\,|\phi_0\rangle$$

$$\mathbf{G}_{simon} = \left(\mathbf{H}^{\otimes n} \otimes \mathbf{I}_n\right)\mathbf{U}_f\left(\mathbf{H}^{\otimes n} \otimes \mathbf{I}_n\right)$$

11

Periodicity Quantum Algorithm

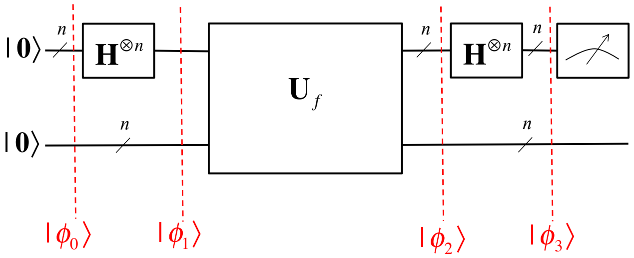


- Places Upper n Qubits in State of Superposition
- EXAMPLE: n=3:

$$|\phi_1\rangle = \left(\mathbf{H}^{\otimes 3} \otimes \mathbf{I}_3\right)|\phi_0\rangle = \frac{\sum_{\mathbf{x}\in\{0,1\}^3} |\mathbf{x},\mathbf{0}\rangle}{\sqrt{2^3}} = \frac{|000,000\rangle + |001,000\rangle + \ldots + |111,000\rangle}{\sqrt{8}}$$

12

Periodicity Quantum Algorithm



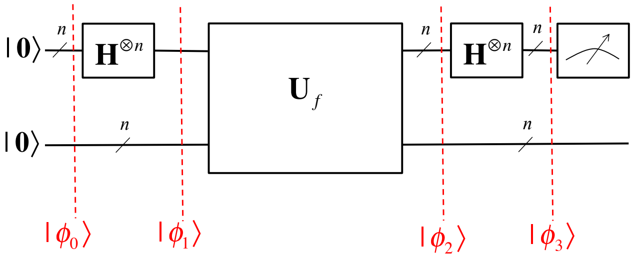
- $\mathbf{U}_f$ causes evaluation of f for the superimposed quantum state

$$|\phi_2\rangle = \mathbf{U}_f\left(\mathbf{H}^{\otimes n} \otimes \mathbf{I}_n\right)|\phi_1\rangle = \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |\mathbf{x}, f(\mathbf{x})\rangle$$

Hadamard is Applied to $|\mathbf{0}\rangle$ so we Express it this way

13

Periodicity Quantum Algorithm



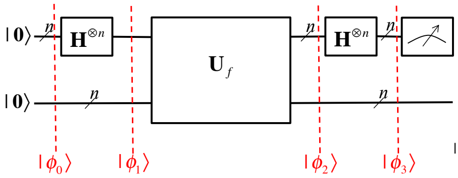
- n-Dimensional Hadamard Transform applied again

$$|\phi_3\rangle = \left(\mathbf{H}^{\otimes n} \otimes \mathbf{I}_n\right) |\phi_2\rangle = \frac{\sum_{\mathbf{x}\in\{0,1\}^n} \sum_{\mathbf{z}\in\{0,1\}^n} (-1)^{\langle \mathbf{z},\mathbf{x}\rangle} |\mathbf{z}, f(\mathbf{x})\rangle}{\sqrt{2^n}}$$

Hadamard is Expressed Generally as Explained in Class Notes on "Hadamard Transform"

14

Periodicity Quantum Algorithm



$$|\phi_3\rangle = \frac{\sum_{\mathbf{x}\in\{0,1\}^n}\sum_{\mathbf{z}\in\{0,1\}^n}(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}|\mathbf{z}, f(\mathbf{x})\rangle}{\sqrt{2^n}}$$

