Quantum Computing: Solving Non-deterministic Polynomial (NP) Problems in Polynomial Time

Anthony Redamonti *(Author)*
Dept. of Computer Science
Syracuse University College of Engineering & Computer Science
Syracuse, NY, USA
aeredamo@syr.edu

***Abstract*—Background: In algorithmic design and analysis, algorithms are generally categorized into two groups: Polynomial (P) and Nondeterministic polynomial (NP). Algorithms in the P class can be solved in polynomial time using S steps, where S is a function of its input size, N.**

$$\boldsymbol{S = F(N)}$$

Typical P class algorithms include multiplication, sorting, and other trivial calculations. Algorithms belonging to the NP class can be solved in non-deterministic polynomial time. In layman's terms, an NP problem can be solved in polynomial time if all possible answers to the problem can be checked for correctness at the same time. But how would that be possible? The answer: Quantum computing.

Super computers were previously used to solve NP problems, commonly taking days, weeks, or even years. The need for a polynomial solution to NP problems could not be greater. The applications are seemingly endless: balancing risk portfolios in investment banking, molecular simulation to improve battery design, finding a DNA sequence that best fits a collection of fragments (Human Genome Project), and protein folding (finding a cure for cancer).

Keywords: Quantum, Qubit, Entanglement, Superposition, Superconducting Circuits, Ion, Ion Trap, Optical Tweezers, Quantum Dot, Semiconductor Impurities, Quantum Noise, Grover's Algorithm, Hadamard Gate.

I. Introduction: How Quantum Computers Work

The first theoretical computer, the Turing Machine, was developed to help decode encrypted messages during WWII. Alan Turing's machine consisted of a long piece of tape that was broken into tiny squares. Each square had the capacity to hold a zero or one (binary). The user would write a program using the tape and send it into the machine. The machine would be able to translate the zeroes and ones into instructions and rapidly execute them. Even the supercomputers currently in use still abide by this same binary logic.

The quantum computer uses qubits to store information. Qubits are very small subatomic particles that represent electrons, protons, neutrons, or atoms, which allow scientists to use them to model the behavior of a computer system (CPU, Memory, etc.). They can be thought of as existing in a superposition of 0 and 1, meaning that they represent all possible values between the two states. While it is difficult to imagine, qubits allow quantum computers to execute all possible iterations of a program at the same time. The phenomena, also known as parallelism, allows quantum computers to execute millions of instructions at the same time, while a typical computer can only execute one (1 Instruction Per Cycle). A quantum computer with an architecture of 30 qubits can achieve speeds up to 10 teraflops, which is 1,000 times faster than the average computer [1].

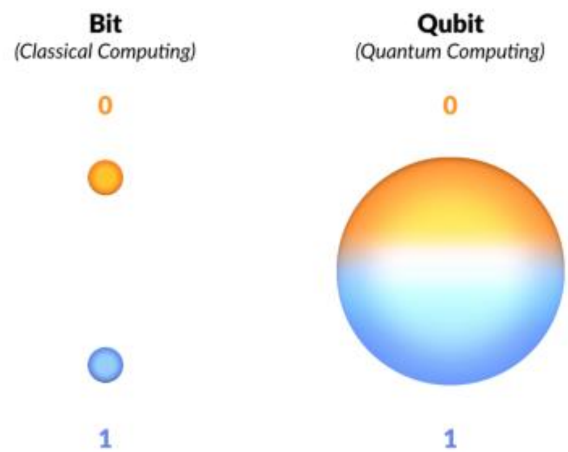

Fig. 1. Classical Bit vs Qubit [14]

There is another key concept in quantum computing: entanglement. The superposition of a qubit can be destroyed just by observing it. Due to quantum mechanics, an observer can interact with, or "bump", a subatomic particle by simply observing its state. A bumped qubit will be forced out of superposition and thus behave like a binary bit (0 or 1). How then can a scientist observe its state? The answer lies in the concept of entanglement. When an outside force interacts with a pair of atoms, the atoms become entangled. One atom will spin in a certain direction and the other atom will be forced by the first to spin in the opposite direction. Therefore, by observing the behavior of the first entangled atom, one can accurately predict the behavior of the second. Applying this external force to a large group of atoms allows computer scientists to observe their behavior indirectly, thus preserving their integrity.

I believe the article "How Quantum Computers Work" by Kevin Bonsor and Jonathan Strickland provides a sufficient overview in the more salient fundamentals of quantum computing. The building blocks of quantum computing are qubits, which exist in a quantum state, represent all possible values between zero and one. Parallelism and entanglement are key concepts to understanding the behavior of the qubits in the quantum realm. One fascinating concept is the subatomic "bump" that occurs when a qubit is directly observed. To affect something through observation is a strange phenomenon that is only observed in the quantum realm. It provides an accurate depiction of how sensitive quantum computers are to the outside world.

II. Qubit Control Method: Ion Traps

Ion traps are a method of manipulating ions serving as qubits in a quantum computer. Lasers are used to alter the state of a pair of entangled ions. But which ion pair should be chosen? MIT researchers at the Lincoln Laboratory have discovered a most promising pair: Calcium (Ca) and Strontium (Sr) ions. The reason they are favorable compared to other pairs is that the light from the lasers needed to manipulate them does not need to be ultraviolet. The light needed to manipulate this ion pair can be in the visible or infrared spectrum. The advantage is that there are presently more technologies in existence that use lasers with light in this spectrum [2].

The process is fascinating. To trap these ions, the researchers will house a chip in a steel vacuum chamber. The chip is cooled to nearly 450 degrees below zero Fahrenheit. Calcium and Strontium atoms are then flooded into the chamber, and the lasers shoot the atom's electrons, turning them into ions. The ions are trapped by electrodes which are attached to the chip's surface. Other lasers are used to cool the ions keeping them in the trap. The ions come together to form a Ca^+/Sr^+ crystal. The quantum computer is concerned with reading the energy level of the outermost electron in this pairing, which will be used for computational analysis. The electron could be at a ground state $|0\rangle$ or excited state $|1\rangle$ or both states simultaneously. The ability of the electron to be at both states at one time provide the quantum computer with its superpositional characteristic.

