Noisy Simon Period Finding

Alexander May, Lars Schlieper, Jonathan Schwinger
Ruhr-University Bochum
arXiv:1910.00802

Simon's Institute – Feb 2020

Simon's problem

Simon problem

Given: $f : \mathbb{F}_2^n \to \mathbb{F}_2^n$ with $f(x) = f(y) \Leftrightarrow y \in \{x, x + s\}$
Find: period $s \in \mathbb{F}_2^n \setminus \vec{0}$

- Want to implement on IBM Q16 (15 qubits).
- Which errors? Can we handle them classically?

Definition of function f

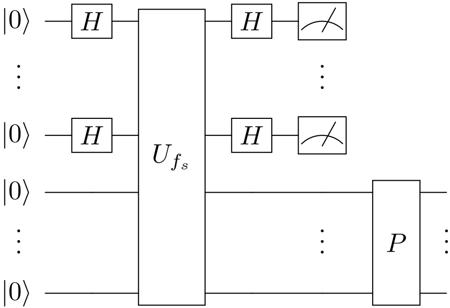
- Wlog $s_1 = 1$. Define $f_s : \mathbb{F}_2^n \to \mathbb{F}_2^n, \quad x \mapsto x + x_1 \cdot s$.

Lemma

(1) *f_s is Simon with period s, i.e. $f_s(x) = f_s(y)$ iff $y \in \{x, x+s\}$.*

(2) *Any Simon function is of the form $P \circ f_s$, for some bijection P.*

Warning: $f(1^n) = 1^n + s$.



