THE EFFECTS OF NOISE ON QUANTUM ALGORITHMS SOLVING 3-SAT

MARTIJN SWENNE (SUPERVIZED BY VEDRAN DUNJKO)

LEIDEN INSTITUTE OF ADVANCED COMPUTER SCIENCE

s1923889@liacs.leidenuniv.nl

Universiteit Leiden
The Netherlands

1. OVERVIEW

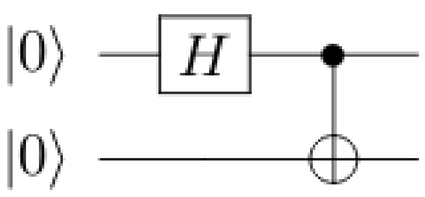
QUANTUM is a term you hear a lot lately. Quantum computing uses quantum-mechanical phenomena, such as superposition and entanglement. A quantum computer is therefor completely different than a classical computer. Whereas a classical computer uses binary digits, which are always in one of two definite states (0 or 1), a quantum computer uses quantum bits, or QUBITS. These qubits can be in a quantum superposition of states. Quantum computers are extremely powerful, but also very noisy. This is currently a big problem in quantum computers. Quantum computers can speed up certain combinatorial optimization problems, and here we investigate how noise affects the solving of 3SAT problems in particular.

2. BASICS

A QUANTUM COMPUTER operates on a register of n qubits using quantum gates and measurement. These n qubits be in any quantum superposition. Quantum states are represented by vectors of a 2^n dimensional vector space called the Hilbert space. Quantum gates are represented by unitary matrices acting on the state vectors, thereby changing it. Quantum computing elements are typically denoted using the DIRAC notation, for instance, the basis states of a single qubit are the ket $|0\rangle$ and the hermitan conjugate called the bra $\langle 0|$. Qubits and quantum gates can be described with matrices. For instance:
$|0\rangle = \left[\begin{smallmatrix}1\\0\end{smallmatrix}\right]$ and $|1\rangle = \left[\begin{smallmatrix}0\\1\end{smallmatrix}\right]$
Just as in classical computations, where you work with classical logic gates, quantum computations work with quantum logic gates, which are 1 or 2 qubit gates. Unlike classical logic gates, these gates are reversible. For instance, a basic quantum logic gate is the Hadamard gate:

$$H|0\rangle = \frac{1}{\sqrt{2}}\left[\begin{smallmatrix}1 & 1\\1 & -1\end{smallmatrix}\right]\left[\begin{smallmatrix}1\\0\end{smallmatrix}\right] = \frac{1}{\sqrt{2}}\left[\begin{smallmatrix}1\\1\end{smallmatrix}\right] = \frac{|0\rangle+|1\rangle}{\sqrt{2}}$$
$$H|1\rangle = \frac{1}{\sqrt{2}}\left[\begin{smallmatrix}1 & 1\\1 & -1\end{smallmatrix}\right]\left[\begin{smallmatrix}0\\1\end{smallmatrix}\right] = \frac{1}{\sqrt{2}}\left[\begin{smallmatrix}1\\-1\end{smallmatrix}\right] = \frac{|0\rangle-|1\rangle}{\sqrt{2}}$$

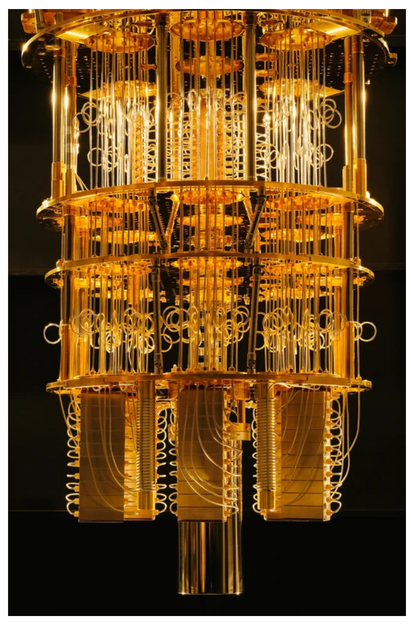
The Hadamard gate puts the qubit in a superposition of basis states. The probability of the resulting qubit being measured in the state is $|0\rangle$ is $(\frac{1}{\sqrt{2}})^2 = 0.5$ as given by the born rule.
There is another basic quantum gate, which is called the CNOT-gate. This is a controlled-NOT gate. The gate flips the state of the target qubit if and only if the control qubit is 1.



If you now apply a Hadamard gate on the control qubit before a CNOT-gate, the two qubits become ENTANGLED. This means that you cannot describe a qubit independently of the state of the other qubit.
Entanglement is a type of quantum correlation, which is at the basis of most of quantum information, and cannot be explained just by classical correlations.

3. QUANTUM COMPUTING

The image below is a superconducting qubit-based quantum computer from IBM. The biggest current QC's manipulate circa 50-60 qubits.