In my opinion, the article "A Trapped-ion Pair May Help Scale Up Quantum Computers" by Kylie Foy is an informative and comprehensive piece. It informs the reader of a method of qubit control: Ion Traps. I find the whole process of ion traps intriguing as many sciences are brought together to provide the most important characteristic of the quantum computer: superposition. As previously stated, superposition gives the quantum computer the ability to analyze millions of possible combinations of inputs to a given problem simultaneously (millions of instructions per cycle).

III. Manipulating Qubits Using Optical Tweezers

The article "Scalable Quantum Computing Stabilized by Optical Tweezers on an Ion Crystal" by Yu-Ching Shen and Guin-Dar Lin discusses some drawbacks of the previously discussed ion trap method. In an ion trap, electrodes construct a time-varying electric field used to trap the ions, serving as qubits that store encoded information. One drawback is scalability. To add more ions, one must construct a bigger ion trap. Each ion must be able to be addressed by the computer, thus each ion must have a minimum spacing of 10 micrometers. The spacing length is defined by the beam width used to excite each ion. As the array of ions grows, it becomes increasingly difficult to cool. The temperature must be at a stable -450°F and as more qubits interact due to entanglement, the temperature will rise [3].

The obstacles can be overcome using optical tweezers. Optical tweezers are dipole traps that can be focused to widths of a few microns. They are used to illuminate individual ions and have advanced reconfigurability. Optical tweezers can be activated and used to address an individual ion in the array in nanoseconds. Electrode control cannot achieve such a feat as varying voltages raise the temperature of the array of ions. Because of its speedy operating time, the integrity of the array and temperature are maintained when using optic tweezers to address individual ions.

I believe the article provides insight into some of the barriers that scientists are encountering when addressing individual ions in an ion trap. Temperature and array integrity (spacing between ions) must be maintained as array size grows. The quick operating time of optical tweezers not only allows the trap's architecture and temperature to remain intact but minimizes the amount of unwanted movement, called shuttling, within the trap. Optical tweezers are so fast and reconfigurable that they can pin a localized ion between the tweezered ion pair so that its motion profile can be studied. Due to entanglement, by studying the movement (spin rate and direction) of an ion, scientists can learn more about the localized region of ions in which the pinned ion is located without affecting the entangled ions within that region.

IV. Quantum Dots

Quantum dots are nanoscale pieces of semiconductor material that contain thousands of atoms. They absorb and emit light through electron interaction, much like an atom. Researchers are experimenting with quantum dots to serve as the qubit in quantum computer logic circuit design because they operate at much higher temperatures than existing qubits. The quantum dot would advance the goal of making the quantum computer less expensive and more practical.

The existing designs using normal qubits demand temperature of 100mK which is very difficult to maintain [4]. The researchers create quantum dots that are about 20 nanometers long. Because of their size, a quantum dot is governed by the laws of quantum mechanics. A single qubit can be made from a pair of quantum dots which have one or a few electrons associated with it. The electrons provide the state of the qubit (value between 0 and 1).

Andrew Dzurak and his group of researchers at the University of Wales, Australia, chose to create quantum dots from silicon-28, which is an isotope that does not naturally spin. They did not want the quantum dot to interfere with the spin of its electron, which would alter its state. They also found that these silicon qubits could operate at temperatures reaching up to 1 degree kelvin, which is -272.15 °C. Compared to previous qubit designs, which required temperatures of -450 °C, these qubits will provide a more cost-effective manufacturing and operating process for companies creating quantum computers.

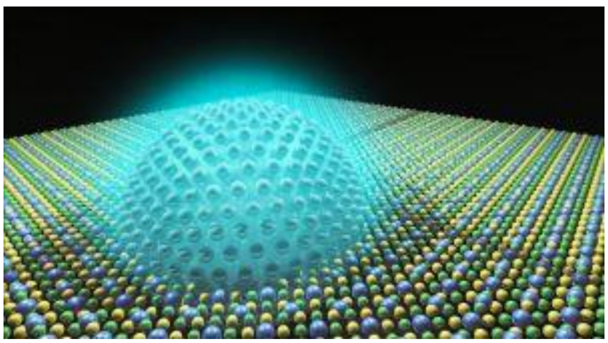

Fig. 2. Artist's Rendition of a Spherical Quantum Dot [15]

In my opinion, the article provides an inspiring development in the field of qubit design. By selecting Silicon-28 as their quantum dot material, Andrew Dzurak and his team have discovered a way to save potentially millions of dollars in quantum computer maintenance and design. While the operating temperature of the quantum dots are still quite cold, they are an order of magnitude greater than their predecessor. The article is only a few months old, which tells me that more developments and advancements are assuredly underway.

In October of 2020, Google released a new quantum computer that contained 53 superconducting qubits. While this computer was a technological breakthrough in that it could perform calculations that no supercomputer could, Google did announce that it was not possible to increase its size needed for practical use. I believe that the development from Dzurak's team will help technology giants like Google make their designs more feasible and cost effective. A physicist from the University of California, Hong Wen, stated that the work by Dzurak and his team "represents a technological breakthrough for semiconductor-based quantum computing [4]."

V. Superconducting Circuits

Quantum computers use superconducting circuits to decrease power consumption while increasing processing power. Superconducting circuits have zero electrical resistance, meaning that electrons can travel unimpeded. As a result, no energy is lost to voltage drop. To produce this effect, they operate at extremely low temperatures to reduce the vibration of their atoms. MIT researcher Karl Berggren uses circuits made of niobium nitride with liquid helium circulating through the chip to keep it at around minus 257 degrees Celsius [5].

While introducing liquid helium to a conventional chip would increase power consumption, superconducting circuits only require about one percent of the energy in a conventional circuit. Thus, they can afford to use liquid helium to keep the chips at extremely cold temperatures. The energy savings is very desirable for the quantum computing market. Among other functions, the superconducting circuits would be used to read the state of the qubits in the quantum computer.