- n-Dimensional Hadamard Transform applied again
- For each $\mathbf{x}$ and $\mathbf{z}$ in the Summations, Consider when: $|\mathbf{z}, f(\mathbf{x})\rangle = |\mathbf{z}, f(\mathbf{x}\oplus\mathbf{c})\rangle$
- When this occurs, same coefficients appear twice in Double Summations:

first instance for $f(\mathbf{x})$

second instance for $f(\mathbf{x}\otimes\mathbf{c})$

same result from 2nd second set of Hadamard Gates after Oracle since $f(\mathbf{x}) = f(\mathbf{x}\otimes\mathbf{c})$

$$\frac{(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}+(-1)^{\langle \mathbf{z},\mathbf{x}\oplus\mathbf{c}\rangle}}{2} = \frac{(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}+(-1)^{\langle \mathbf{z},\mathbf{x}\rangle\oplus\langle \mathbf{z},\mathbf{c}\rangle}}{2}$$

$$= \frac{(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}+(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}(-1)^{\langle \mathbf{z},\mathbf{c}\rangle}}{2}$$

Make use of "Bit String Inner Product Properties" Slide in Class Notes for "Hadamard Transform" re: $\langle , \rangle$ Notation (*refer to this slide set for reminder*)

15

Periodicity Quantum Algorithm

$$\frac{(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}+(-1)^{\langle \mathbf{z},\mathbf{x}\oplus\mathbf{c}\rangle}}{2} = \frac{(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}+(-1)^{\langle \mathbf{z},\mathbf{x}\rangle\oplus\langle \mathbf{z},\mathbf{c}\rangle}}{2} = \frac{(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}+(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}(-1)^{\langle \mathbf{z},\mathbf{c}\rangle}}{2}$$

- When: $\langle \mathbf{z},\mathbf{c}\rangle = 1$ (Destructive Interference – Amplitudes go to 0)

$$\frac{(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}+(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}(-1)^{\langle \mathbf{z},\mathbf{c}\rangle}}{2} = \frac{(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}+(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}(-1)^{1}}{2} = \frac{(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}-(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}}{2} = 0$$

- When: $\langle \mathbf{z},\mathbf{c}\rangle = 0$ (Constructive Interference – Amplitude Magnitudes are Non-zero)

$$\frac{(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}+(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}(-1)^{\langle \mathbf{z},\mathbf{c}\rangle}}{2} = \frac{(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}+(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}(-1)^{0}}{2} = \frac{(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}+(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}}{2} = \pm 1$$

- Therefore, when measuring the top qubits, we only find those binary strings where: $\langle \mathbf{z},\mathbf{c}\rangle = 0$
- This Happens Because Zero-valued Hadamard Coefficients Cancel Out Cases for $\langle \mathbf{z},\mathbf{c}\rangle$ (0-*valued multiplier*)
- We Want These Cases in Our Measurement Since It Allows Us to Solve for the Period (*as you will see later*)

16

Example Function

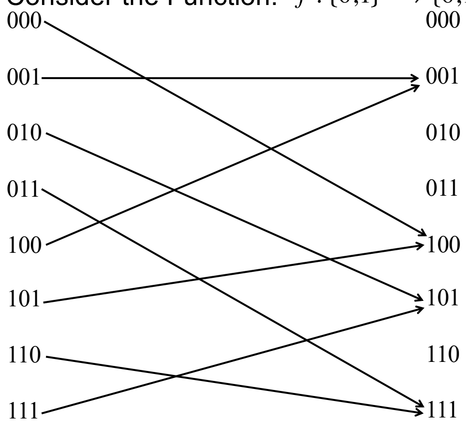
- Consider the Function: $f:\{0,1\}^3 \rightarrow \{0,1\}^3$
- Truth Table Representation – Embedded Inside $\mathbf{U}_f$

$x_1\ x_2\ x_3$	f
0 0 0	100
0 0 1	001
0 1 0	101
0 1 1	111
1 0 0	001
1 0 1	100
1 1 0	111
1 1 1	101

17

Periodicity Algorithm Example

- Consider the Function: $f:\{0,1\}^3 \rightarrow \{0,1\}^3$



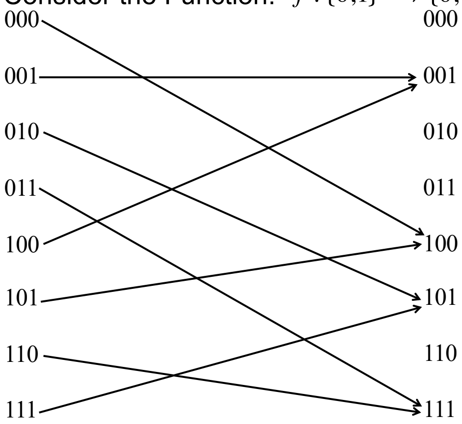
Is this Function Periodic?

