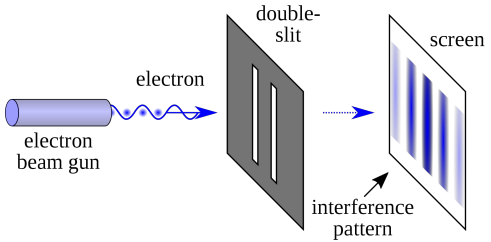
Grover’s Algorithm via Simulated Quantum Circuits

Stuart Larsen
Cloud Security

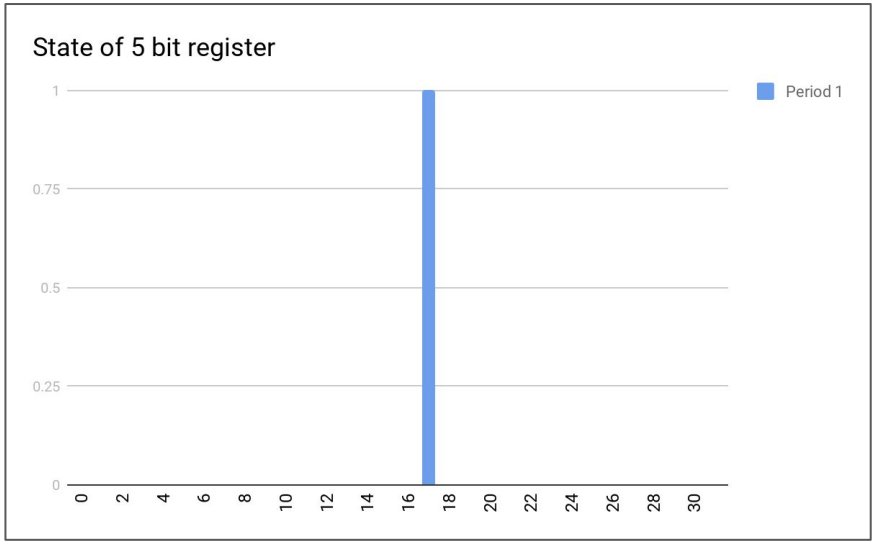
Agenda

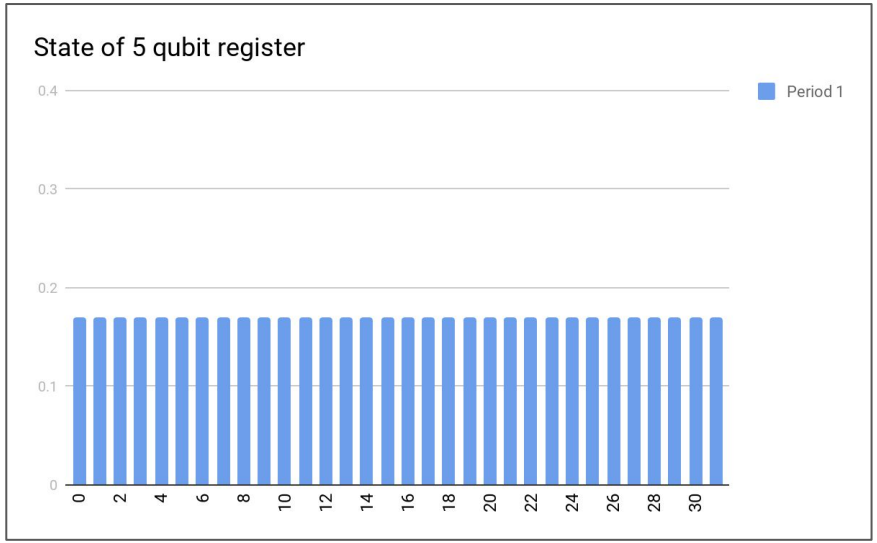
- Minute 1: What’s quantum computing?
- Minute 2: What’s Grover’s algorithm?
- Minute 3: Implementation