I believe that the article highlights the importance of superconducting circuits in quantum computing. Due to their speedy calculations of 770 gigahertz and 100 times more energy efficient than the standard chip, they are an essential factor in making quantum computers marketable and sustainable. One interesting point is that these devices feature three terminals, which is a big advantage over the common Josephson junction (two terminals).

VI. Semiconductor Impurities: Noise Redcution

In recent news, researchers at the University of New South Wales have made an exciting discovery in noise reduction. The qubit is more easily read (decoded) by the superconducting circuits when less noise is present in the system. At a quantum level, Ludwik Kranz (Ph. D. at UNSW) discovered that much of the noise in quantum computers is generated by impurities in the semiconductor materials being used to contain the qubits [6]. If too much noise is introduced into the system, the readings of the qubits will not be as accurate as desired. Accuracy must be nearly perfect to scale up the size of the quantum computer (increase the size of the ion trap array).

To reduce noise, the research team has focused on the manufacturing process of the silicon chips. By carefully creating these chips in a way that decreases the number of imperfections, Dr. Kranz and his team were able to reduce the noise level by a factor of ten. The results are very promising as they recorded the lowest noise on record for any semiconductor qubit. Because silicon is inherently spinless, it will not interfere with the spin of the qubit it contains. Therefore, it is an excellent candidate material for scaling up the size of the quantum computer. However, silicon also is susceptible to charge noise. By spending more time on the silicon fabrication process to reduce the number of impurities in their silicon samples, the record setting noise level was achieved.

The article, in my opinion, details an exciting development in quantum computing accuracy. By focusing on the silicon fabrication process, Dr. Kranz and his team were able to reduce the amount of charge noise per qubit by a factor of ten. As science delves smaller into the nanoparticles, quantum science will govern this realm. Imperfections are easily created on this scale, so careful priority must be made to minimize the number of imperfections if silicon is to be used as a host material. If quantum computing is to become a more sustainable tool, materials science must meet the high standards of silicon purity at the nanometer scale.

VII. Ultrabroadband Quantum Entanglement

As of October 29, 2021, researchers at the University of Rochester have developed a thin-film nanophotonic device that set the record for most entangled photons. The development could be a major leap forward in information processing, data mining, and communication networks. When two photons become entangled, their entanglement resonates at a frequency that can be controlled.

The current method of generating bandwidth between photons is to use light to generate broadband entanglement. The process involves splitting a crystal into very small pieces and dividing the pieces among the photon pairs. The frequencies vary between these pieces and their sum make up the broadband of all the photon pairs. Engineers would have a tradeoff between light generated and the bandwidth of the photon frequency.

The new method involves a thin-film lithium niobate nanophotonic device that uses a waveguide that has electrodes on both of its sides. The device is 600 nanometers thick, which is one million times smaller than the cross-sectional area of a bulk crystal [7]. Because it is so small, the light generated from the photon pair is extremely sensitive to changes in the dimensions of the waveguide. A simple geometric algorithm is used to generate the parameters for the waveguide that will maximize the bandwidth of the light.

In my opinion, the article provides a very recent breakthrough in quantum computing. Being able to maximize the bandwidth of the frequency inherent in entangled qubits without losing energy will broaden the number of applications for quantum computing. Specifically, the breakthrough would allow for a larger field of qubits in information processing and communications. According to the researchers, the device is ready to be deployed for lab testing but not commercial use. They also noted that this device uses lithium niobate which is very expensive to fabricate. A cheaper method of fabrication would need to be created to allow it to be commercially feasible.

There are so many exciting developments involved in quantum computing. Some scientists believe that quantum computers are a very powerful tool that should only be used to solve complex problems. But what if researchers like the ones at MIT and others previously discussed find alternate methods of qubit control and device fabrication? The future for quantum computing is exciting indeed with many large corporations and governments investing billions in research.

VIII. Quantum Computing Application: Black Hole Simulation

The paper, "Verified Quantum Information Scrambling," details the work done by seven colleagues from various universities around the world. The paper details how a seven-qubit quantum computer was used to model a black hole using quantum scrambling. A black hole is a phenomenon that occurs in space when gravity is so strong that even light cannot escape it. Quantum scrambling is when local information is dispersed among multiple quantum entanglements and distributed throughout the entire system. It is a key concept of thermalization, which is the process of achieving a thermal equilibrium through interaction [8]. Specifically, thermalization in quantum scrambling can mimic the chaotic nature of a black hole.

Due to the chaotic nature of entanglement, taking measurements of quantum scrambling in a laboratory setting is quite difficult. One method the researchers used to effectively overcome this issue was by using out-of-time-ordered correlation functions (OTOC). The method involves using a quantum circuit to teleport a quantum state through the circuit, which provides an excellent way to indirectly read the state of a qubit. As previously mentioned, directly reading a qubit will alter its state, which is undesirable. The researchers created this scrambling circuit using a 3-qubit unitary operation on a 7-qubit circuit that was part of an ion trap.

Fig. 3. Artist's Rendition of a Black Hole in Space [16]

In my opinion, the article represents a unique cosmic application where quantum computing is used. Nature is full of events that are difficult to test or replicate in a lab due to their chaotic behavior. Due to the chaotic nature of qubits, quantum computers can be used to model these events in a laboratory setting for further study. The researchers in the article used a 7-qubit scrambling circuit to teleport the state of one qubit to the state of another. The teleportation mechanism successfully models the behavior of a black hole in space and is an excellent tool for measuring the randomness of other natural phenomena. Also, the circuit can be used to detect noise in a quantum system better than simply reading the state of an entangled photon pair, since it acts as a completely passive probe. Lastly, by sending qubit states through the scrambling circuit, one can simulate what it would be like to send a probe into space to model the geometry of a black hole.