If "Yes," what is the Period?

18

Periodicity Algorithm Example

- Consider the Function: $f:\{0,1\}^3 \rightarrow \{0,1\}^3$



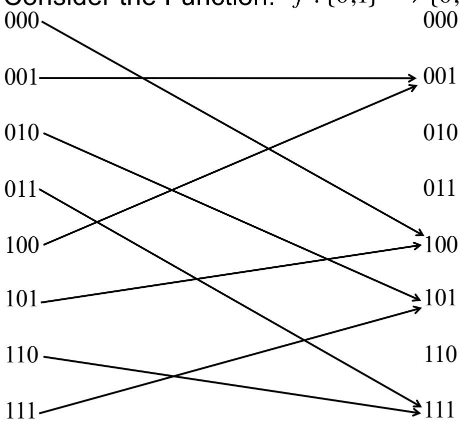
Is this Function Periodic?
YES

If "Yes," what is the Period?

19

Periodicity Algorithm Example

- Consider the Function: $f:\{0,1\}^3 \rightarrow \{0,1\}^3$

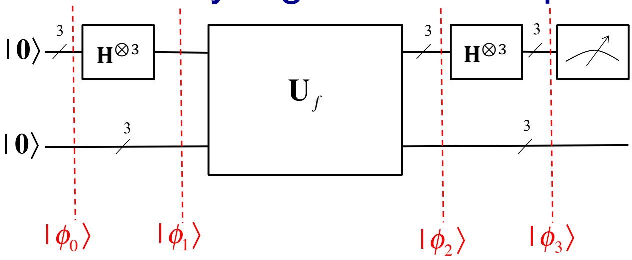


Is this Function Periodic?
YES

If "Yes," what is the Period?
The Period is: 101

20

Periodicity Algorithm Example



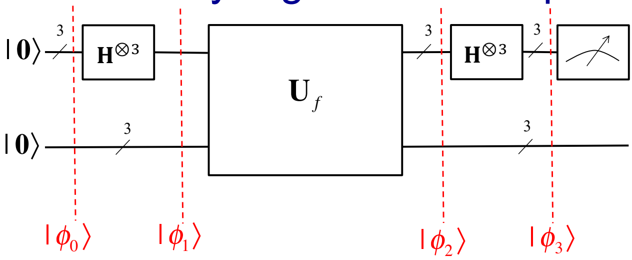
$$|\phi_0\rangle = |\mathbf{00}\rangle = |\mathbf{0}\rangle \otimes |\mathbf{0}\rangle = |000\rangle \otimes |000\rangle$$

$$|\phi_1\rangle = \left(\mathbf{H}^{\otimes 3} \otimes \mathbf{I}_3\right)|\phi_0\rangle = \frac{\sum_{\mathbf{x}\in\{0,1\}^3} |\mathbf{x},\mathbf{0}\rangle}{\sqrt{2^3}} = \frac{|000,000\rangle + |001,000\rangle + \ldots + |111,000\rangle}{\sqrt{8}}$$

$$= \frac{1}{\sqrt{8}}(|000\rangle\otimes|000\rangle + |001\rangle\otimes|000\rangle + |010\rangle\otimes|000\rangle + |011\rangle\otimes|000\rangle$$
$$+ |100\rangle\otimes|000\rangle + |101\rangle\otimes|000\rangle + |110\rangle\otimes|000\rangle + |111\rangle\otimes|000\rangle)$$

21

Periodicity Algorithm Example



$$|\phi_2\rangle = \mathbf{U}_f\left(\mathbf{H}^{\otimes 3} \otimes \mathbf{I}_3\right)|\phi_0\rangle = \frac{\sum_{\mathbf{x}\in\{0,1\}^3} |\mathbf{x}, f(\mathbf{x})\rangle}{\sqrt{2^3}} = \frac{\sum_{\mathbf{x}\in\{0,1\}^3} |\mathbf{x}\rangle \otimes |f(\mathbf{x})\rangle}{\sqrt{8}}$$