Quantum Computing



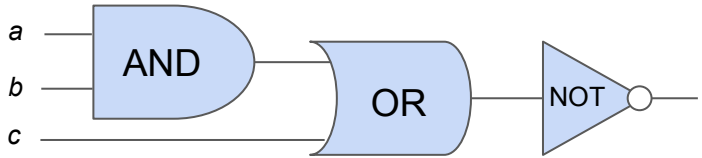
- System that takes advantage of certain quantum “weirdness”, specifically entangled states (superpositions)
- Certain algorithms can have exponential reductions in complexity (factoring (shor’s), particle simulation, AI, chemistry simulations, search)





Quantum Computing

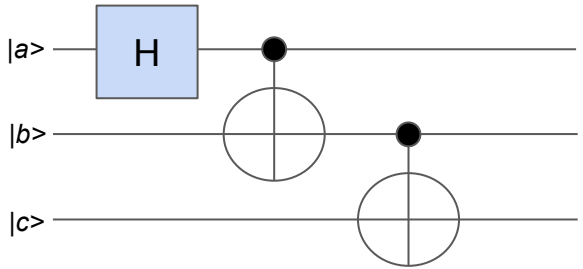
Classical Circuit




```
func f(a,b,c):
        return !((a&b) | c)
```

f(0, 0, 0) => 1
f(1, 0, 1) => 0

Quantum Circuit



I ⊕ CNOT CNOT ⊕ I H ⊕ I ⊕ I

f(0, 0, 0) = .7|*000*> + .7|*111*>

Grover’s Algorithm Problem

- Unindexed Search / Black box oracles
- Black box function f(x)
  - f(000) = 0 no
  - f(001) = 0 no
  - f(010) = 0 no
  - f(011) = 0 no
  - f(100) = 0 no
  - f(101) = 1 YES!
- SAT solvers (is there a set of input that returns true?)
- Unindexed database search (is x the index of a record with {name: ‘stuart’})
- Password hash cracking (what value gives the hash 230984792w3f8u2398) 😈

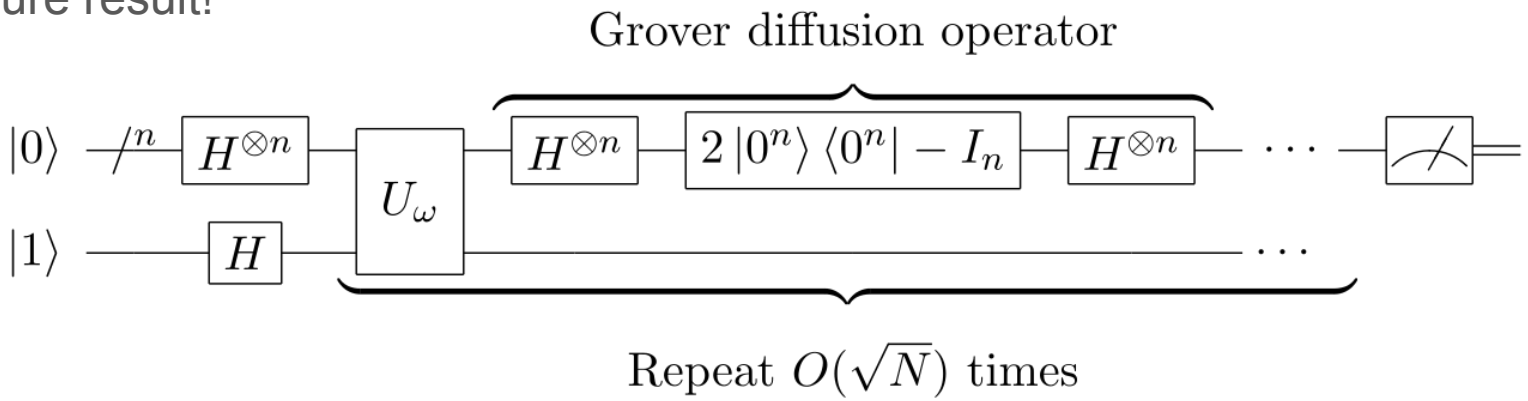
Classically, you need to call f with every possible guess (N) or 2^n.