IBMQ measurements

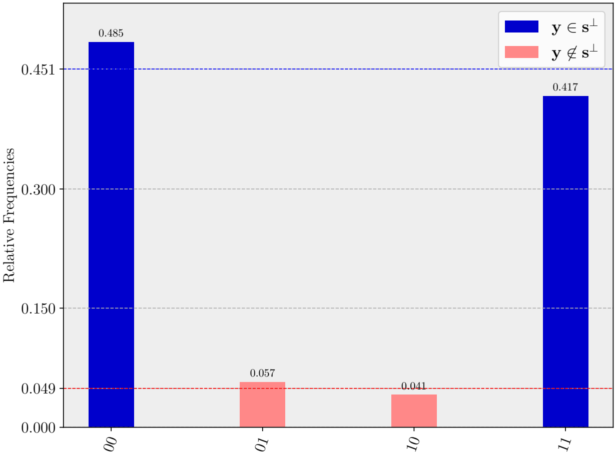


Figure: $\tau(2) = 0.099$

IBMQ measurements

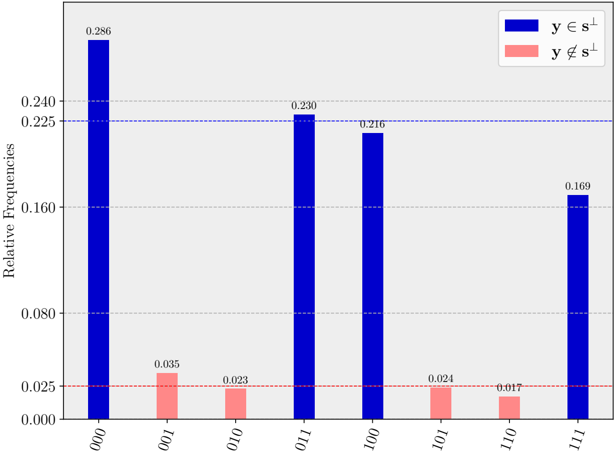


Figure: $\tau(3) = 0.098$

IBMQ measurements

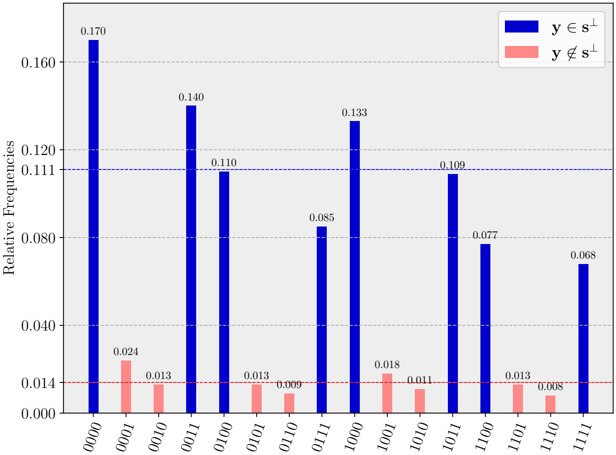


Figure: $\tau(4) = 0.102$

IBMQ measurements

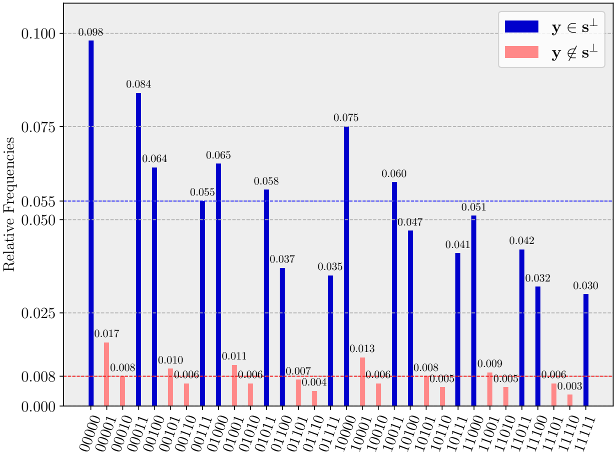


Figure: $\tau(5) = 0.107$

IBMQ measurements

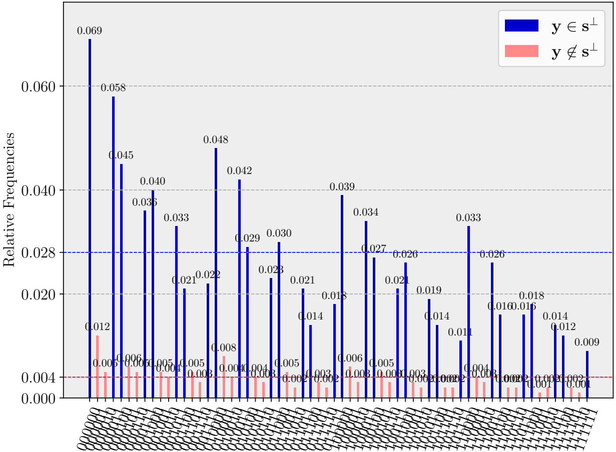


Figure: $\tau(6) = 0.112$

IBMQ measurements

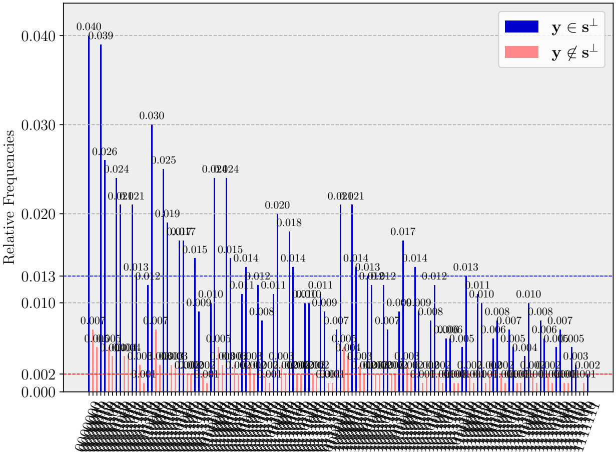


Figure: $\tau(7) = 0.117$

Smoothing

Experimental observations

- Good: Orthogonal vectors more frequent.
- Bad: Different qubit quality, bias towards zero.
- Inherent: $\tau(n)$ grows as a function of n.

Smoothing: Permutation of qubits + classical post-processing

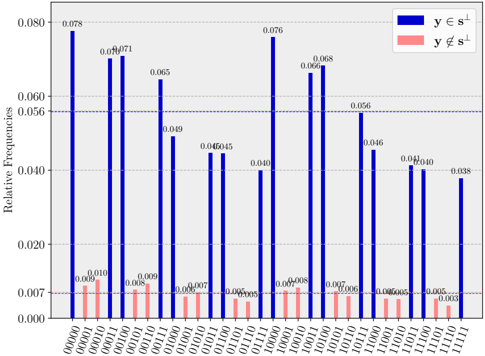


Figure: Raw IBMQ data, n=5.