IX. Error Correction & The World's First 1,000 Qubit Computer

Due to the volatile nature of qubits, reading them incorrectly can result in incorrect calculations. In October 2021, scientists at the University of Sussex, England have designed a quantum computer that designates nine qubits for error detection. Because of entanglement, quantum information (ion energy levels) is stored in pairs. The physics provides a redundancy that can be used for error detection and correction [10]. If a qubit has a value different that its copy, the nine error detection bits are used to recopy the correct value from the original qubit to its working copy.

For quantum computers to become more viable in the marketplace, many hurdles must be overcome; manufacturing costs, performance guarantees, and scalability are but a few of the important items on the list. The largest known quantum computer, revealed September of 2020, was built by IBM, and only contains 65 qubits. To perform more complex calculations, the number of qubits must be increased. By 2023, IBM has promised to make a quantum computer containing 1000 qubits. They are currently playing catchup with Google, whose 53-qubit quantum computer broke the "quantum supremacy" threshold by solving a problem that no supercomputer in existence could ever solve. The size of 1,000 qubits according to IBM would still be 1000 times too small, as they look for viable methods of breaking internet encryptions [9].

I think that both articles discuss recent breakthroughs in quantum computing: error detection/correction and increasing the number of qubits in the ion trap array. The world's largest technological companies are working to advance quantum computing to solve a myriad of unsolvable problems such as hacking internet encryption. These problems will require much more than 1,000 qubits to solve but building a quantum computer containing that many qubits is still an important milestone that must be achieved. While scalability is a primary concern, error avoidance and correction are essential features needed to detect whether the solutions calculated by these computers are correct.

X. CLOPS: Circuit Layer Operations Per Second

One of the pitfalls of the early stages of technological advancement is overhype and performance embellishment. To mitigate these issues, a standard metric must be defined that accurately measures a system's performance. To accurately assess a quantum computer's performance, IBM has recently defined a new metric for quantum speed: Circuit Layer Operations Per Second (CLOPS). The metric measures the number of superconducting circuits that can be addressed by the quantum processing unit (QPU) per unit time. IBM has defined the three biggest factors in quantum computing as speed (CLOPS), scalability, and quality (accuracy). Back in 2017, as they were working on increasing the number of qubits in a quantum system, IBM also coined the term "quantum volume" to define the number of qubits a system can support [11].

CLOPS measures the speed of the superconducting circuits when performing a calculation as well as translating that result to a classical computer. The delay time of the superconducting circuit is primarily the time to access a given quantum slot (array) and the time to assemble the circuits, preparing them to run. In numerous tests, IBM has found that smaller quantum computers tend to have a higher CLOPS. The metric, therefore, depicts more than just the volume of the system, but the performance of the hardware in use. As mentioned earlier, there are different methods of qubit control. As they develop, it will be interesting to test their effect on the CLOPS of the system. IBM has also implemented a new architecture "Qiskit Runtime" which places the runtime environment closer to the quantum processor. The architecture reduces the latency at runtime dramatically because communication between the quantum computer and classical computer has been improved. One test showed that adding Qiskit to a system decreased the runtime from 45 days to 9 hours.

Many emerging companies claim to have developed speedy quantum systems, but how would one definitively know if this were true? In my opinion, CLOPS and other well-defined metrics are essential to accurately measure quantum performance. As quantum technology improves, there will be new methods of qubit control and new architectures that interact with classical systems. Emerging technological advancements must be tested against a known metric, such as CLOPS so that scientists can better understand their nature and behavior in a quantum system. For example, a well-known company that develops quantum systems, Honeywell, claimed to have developed the fastest quantum computer with a quantum volume of 128 using trapped ions. But how well does it communicate with a user's classical computer? What is the delay time between accessing quantum slots and circuit preparation? CLOPS will measure not only the speed of the calculations but also the architecture of the whole system.

XI. Implementation: Grover's Algorithm

Through the power of the cloud, IBM has allowed access to their quantum computers to all registered account holders. Simply create an account to gain an access token and voilà. IBM has integrated several quantum algorithms to use in conjunction with Python version 3.8 (as of December 2021).

One algorithm that is implemented in the IBM quantum library is Grover's Algorithm. The algorithm takes two items as its input: a function and an output. The function is to be applied to many inputs. The algorithm will provide with high probability the input(s) that produce the desired output.

The algorithm works by constructing a circuit that will manipulate the number of qubits needed to solve the problem. First, the qubits are superimposed using what is called a Hadamard Gate so that their states are represented as complex values. Once in value form, all the inputs can be simultaneously fed through the circuit. The circuit will flip the sign(s) of the state vector(s) of the inputs that satisfy the circuit so that they stand out. Then before returning the answer(s), the algorithm will square the state vector to calculate their probability of occurring.

At this point, the state vectors need to be amplified through a process called amplitude amplification. First, they are reflected around a vector that is orthogonal to the winning vector. After each reflection, the probabilities of the winning state vectors increase, and the probabilities of the losing state vectors decrease. The number of necessary reflections to differentiate the winning states from the losing states is $\sqrt{2^q}$, where q is the number of qubits in the circuit.

In a classical machine, performing operations to find which inputs provide the desired output would take O(N) time, where N is the size of the input. However, using Grover's algorithm which utilizes amplitude amplification, the quantum circuit needs to be executed just $O(\sqrt{N})$ times.