22

Periodicity Algorithm Example

$$|\phi_2\rangle = \mathbf{U}_f\left(\mathbf{H}^{\otimes 3}\otimes \mathbf{I}_3\right)|\phi_0\rangle = \frac{\sum_{\mathbf{x}\in\{0,1\}^3} |\mathbf{x}, f(\mathbf{x})\rangle}{\sqrt{2^3}} = \frac{\sum_{\mathbf{x}\in\{0,1\}^3} |\mathbf{x}\rangle\otimes|f(\mathbf{x})\rangle}{\sqrt{8}}$$

$$|\phi_2\rangle = \frac{1}{\sqrt{8}}(|000\rangle\otimes|100\rangle + |001\rangle\otimes|001\rangle + |010\rangle\otimes|101\rangle + |011\rangle\otimes|111\rangle$$
$$+|100\rangle\otimes|001\rangle + |101\rangle\otimes|100\rangle + |110\rangle\otimes|111\rangle + |111\rangle\otimes|101\rangle)$$

- In the Next Stage of the Cascade:

$$|\phi_3\rangle = \frac{\sum_{\mathbf{x}\in\{0,1\}^3}\sum_{\mathbf{z}\in\{0,1\}^3}(-1)^{\langle \mathbf{z},\mathbf{x}\rangle}|\mathbf{z}\rangle\otimes|f(\mathbf{x})\rangle}{\left(\sqrt{8}\right)\left(\sqrt{8}\right)}$$

23

Periodicity Algorithm Example

- Writing this out Term by Term Yields:

$$|\phi_3\rangle = \frac{1}{8}((+1)|000\rangle\otimes|f(000)\rangle + (+1)|000\rangle\otimes|f(001)\rangle + (+1)|000\rangle\otimes|f(010)\rangle + (+1)|000\rangle\otimes|f(011)\rangle$$
$$+(+1)|000\rangle\otimes|f(100)\rangle + (+1)|000\rangle\otimes|f(101)\rangle + (+1)|000\rangle\otimes|f(110)\rangle + (+1)|000\rangle\otimes|f(111)\rangle$$

$$+((+1)|001\rangle\otimes|f(000)\rangle + (-1)|001\rangle\otimes|f(001)\rangle + (+1)|001\rangle\otimes|f(010)\rangle + (-1)|001\rangle\otimes|f(011)\rangle$$
$$+(+1)|001\rangle\otimes|f(100)\rangle + (-1)|001\rangle\otimes|f(101)\rangle + (+1)|001\rangle\otimes|f(110)\rangle + (-1)|001\rangle\otimes|f(111)\rangle$$

$$+((+1)|010\rangle\otimes|f(000)\rangle + (+1)|010\rangle\otimes|f(001)\rangle + (-1)|010\rangle\otimes|f(010)\rangle + (-1)|010\rangle\otimes|f(011)\rangle$$
$$+(+1)|010\rangle\otimes|f(100)\rangle + (+1)|010\rangle\otimes|f(101)\rangle + (-1)|010\rangle\otimes|f(110)\rangle + (-1)|010\rangle\otimes|f(111)\rangle$$

$$+((+1)|011\rangle\otimes|f(000)\rangle + (-1)|011\rangle\otimes|f(001)\rangle + (-1)|011\rangle\otimes|f(010)\rangle + (+1)|011\rangle\otimes|f(011)\rangle$$
$$+(+1)|011\rangle\otimes|f(100)\rangle + (-1)|011\rangle\otimes|f(101)\rangle + (-1)|011\rangle\otimes|f(110)\rangle + (+1)|011\rangle\otimes|f(111)\rangle$$

24

Periodicity Algorithm Example

$$+(+1)|100\rangle\otimes|f(000)\rangle+(+1)|100\rangle\otimes|f(001)\rangle+(+1)|100\rangle\otimes|f(010)\rangle+(+1)|100\rangle\otimes|f(011)\rangle$$
$$+(-1)|100\rangle\otimes|f(100)\rangle+(-1)|100\rangle\otimes|f(101)\rangle+(-1)|100\rangle\otimes|f(110)\rangle+(-1)|100\rangle\otimes|f(111)\rangle$$