IBM Research Flickr (2018)

Quantum computers can be programmed by a number of programming languages. One of the most prevalent ones is QISKIT, which is a python library.

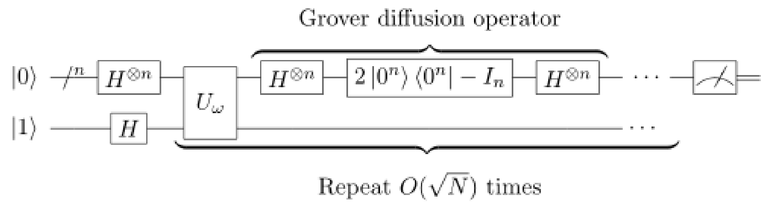
```
10      q = QuantumRegister(n)
11      qc = QuantumCircuit(q)
12
13      qc.cx(q[2],q[1])
14      qc.barrier()
15      cx_sq(qc,q,0,1, True)
16      qc.barrier()
17      qc.cx(q[2],q[1])
18      qc.barrier()
19      cx_sq(qc,q,0,1, False)
20      qc.barrier()
21      cx_sq(qc,q,0,2, False)
```

4. SOLVING 3-SAT

The BOOLEAN SATISFIABILITY problem determines if there is a satisfying assignment to a given Boolean formula. SAT is proven to be NP-complete. This means that it is solvable in polynomial time.
A special case of boolean satisfiability is the 3-SAT. This means the boolean formula is in CNF where each clause is limited to at most 3 literals. Such a boolean formula can be written like:

$$F(x_1 \dots xn) = (l_1 \vee l_2 \vee l_3) \wedge \dots \wedge (l_1 \vee l_2 \vee l_3)$$
$$\text{with } l_i = x_i \text{ or } l_i = \neg x_i$$

Quantum computers are not believed to be able to solve it in polynomial time. However, the GROVER'S SEARCH algorithm can solve it brute force in $O(2^{n/2})$, while a classical brute force solves it in $O(2^n)$.



Wikipedia; Grover's algorithm

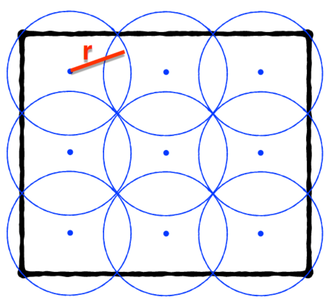
The SCHÖNING algorithm (walkSAT), is a classical algorithm that solves it in $O(\frac{4}{3}^n) = O(2^{0.415*n})$. We can apply Grover's search to Schöning's algorithm and this is faster than the fastest known classical algorithm. But such an algorithm needs many qubits. To make everything work with fewer qubits, we reduce SAT solving to Solving PromiseBall.

5. QUANTUM NOISE

QUANTUM DECOHERENCE, or quantum noise, happens when qubits lose information to the environment over time. This doesn't start until the qubits are used, like measuring them or performing a computation.
The effects of noise on real quantum computers can be studied via SIMULATIONS, by utilizing some of the standard noise models. Quantum noise currently gets simulated by adding random gates (such as the Pauli X gate) into the circuit.I am studying the effects of noise on the performance of quantum 3-SAT solver.

6. PROMISEBALL

The PROMISEBALLSAT (or PBS) is using the central subroutine of the derandomized Schöning algorithm. Given the randomized initial assignment x, let $B_r(x)$ denote the r-ball centered at x, i.e. the set of all bitstrings y has a HAMMING DISTANCE $\leq r$. To use PBS to solve the original kSAT problem, the space of possible assignments is covered by a number of r-balls.



Vedran Dunjko (2018)

$Promise\text{-}Ball(F, r, x)$ is a deterministic recursive divide-and-conquer algorithm. It first checks some conditions for (un)satisfiability, or if x is a satisfying assignment. Otherwise, in the recursive step, it finds the first unsatisfied clause C, and calls $PromiseBall(F^{[l=1]}, r1, x)$ for every literal $l \in C$. All clauses involving l are removed (or truncated).
This strategy is suitable for using quantum computers that are small in size. This is done by using grover search over all 3^r possibilities of all the choices in each recursive step. However, it is not clear what happens when these calls are made to a quantum device which is noisy.

7. FURTHER RESEARCH

[1] R. A. Moser, D. Scheder; A full derandomization of schöning's k-SAT algorithm (2011)

[2] E. Dantsin, A. Goerdt, E. A. Hirsch, R. Kannan, J. Kleinberg, C. Papadimitriou, P. Raghavan, and U. Schöning; Theoretical Computer Science 289, 69 (2002)

[3] T. Hertli; 3-SAT Faster and Simpler - Unique-SAT Bounds for PPSZ Hold in General (2014)

[4] V. Dunjko, Y. Ge, J. I. Cirac; Computational speedup using small quantum devices (2018)