The test below was applied using Anaconda version 3, which is Qiskit's quantum library wrapped with Python 3.8. The six-bit input size is fed through a logic circuit representing the logical expression below. The code is based on a tutorial provided by Qiskit [12].

$$Logical\ Expression\colon \Big(\big((a\&b)\big|(c\&d)\big)\&{\sim}(e\&f)\Big)$$

```python
# Anthony Redamonti
# Advanced Computer Architecture
# Qiskit Logical Expression Test
# 11-4-2021

import time
from qiskit import BasicAer
from qiskit.aqua.algorithms.amplitude_amplifiers import Grover
from qiskit.aqua.components.oracles import LogicalExpressionOracle
from qiskit.tools.visualization import plot_histogram

# a logical expression to be evaluated by Grover's Algorithm
logicalExpression = '(((a & b) | (c & d)) & ~(e & f))'

# initialize Grover's Algorithm with the logic expression
algorithm = Grover(LogicalExpressionOracle(logicalExpression))

# initialize the back end with the quantum simulator provided by IBM
backend = BasicAer.get_backend('qasm_simulator')

# Start the timer
StartTime1 = time.time()

# Execute the algorithm
result = algorithm.run(backend, shots = 1024)

# Stop the timer
EndTime1 = time.time()

# Calculate and display the total time taken by the quantum simulator
```

```
TotalTime = (EndTime1 - StartTime1)
print("The total time for the 6-bit calculation: {0}".format("{:.{}f}".format(TotalTime,
8)))

# Plot the results.
plot_histogram(result['measurement'], title = "Possible Combinations", bar_labels = True,
figsize = (15,5))
```

```
The total time for the 6-bit calculation: 0.41300201
```

Fig. 4. Time of Quantum Simulation

The total number of iterations through the logic circuit was $\sqrt{64} = 8$.

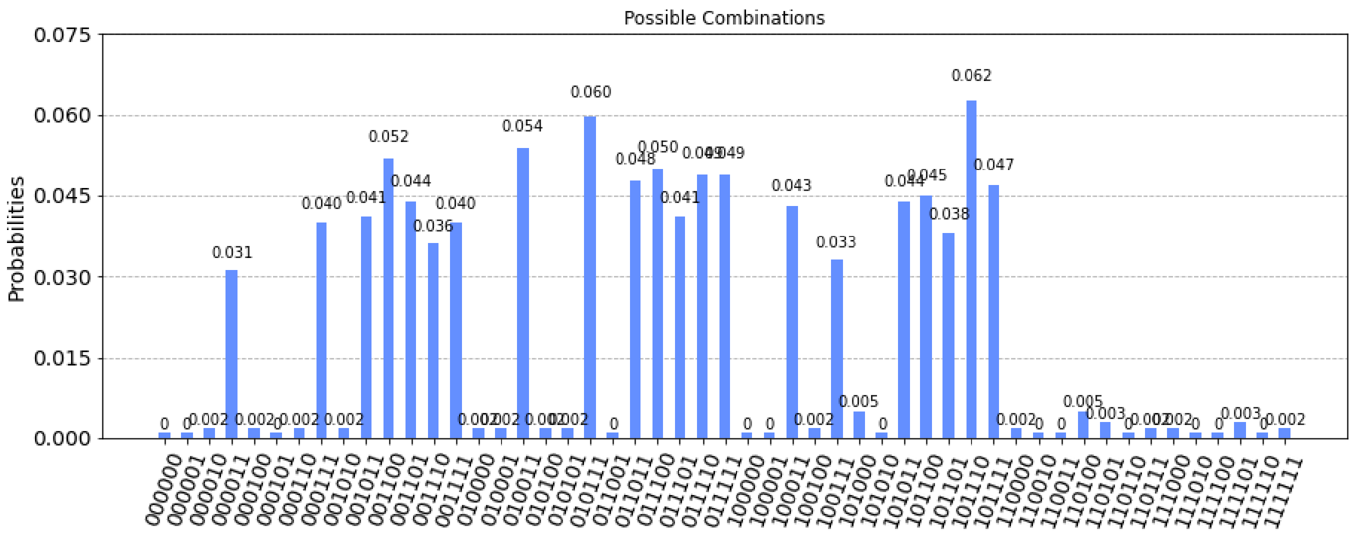

Fig. 5. Histogram of the Inputs vs. Probabilities

The results of the quantum circuit are above. Notice that the probabilities of the winning state vectors are amplified so that they are much greater than those of the losing state vectors. Some quantum circuits have difficulty separating these due to quantum noise, previously discussed. The same test was performed on a classical computer using the program seen below. The same results were achieved, though all 64 test cases were tested.

```
# Anthony Redamonti
# Advanced Computer Architecture
# 11-4-2021
# Final Paper: Quantum Computing

# The following program returns all possible numbers that
# satisfy the logical expression using a classical computer.

import time
import random

possibleCombinations = []

# return the value of the requested bit ('0' or '1')
def getBitValue(integer, bitNumber):
```

```
    twoValue = pow(2, bitNumber)
    if(integer & twoValue != 0):
        return 1
    else:
        return 0

# return the number of combinations that satisfy the logical
# expression.
def testAllCases(highestOrderBit):
    greatestBit = highestOrderBit
    maxNumber = pow(2, greatestBit)

    # iterate through each possible answer and test it using the circuit (if-statement).
    for i in range(maxNumber):
        if(((getBitValue(i, 0) & getBitValue(i, 1)) | (getBitValue(i, 2) & getBitValue(i,
3))) & ~(getBitValue(i, 4) & getBitValue(i, 5))):
            possibleCombinations.append(i)
    return

StartTime1 = time.time()
testAllCases(6)  # test all six-bit numbers through logical expression (circuit)
EndTime1 = time.time()

print("The number of combinations that satisfied the logical expression for a 6-bit number:
" + str(len(possibleCombinations)))
TotalTime = (EndTime1 - StartTime1)
print("The total time for the 6-bit calculation: {0}".format("{:.{}f}".format(TotalTime,
8)))

for i in range(len(possibleCombinations)):
    print(bin(possibleCombinations[i]))
```

The output of the program is on the next page.

```
C:\Python38\python38.exe "C:/Users/aredamonti/projects/Advanced Computer Architecture/Quantum Computing/PossibleCombinations.py"
The number of combinations that satisfied the logical expression for a 6-bit number: 21
The total time for the 6-bit calculation: 0.00100064
0b11
0b111
0b1011
0b1100
0b1101
0b1110
0b1111
0b10011
0b10111
0b11011
0b11100
0b11101
0b11110
0b11111
0b100011
0b100111
0b101011
0b101100
0b101101
0b101110
0b101111

Process finished with exit code 0
```

