ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24AUG998

Quantum-Accelerated Flight Selection: Probing Grover's Algorithm and Quantum Device Efficiency

Jayesh Hire[1]
Quantum Computing Developer
TrevasQ Private Ltd
Pune, India

Vaidehi Gawande[2]
Head of Quantum Computing
Creed & Bear Network LLC
Dubai, UAE

Sagar Dhande[3]
Founder
TrevasQ Private Ltd
Pune, India

Abstract:- Current flight search platforms primarily consider four essential factors when planning a trip: departure/arrival dates, as well as the origin and destination locations. However, when additional parameters are added to this search, the problem shifts from a simple to a complex search, as the engine must sift through a massive dataset of flights, including information on airlines, flight routes, fees, and more. To address this challenge and improve flight search efficiency amidst data-intensive and resource-demanding environments, this paper proposes the use of Grover's search algorithm. This algorithm is demonstrated as the optimal solution for searches with increased constraints. The paper highlights the practical application of Grover's search algorithm across three datasets of assorted sizes, as well as various quantum hardware and simulators available in the NISQ era. Furthermore, this paper provides an in-depth understanding of the complexity of quantum circuit design, including the key phases of state encoding and amplitude amplification. The effectiveness of these approaches is evaluated through analysis of execution times and quantum measurement results. The aim is to showcase the potential of quantum computing in revolutionizing real-world search tasks, particularly in the realm of flight selection.

Keywords:- *Quantum Computing, Quantum Search, Quantum Algorithm, Grovers Algorithm, Quantum Application.*

I. INTRODUCTION

As compared to traditional techniques, quantum computing has been shown to considerably speed up computations, making it a rapidly developing subject [1]. This acceleration is especially important since more time-effective data processing methods are required as data volume continues to grow quickly, frequently exceeding the Petabyte scale [2]. As a result, there is a rising desire among computer scientist to apply quantum algorithms on a larger scale [3].

The matrix inversion method Harrow-Hassidim-Lloyd (HHL) is a notable example of a quantum algorithm in the field of machine learning [4]. By using logN quantum bits, or qubits, to represent N-bit classical data, the HHL technique can exponentially speed up matrix inversion when compared to classical solutions [4]. The use of this innovation has considerably increased the speed of matrix inversion in many machine learning techniques. Another notable example is Shor's factoring algorithm, which has demonstrated the ability to factor large numbers exponentially faster than the best conventional methods [5]. This development is crucial for security because Rivest-Shamir-Adleman (RSA) encryption is widely used and relies on the difficulty of factoring large numbers [6].

The process of selecting an optimal flight from a vast data-set is often a time-consuming and daunting task, compounded by a myriad of considerations such as departure and arrival dates, ticket prices, preferred airlines, and the convenience of halts during the journey. In an era marked by rapid technological advancements, quantum computing has emerged as a transformative force offering the potential for exponential acceleration in solving complex problems. Specifically, Grover's search algorithm, conceived by Lov Grover in 1996 [7], promises to revolutionize the process of sifting through extensive data-sets. This algorithm is uniquely equipped to search an unsorted database in quadratically fewer iterations than the most efficient classical algorithms.

This study focuses on obtaining flights that match specific criteria from a large dataset, improving the customer experience, saving processing time, server resources, and cost; and enabling travelers to quickly identify flights that precisely align with their personalized criteria.

It's important to recognize that Grover's search is not just a hypothetical concept, but a tangible technology that has the potential to revolutionize the approach to complex search problems.

In the field of quantum computing, the knowledge landscape is rapidly evolving, driven by the groundbreaking work of researchers worldwide. Quantum computing has the potential to disrupt various industries by resolving challenges that were previously considered intractable. Grover's algorithm is particularly noteworthy for its ability to accelerate search tasks, an essential function in an increasingly data-driven world.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24AUG998

The significance of this work lies in its pursuit of pragmatic applications of quantum computing in everyday life, particularly in the travel sector. As a pioneering effort, this research aims to shed light on how Grover's algorithm can be harnessed to address the intricate problem of flight selection by introducing efficiency, speed, and precision into the process.

In the upcoming sections, the intricacies of Grover's algorithm, the methodology of encoding flight data, the principles of amplitude amplification, and the nuances of quantum circuit design will be explored. Through tangible examples and results, the ability of this quantum approach to expedite and streamline the process of selecting flights that cater to the traveler's unique preferences will be showcased. Ultimately, this research not only underscores the potential of quantum computing but also highlights its transformative impact on decision-making in an increasingly complex world.

II. BACKGROUND

This section aims to provide readers with a concise background to familiarize themselves with key concepts. The discussion begins with a succinct introduction to quantum computing, followed by a focused examination of Grover's algorithm. Finally, the pivotal role of quantum simulators, specifically employed in this paper, in the practical execution of the quantum algorithm is introduced.

➢ *Quantum Computing*

Quantum computing offers a revolutionary change in the field of computation by utilizing quantum physics unusual behavior. It's quite distinct from traditional computing, owing to the fundamental unit of information known as a quantum bit or qubit [8]. In comparison to classical bits, which can only be in a 0 or 1 state, qubits can exist in a superposition of states. This implies that a qubit can at the same time represent 0 and 1, allowing it to carry out many computations in parallel. The real capability of Quantum computing appears in quantum entanglement phenomena, where qubits show correlation with each other unlike classical bits.

Key principles of quantum computing include:

- *Superposition:*

Qubits can exist in a linear combination of states, which enables computation in parallel. A 3-qubit quantum register, for example, can correspond to and manage all eight possible 3-bit binary strings at the same time.