$$+((+1)|101\rangle\otimes|f(000)\rangle+(-1)|101\rangle\otimes|f(001)\rangle+(+1)|101\rangle\otimes|f(010)\rangle+(-1)|101\rangle\otimes|f(011)\rangle$$
$$+(-1)|101\rangle\otimes|f(100)\rangle+(+1)|101\rangle\otimes|f(101)\rangle+(-1)|101\rangle\otimes|f(110)\rangle+(+1)|101\rangle\otimes|f(111)\rangle$$

$$+((+1)|110\rangle\otimes|f(000)\rangle+(+1)|110\rangle\otimes|f(001)\rangle+(-1)|110\rangle\otimes|f(010)\rangle+(-1)|110\rangle\otimes|f(011)\rangle$$
$$+(-1)|110\rangle\otimes|f(100)\rangle+(-1)|110\rangle\otimes|f(101)\rangle+(+1)|110\rangle\otimes|f(110)\rangle+(+1)|110\rangle\otimes|f(111)\rangle$$

$$+((+1)|111\rangle\otimes|f(000)\rangle+(-1)|111\rangle\otimes|f(001)\rangle+(-1)|111\rangle\otimes|f(010)\rangle+(+1)|111\rangle\otimes|f(011)\rangle$$
$$+(-1)|111\rangle\otimes|f(100)\rangle+(+1)|111\rangle\otimes|f(101)\rangle+(+1)|111\rangle\otimes|f(110)\rangle+(-1)|111\rangle\otimes|f(111)\rangle)$$

- Coefficients in Red are the Elements of: $\mathbf{H}^{\otimes 3}$

- Evaluating the Function f in the Previous Equation Yields the Following:

25

Periodicity Algorithm Example

- Evaluating the Function f in the Previous Equation Yields the Following:

$$|\phi_3\rangle=\frac{1}{8}((+1)|000\rangle\otimes|100\rangle+(+1)|000\rangle\otimes|001\rangle+(+1)|000\rangle\otimes|101\rangle+(+1)|000\rangle\otimes|111\rangle$$
$$+(+1)|000\rangle\otimes|001\rangle+(+1)|000\rangle\otimes|100\rangle+(+1)|000\rangle\otimes|111\rangle+(+1)|000\rangle\otimes|101\rangle$$

$$+((+1)|001\rangle\otimes|100\rangle+(-1)|001\rangle\otimes|001\rangle+(+1)|001\rangle\otimes|101\rangle+(-1)|001\rangle\otimes|111\rangle$$
$$+(+1)|001\rangle\otimes|001\rangle+(-1)|001\rangle\otimes|100\rangle+(+1)|001\rangle\otimes|111\rangle+(-1)|001\rangle\otimes|101\rangle$$

$$+((+1)|010\rangle\otimes|100\rangle+(+1)|010\rangle\otimes|001\rangle+(-1)|010\rangle\otimes|101\rangle+(-1)|010\rangle\otimes|111\rangle$$
$$+(+1)|010\rangle\otimes|001\rangle+(+1)|010\rangle\otimes|100\rangle+(-1)|010\rangle\otimes|111\rangle+(-1)|010\rangle\otimes|101\rangle$$

$$+((+1)|011\rangle\otimes|100\rangle+(-1)|011\rangle\otimes|001\rangle+(-1)|011\rangle\otimes|101\rangle+(+1)|011\rangle\otimes|111\rangle$$
$$+(+1)|011\rangle\otimes|001\rangle+(-1)|011\rangle\otimes|100\rangle+(-1)|011\rangle\otimes|111\rangle+(+1)|011\rangle\otimes|101\rangle$$

26

Periodicity Algorithm Example

- Evaluating the Function f in the Previous Equation Yields the Following (continued):

$$+(+1)|100\rangle\otimes|100\rangle+(+1)|100\rangle\otimes|001\rangle+(+1)|100\rangle\otimes|101\rangle+(+1)|100\rangle\otimes|111\rangle$$
$$+(-1)|100\rangle\otimes|001\rangle+(-1)|100\rangle\otimes|100\rangle+(-1)|100\rangle\otimes|111\rangle+(-1)|100\rangle\otimes|101\rangle$$

$$+((+1)|101\rangle\otimes|100\rangle+(-1)|101\rangle\otimes|001\rangle+(+1)|101\rangle\otimes|101\rangle+(-1)|101\rangle\otimes|111\rangle$$
$$+(-1)|101\rangle\otimes|001\rangle+(+1)|101\rangle\otimes|100\rangle+(-1)|101\rangle\otimes|111\rangle+(+1)|101\rangle\otimes|101\rangle$$