WIth Grover’s you only need sqrt(N), or $2^{n/2}$

For cracking SHA256: 2^{256} = 115792089237316195423570985008687907853269984665640564039457584007913129639936L
2^{128} = 340282366920938463463374607431768211456L

Grover's Algorithm "test all inputs at the same time"

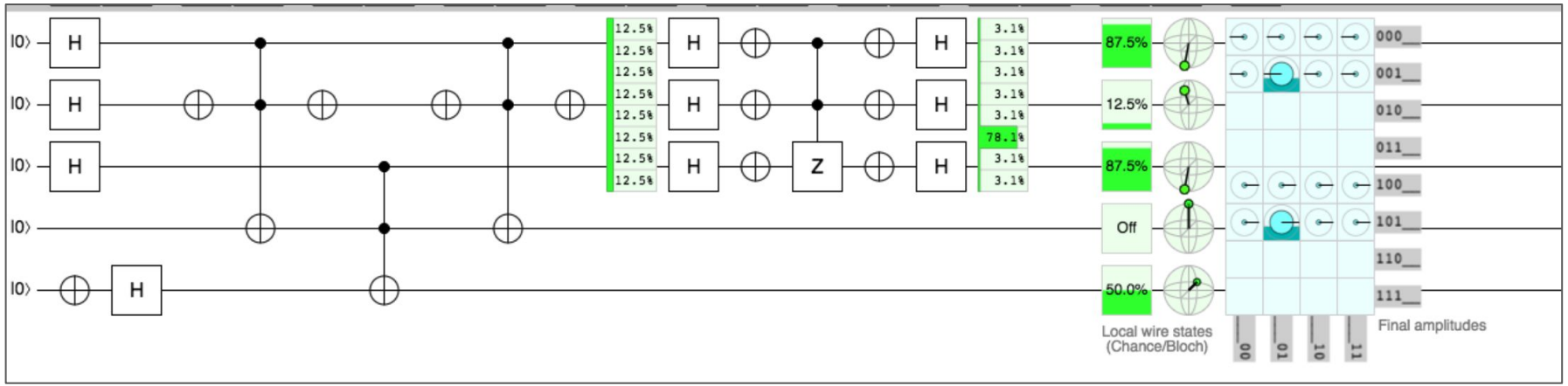
- Put all qubits into an "equal state" (Hadamard gate)
- Run the oracle function (and reverse)
  - Make sure to clean ancillary bits (all operations reversible)
- Phase inversion of the "correct answer"
- Diffusion Operation
  - Reflection around the mean, this is the mathy part (and the core of grovers)
- Measure result!



Implementation

- Demo f(5) = 1, f(x) = 0
  - Single Iteration!! (I run (fx) once!)
- complex/cvector/cmatrix/gates/circuit/algo's
- Grover's Algorithm n=3
- Quantum Superdense Coding / Bell's Inequality / GHZ Inequality

```
➜ qrad git:(master) ✗ go test
Grover Results f(x) = 1, x =...
010 0.02
011 0.03
101 0.79
000 0.04
001 0.04
110 0.03
111 0.03
100 0.04
PASS
ok      github.com/c0nrad/qrad  0.888s
```



https://github.com/c0nrad/qrad

Where to learn more?

slack:@stuart
email: stuart@mongodb.com

- Play with a real quantum computer!
  - IBM Q Experience
- Books:
  - Quantum Computation for Computer Scientist
  - Mastering Quantum Computing with IBM QX
  - No-nonsense quantum mechanics
  - Practical Quantum Computing for Developers
- Youtube series
  - “Quantum Computing for the determined”