Fig. 6. Output of the Classical Computer

I was able to find a program [13] written in Qiskit which builds a Hadamard gate from scratch by constructing a quantum circuit. The program searches for the output '01011' given all possible combinations of a five-bit number (0 to 31 in decimal). The program refines the output by sending it back into the Hadamard gate $\sqrt{2^q}$ times. I edited the program so that it will display the results in between each iteration through the Hadamard Gate to illustrate the refining process.

```
# Anthony Redamonti
# Advanced Computer Architecture
# Applying Grover's Algorithm Through Circuitry
# 11-9-2021

import time
import matplotlib.pyplot as plt
from IPython import display

def hold(circuit, state, counts, fig1, fig2, fig3):
    circuit.draw(output='mpl', fold=100, ax=fig1.gca())
    plot_state_qsphere(state, ax=fig2.gca())
    plot_histogram(counts, ax=fig3.gca())
    display.display(fig1)
    display.display(fig2)
    display.display(fig3)
    display.clear_output(wait=True)
    time.sleep(2)

import numpy as np
from qiskit.circuit.library import Diagonal
from qiskit import QuantumCircuit
from qiskit.quantum_info import Statevector, Operator, DensityMatrix, ScalarOp
```

```
from qiskit.visualization import plot_state_qsphere, plot_histogram
from qiskit.converters import circuit_to_dag, dag_to_circuit

# Problem size: width and number of iterations
n = 5
steps = int(np.sqrt(2 ** n))

# Diagonal operators for mark and diffuse
mark_state = Statevector.from_label('01011')
diffuse_operator = 2 * DensityMatrix.from_label(n * '0') - Operator.from_label(n * 'I')
mark_circuit = Diagonal((-1) ** mark_state.data)  # circuit that induces a -1 phase on the
mark_state
diffuse_circuit = Diagonal(diffuse_operator.data.diagonal())  # circuit that reflects about
average = 2|0><0| - I

# A single Grover's step: encode + reflect
mark_and_diffuse = QuantumCircuit(n)
all_qubits = mark_and_diffuse.qubits
mark_and_diffuse.append(mark_circuit, all_qubits)
mark_and_diffuse.h(all_qubits)
mark_and_diffuse.append(diffuse_circuit, all_qubits)
mark_and_diffuse.h(all_qubits)
print(mark_and_diffuse)

% matplotlib
inline

# Draw and update 3 figures: circuit, state, probabilities
fig1 = plt.figure(figsize=(12, 5))
fig2 = plt.figure(figsize=(5, 5))
fig3 = plt.figure(figsize=(15, 5))

# Build and analyze incrementally
grover_circuit = QuantumCircuit(n)
all_qubits = grover_circuit.qubits

# First layer: Prepare uniform superposition
grover_circuit.h(all_qubits)
state = Statevector.from_label('+' * len(all_qubits))

# Keep iterating to amplify the correct amplitude
for step in range(steps):
    # visualize circuit, state, probabilities
    hold(grover_circuit, state.data, state.probabilities_dict(), fig1, fig2, fig3)

    # step circuit
    grover_circuit = grover_circuit.compose(mark_and_diffuse)

    # step state
    state = state.evolve(mark_and_diffuse)
```

The output of the code is below. Notice how the state vector becomes more clearly defined as the number of iterations increases. Also, the probability of the correct output increases while the probabilities of the other possible outputs decrease. For the sake of space, the outputs after the 2nd iteration and last (5th) iteration are displayed below.