Qubit Permutation

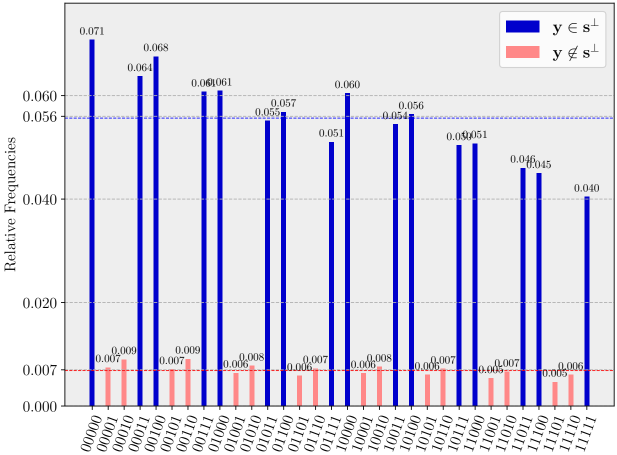


Figure: Permutation

Hamming Technique

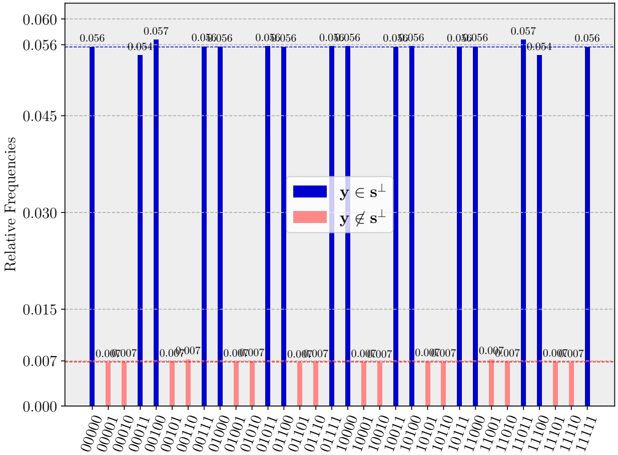


Figure: Smoothed IBMQ data.

Hardness of Error Correction

Learning Simon with Error (LSN)

We obtain $y \perp s$ with probability $1 - \tau$, uniformly distributed.
We obtain $y \not\perp s$ with probability τ, uniformly distributed.
Problem: Compute $s \in \mathbb{F}_2^n$?

Learning Parity with Noise (LPN)

Oracle: $(a, \langle a, s \rangle + \epsilon)$, where $a \in \mathbb{F}_2^n$ and $\Pr[\epsilon = 1] = \tau$.
Problem: Compute $s \in \mathbb{F}_2^n$?

Theorem

LSN(n, τ) and LPN(n, τ) are tightly polynomial equivalent.

Hard, but still Quantum Speedup

Theorem

For any $\tau < \frac{1}{2}$, we solve LPN(n, τ) in time $\mathcal{O}(2^{c(\tau)n})$ for some $c(\tau) < \frac{1}{2}$.

Numerical example

- Suppose we had quantum device with 468 qubits and error $\tau = \frac{1}{8}$.
  Compression technique: 235 qubit, $\tau = \frac{1}{8}$.
- We could run period finding on an $n = 234$ bit function f.
- Translates into an LPN-instance $(n, \tau) = (234, \frac{1}{8})$.
- Esser, Kübler, M. (2017): $(234, \frac{1}{8})$-LPN in **15 days** (64 threads).
- Purely classical, we would need $\approx 2^{117}$ steps.

Comparison

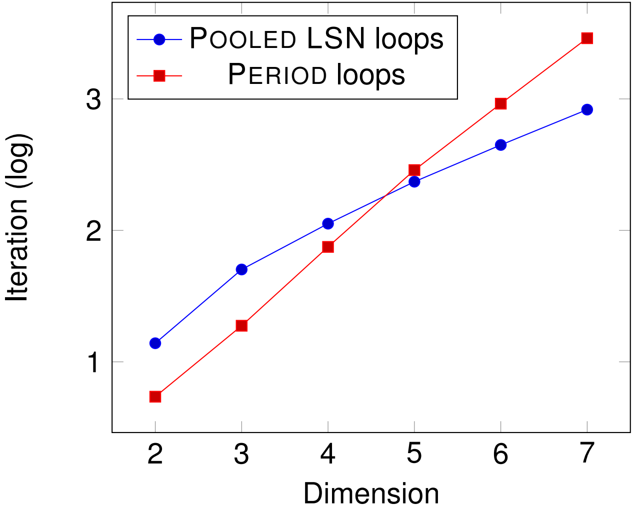


Figure: Loop comparison

Summary

- IBMQ error for Simon can be modeled as LPN samples.
- Correcting errors is hard, but not as hard as period finding.
- Still obtain polynomial speedups, rather than exponential.
- Even mid-scale noisy quantum devices might be useful.
- For Simon we do not necessarily need full error correction.
- Where is the break-even point in practice?