$$+((+1)|110\rangle\otimes|100\rangle+(+1)|110\rangle\otimes|001\rangle+(-1)|110\rangle\otimes|101\rangle+(-1)|110\rangle\otimes|111\rangle$$
$$+(-1)|110\rangle\otimes|001\rangle+(-1)|110\rangle\otimes|100\rangle+(+1)|110\rangle\otimes|111\rangle+(+1)|110\rangle\otimes|101\rangle$$

$$+((+1)|111\rangle\otimes|100\rangle+(-1)|111\rangle\otimes|001\rangle+(-1)|111\rangle\otimes|101\rangle+(+1)|111\rangle\otimes|111\rangle$$
$$+(-1)|111\rangle\otimes|001\rangle+(+1)|111\rangle\otimes|100\rangle+(+1)|111\rangle\otimes|111\rangle+(-1)|111\rangle\otimes|101\rangle)$$

27

Periodicity Algorithm Example

- Combining Like Terms and Cancelling Out Where Possible Yields the Following:

$$|\phi_3\rangle=\frac{1}{8}((+2)|000\rangle\otimes|100\rangle+(+2)|000\rangle\otimes|001\rangle+(+2)|000\rangle\otimes|101\rangle+(+2)|000\rangle\otimes|111\rangle$$
$$+(+2)|010\rangle\otimes|100\rangle+(+2)|010\rangle\otimes|001\rangle+(-2)|010\rangle\otimes|101\rangle+(-2)|010\rangle\otimes|111\rangle$$
$$+(+2)|101\rangle\otimes|100\rangle+(-2)|101\rangle\otimes|001\rangle+(+2)|101\rangle\otimes|101\rangle+(-2)|101\rangle\otimes|111\rangle$$
$$+(+2)|111\rangle\otimes|100\rangle+(-2)|111\rangle\otimes|001\rangle+(-2)|111\rangle\otimes|101\rangle+(+2)|111\rangle\otimes|111\rangle)$$

$$|\phi_3\rangle=\frac{1}{8}((+2)|000\rangle\otimes(|100\rangle+|001\rangle+|101\rangle+|111\rangle)$$
$$+(+2)|010\rangle\otimes(|100\rangle+|001\rangle-|101\rangle-|111\rangle)$$
$$+(+2)|101\rangle\otimes(|100\rangle-|001\rangle+|101\rangle-|111\rangle)$$
$$+(+2)|111\rangle\otimes(|100\rangle-|001\rangle-|101\rangle+|111\rangle))$$

28

Periodicity Algorithm Example

$$
\begin{aligned}
|\phi_3\rangle = \frac{1}{8}\big((+2)|000\rangle &\otimes (|100\rangle + |001\rangle + |101\rangle + |111\rangle) \\
+(+2)|010\rangle &\otimes (|100\rangle + |001\rangle - |101\rangle - |111\rangle) \\
+(+2)|101\rangle &\otimes (|100\rangle - |001\rangle + |101\rangle - |111\rangle) \\
+(+2)|111\rangle &\otimes (|100\rangle - |001\rangle - |101\rangle + |111\rangle)\big)
\end{aligned}
$$

- Measuring the Top 3 Qubits Gives (with equal probability): $|000\rangle, |010\rangle, |101\rangle, |111\rangle$
- For Each of These Measured Quantum States, it is True that the Inner Product with the Period Bitstring $\mathbf{c}$ is Zero
- We want These Cases Only Because it Allows Us to Easily Solve for the Period $\mathbf{c}$ (*usually classically*)

29

Periodicity Algorithm Example

- For Each of These Measured Quantum States, it is True that the Inner Product with the Period Bitstring $\mathbf{c}$ is Zero
- The Algorithm/Circuit is Measured a Sufficient Number of Times to Ensure All Possible Measurements are Obtained
- This Yields a Set of Simultaneous Equations:

$$
\begin{aligned}
&(i) && \langle 000, \mathbf{c}\rangle = 0 \\
&(ii) && \langle 010, \mathbf{c}\rangle = 0 \\
&(iii) && \langle 101, \mathbf{c}\rangle = 0 \\
&(iv) && \langle 111, \mathbf{c}\rangle = 0
\end{aligned}
$$