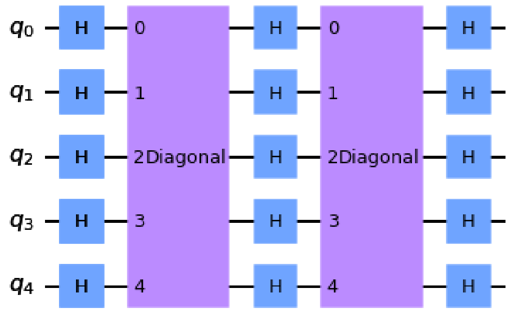

Fig. 7. Circuit Diagram after 2nd Iteration

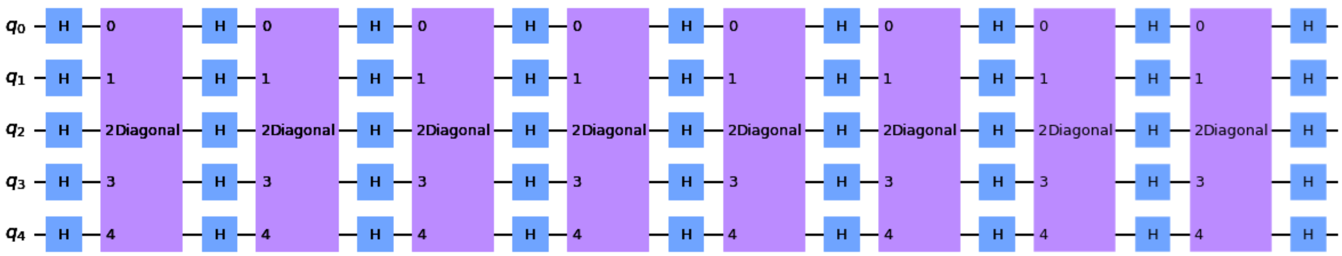

Fig. 8. Circuit Diagram after 5th Iteration

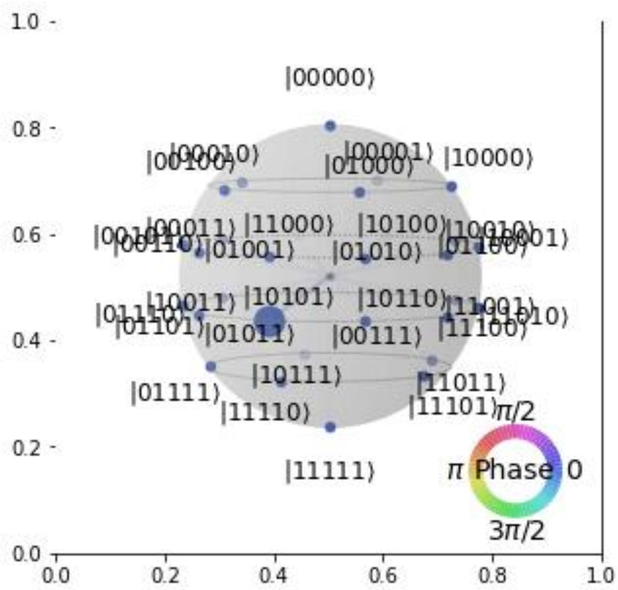

Fig. 9. State Vector after 2nd Iteration

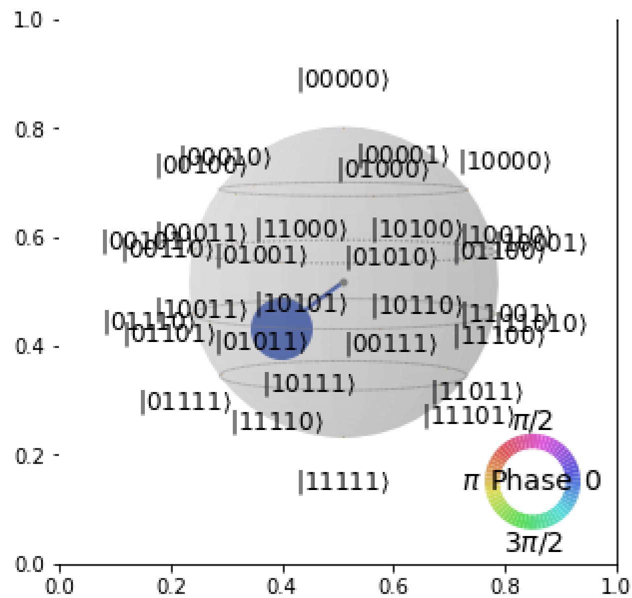

Fig. 10. State Vector after 5th Iteration

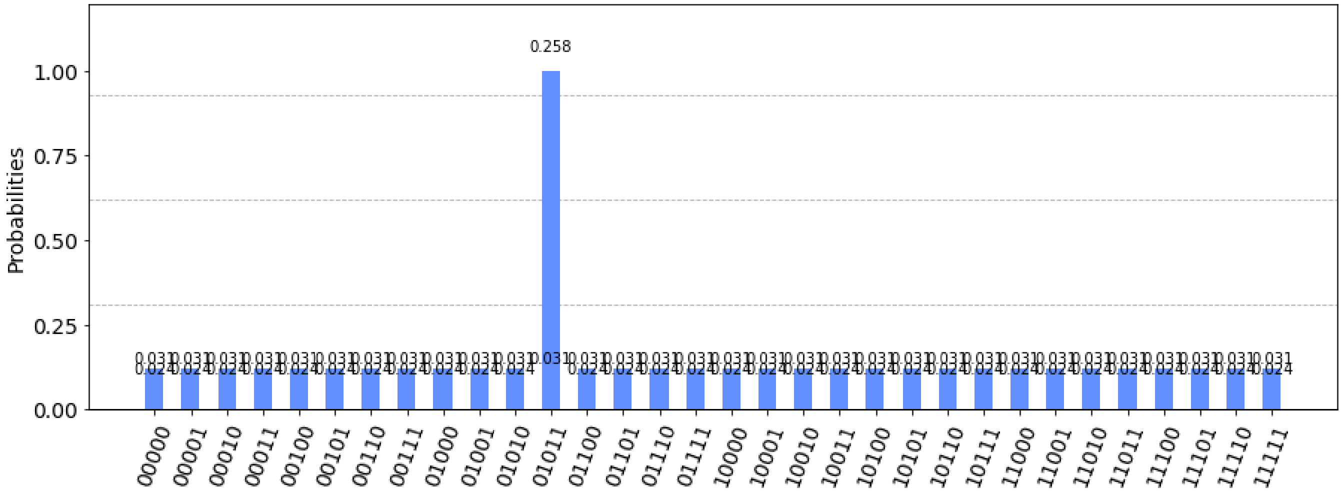

Fig. 11. Histogram of the Inputs vs. Probabilities after 2nd Iteration

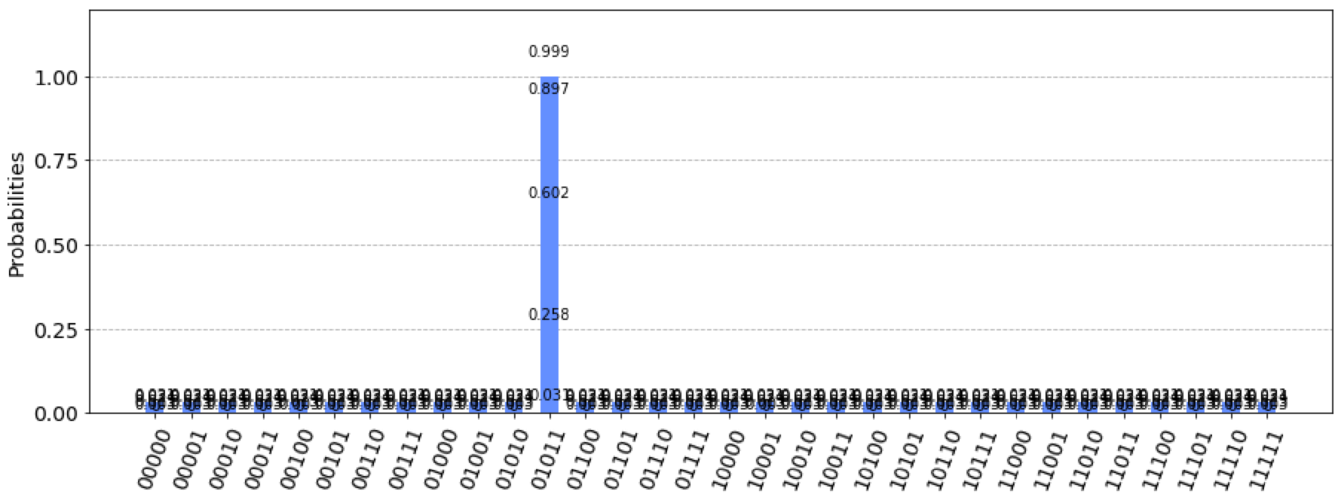

Fig. 12. Histogram of the Inputs vs. Probabilities after 5th Iteration

XII. CONCLUSION

Quantum computers can be used to solve NP problems in polynomial time. The wide range of applications include data modeling, battery design analysis, cancer research, internet encryption hacking, financial modeling, agricultural fertilization, artificial intelligence, astronomical study, drug development, and so much more! The fundamental element of quantum computing is the qubit. Due to its superimposed state, scientists can analyze all possible inputs to an algorithm simultaneously. The algorithms are not coded in a language as in a classical computer but are translated into a superconducting circuit. Quantum computing has a few bottlenecks to overcome such as scalability, performance guarantees (noise reduction), and manufacturing/operational cost (temperature). As new methods of qubit manipulation emerge so too will new terminologies and performance metrics that clearly define the behavior of the quantum system (IBM's CLOPS).

I believe quantum computing could help humanity solve many problems previously thought to be unsolvable. For these machines to justify their cost, their qubit capacity must increase. I found from my testing that there are very few quantum computers available to the public that have a capacity beyond even 10 qubits. Also, it commonly took more than 24 hours to run a program on a real quantum machine, as the number of end-user requests at any given time is very large (IBM's customer queue was very long). Hopefully, as IBM reaches 2023, it will achieve its roadmap goal of the world's first 1,000 qubit machine. The world is booming with excitement as startups and tech giants invest billions of dollars into this rapidly evolving field.

References

[1] Bonsor, Kevin, and Jonathan Strickland. “How Quantum Computers Work.” HowStuffWorks, HowStuffWorks, 8 Dec. 2000, https://computer.howstuffworks.com/quantum-computer1.htm.

[2] Foy, Kylie. “A Trapped-Ion Pair May Help Scale up Quantum Computers.” MIT News | Massachusetts Institute of Technology, Lincoln Laboratory, 28 Jan. 2020, https://news.mit.edu/2020/trapped-ion-pair-may-help-scale-quantum-computers-0128.

[3] Shen, Yu-Ching, and Guin-Dar Lin. “Scalable Quantum Computing Stabilized By Optical Tweezers on an Ion Crystal.” IOP Science, New Journal of Physics, 20 May 2020, https://iopscience.iop.org/article/10.1088/1367-2630/ab84b6.

[4] Savage, Neil. “'Hot' Dots for Quantum Computing.” C&EN, Electronic Materials, 20 Apr. 2020, https://cen.acs.org/materials/electronic-materials/Hotdots-quantum-computing/98/web/2020/04.

[5] Hardesty, Larry. “Superconducting Circuits, Simplified.” MIT News, Massachusetts Institute of Technology, 17 Oct. 2014, https://news.mit.edu/2014/cheaper-superconducting-computer-chips-1017.

[6] “Scientists Create Quietest Semiconductor Quantum Bits on Record.” Centre for Quantum Computation & Communication Technology, PHYS ORG, 8 Oct. 2020, https://phys.org/news/2020-10-scientists-quietest-semiconductor-quantum-bits.html.

[7] Marcotte, Bob. “Researchers Set 'Ultrabroadband' Record with Entangled Photons.” University of Rochester, Science Daily, 29 Oct. 2021, https://www.sciencedaily.com/releases/2021/10/211029114005.htm.

[8] Landsman, K.A., et al. Verified Quantum Information Scrambling. Joint Quantum Institute, 18 June 2018, https://arxiv.org/pdf/1806.02807.pdf.

[9] Cho, Adrian. IBM Promises 1000-Qubit Quantum Computer- a Milestone - by 2023. The Science Journal, 15 Sept. 2020, https://www.science.org/content/article/ibm-promises-1000-qubit-quantum-computer-milestone-2023.

[10] Conover, Emily. Scientists Are One Step Closer to Error-Correcting Quantum Computers. Science News, 4 Oct. 2021, https://www.sciencenews.org/article/quantum-computer-error-correction-multiple-qubits-detect-mistakes.

[11] Leprince-Ringuet, Daphne. Quantum Computing: IBM Just Created This New Way to Measure the Speed of Quantum Processors. ZDNet, 2 Nov. 2021, https://www.zdnet.com/article/quantum-computing-this-is-how-ibm-is-now-measuring-the-speed-of-its-quantum-processors/ .

[12] Marques, Manoel, and Steve Wood. “Using Grover's Algorithm to Perform Quantum Search.” Qiskit-Community-Tutorials, Github, 14 Feb. 2020, https://github.com/qiskit-community/qiskit-community-tutorials/blob/master/optimization/grover.ipynb.

[13] Javadi-Abhari, Ali. “Grover Interactive.” Qiskit-Terra, Github, 20 Apr. 2020, https://github.com/ajavadia/qiskit-terra/blob/Demo/demo/Grover%20Interactive.ipynb.

[14] Dumon, Karel. The Computational Power of Quantum Computers: An Intuitive Guide. 16 Apr. 2019, https://medium.com/@kareldumon/the-computational-power-of-quantum-computers-an-intuitive-guide-9f788d1492b6.

[15] Quantum Dots: Nanocrystals Packed with Potential. Joint Quantum Institute, 20 Jan. 2014, https://jqi.umd.edu/news/quantum-bit/2014/01/20/quantum-dots-nanocrystals-packed-potential.

[16] Fuge, Lauren. Black Holes Exert 'Quantum Pressure' on Their Environment. Cosmos, 15 Sept. 2021, https://cosmosmagazine.com/science/black-holes-exert-quantum-pressure-on-their-environment/.