30

Periodicity Algorithm Example

$$\langle 000, c_1c_2c_3 \rangle = 0$$
$$\langle 010, c_1c_2c_3 \rangle = 0$$
$$\langle 101, c_1c_2c_3 \rangle = 0$$
$$\langle 111, c_1c_2c_3 \rangle = 0$$

$$(0 \wedge c_1) \oplus (0 \wedge c_2) \oplus (0 \wedge c_3) = 0$$
$$(0 \wedge c_1) \oplus (1 \wedge c_2) \oplus (0 \wedge c_3) = 0$$
$$(1 \wedge c_1) \oplus (0 \wedge c_2) \oplus (1 \wedge c_3) = 0$$
$$(1 \wedge c_1) \oplus (1 \wedge c_2) \oplus (1 \wedge c_3) = 0$$

31

Periodicity Algorithm Example

$$c_2 = 0$$
$$c_1 \oplus c_3 = 0$$
$$c_1 \oplus c_2 \oplus c_3 = 0$$

- This Means that $c_1=c_3=0$ or that $c_1=c_3=1$
- We Know that $\mathbf{c}$ is NOT EQUAL to 000 Since Function was Found not to be One-to-One
- Therefore $c_1=c_3=1$
- Period of Function is:

$$\mathbf{c} = c_1c_2c_3 = 101$$

32

Periodicity Algorithm Example

- Must Run Simon's Algorithm Several Times to Measure n Different $\mathbf{z}$ Bitstrings
  - We Need a Complete Set of Linear Equations (modulo-2) in order to solve for each of the n distinct c_i bits in the "Period bitstring" $\mathbf{c}$

- Generally Use a Classical Computer for Solving n Different Linear Equations (modulo-2)
- Simon's Algorithm Generally Viewed as a "Quantum Function" or "Quantum Subroutine"
- Could Use a Quantum Linear Solver Also but Must Consider Runtime and Cost

33

Simon's Alg. Intuition

- First Bank of Hadamard Gates Places All Function Domain Values Into Equal (Perfect) Superposition – Enables Oracle to Produce All Function Values in One Pass and in Superposition
- When Function is 2-to-1, Oracle Produces Two Identical and Distinct Superimposed Function Values for Each $\mathbf{x}$: $f(\mathbf{x})$ and $f(\mathbf{x} \oplus \mathbf{c})$
- Second Bank of Hadamard Gates Cause Constructive Interference for Two Identical and Distinct Function Values, so Probability Amplitudes Increase in Magnitude
  - and Destructive Interference for Function Pairs that Don't Match, so Probability Amplitudes Decrease in Magnitude (go to zero ideally)
- Several Runs Allows Different Function Values that Match to Result as Collapsed Values
  - We Keep Iterating to Obtain n Distinct$f(\mathbf{x}_i)$ and $f(\mathbf{x}_i \oplus \mathbf{c})$
  - Each Measurement Results in a Random Hyperplane Constraint Since it Collapses to a Single $f(\mathbf{x}_i)$ and $f(\mathbf{x}_i \oplus \mathbf{c})$
- These Collapsed Values Define Set of n Linear Equations that we can Solve for the Period
  - Since they are Orthogonal (zero-valued), we can solve a set of Linear Equations

34

Summary

- Quantum Computers are Good for:
  1) Searching Random Database (Grover's Algorithm)
  2) Finding Periodicity in Functions (Simon's & QFT)
  3) Factoring Semiprime Values (Shor's Algorithm)
  4) A *Few* Other Cases (finding QC algorithms is an active research area – but Hard to do!)
- Simon's Method is a Specific Example of Detecting Periodicity
  - Restricted to Functions of the Form:

$$f: \mathbb{B}^n \rightarrow \mathbb{B}^n$$
$$f: \{0,1\}^n \rightarrow \{0,1\}^n$$

- "Quantum Advantage" Known for Searching & Period-finding
- General Case for Period-finding is the "Quantum Fourier Transform" (QFT)
- Shor's Algorithm Depends on Efficient Period Finding of Special Function (the "Modulo-Powers Function")

35