- *Entanglement:*

Quantum entanglement appears when two or more qubits become paired in such a way that the state of one can not be defined independently of the state of the others. This feature allows qubits to work together coherently, opening up new computational possibilities.

- *Interference:*

The wave-like characteristics of quantum states give rise to quantum interference, which enables the constructive or destructive combination of probability amplitudes. This property plays a crucial role in manipulating probabilities within quantum algorithms, providing computational benefits compared to classical approaches.

For a comprehensive understanding of quantum computing, consider referring to the book titled "Quantum computation and quantum information" [9].

➢ *Grover's Algorithm*

Grover's algorithm, proposed by Lov Grover in 1996, is a quantum algorithm designed to excel at unstructured search problems [7]. Locating a specific item in a database without any built-in structure or organization is known as unstructured search. Grover's algorithm stands out by its ability to search a database of N items in only $O(\sqrt{N})$ iterations, which provides a quadratic speedup compared to classical algorithms, which typically require O(N) iterations [10].

The problem is to find the index of the target flight in a list with $N = 2^n$ elements (flights), where N is the list size and n are the number of qubits.

The key components of Grover's algorithm include:

- *Oracle:*

In Grover's algorithm, the target item to be found is represented by a "marked" state. The oracle's role is to mark or identify this state within the superposition of states created by the quantum algorithm. It does so through a series of quantum gates that conditionally modify the quantum state to single out the marked item.

- *Amplitude Amplification:*

After marking the target state, Grover's algorithm applies amplitude amplification, a process that boosts the probability of measuring the marked state while reducing the probability of measuring other states. This amplification phase makes Grover's algorithm remarkably efficient.

- *Measurement:*

The final step involves measuring the quantum state to determine the marked item's location. The algorithm is executed repeatedly to increase the likelihood of getting the right answer.

The Grover algorithm's schematic circuit is displayed in Fig. 1. Approximately $\frac{\pi}{4}\sqrt{N}$ Oracle and Amplitude Amplification iterations later, the target element is expected to be found with a high probability [11].

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24AUG998

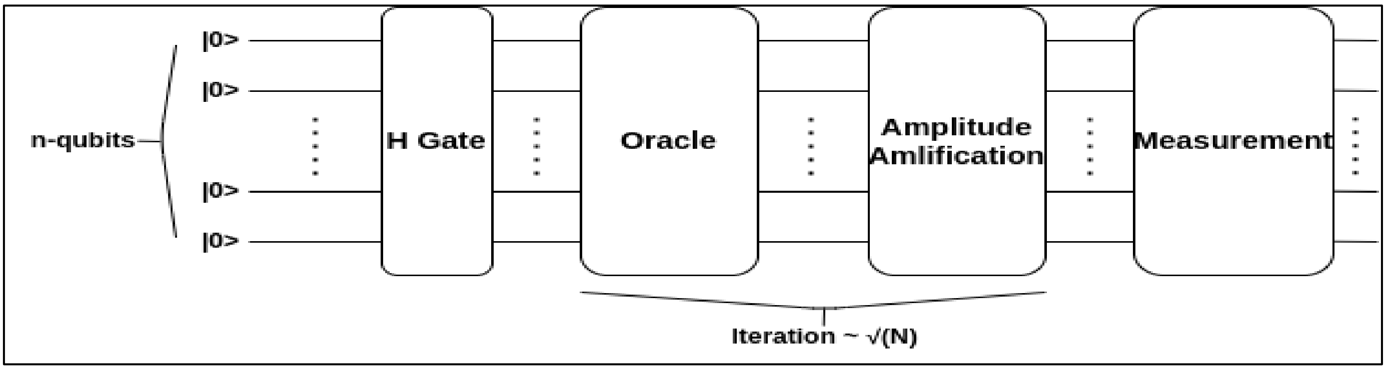

Fig 1 Schematic Circuit for the Grover's Algorithm

Grover's algorithm showcases the quantum advantage by efficiently solving unstructured search problems, making it a powerful tool for optimization and search tasks. Its quadratic speedup over classical search algorithms has applications in various fields, including cryptography, database search, and artificial intelligence [16] [9].

➢ *Quantum Computing Simulators*

The quantum circuit has been executed on five common simulators, each offering unique features and capabilities, as shown in Fig.6. These simulators are briefly described below:

- *Qiskit:*

Qiskit is a quantum computing framework created by IBM that is open source. It provides an intuitive interface, comprehensive functionality, and supports both simulation as well as execution on actual quantum hardware. A QASM simulator within Qiskit was utilized in this study for prototyping and testing the algorithm [12].

- *Cirq:*

Cirq is another open-source quantum computing framework focusing on near-term algorithms for quantum computation. It facilitates the creation, manipulation, and optimization of quantum circuits. The built-in simulator of Cirq was employed in our experiments to validate the design of the quantum circuit [13].

- *Pennylane:*

Pennylane is an open-source library that combines classical machine learning with aspects of Quantum Computing allowing the construction of training neural networks while integrating with popular frameworks. Two CPU-based simulating kernels provided by Pennylane, namely default.qubit and lightning.qubit, were used to simulate the algorithmic setup [14].

- *ProjectQ:*

ProjectQ stands out due to its compilation capabilities alongside simulations, it has support running over numerous varied service provider devices such as those offered through AWS Bracket Azure Quantum or else also includes ion q service amongst others from IBM itself. For conducting experiments, the MainEngine backend located within the ProjectQ simulators suite was utilized [15].

III. MOTIVATION

The motivation for applying Grover's algorithm to flight data search stems from the inherent inefficiency of classical search methods, especially when dealing with extensive datasets. Conventional approaches usually require individuals to manually search through many flight choices to locate the most appropriate one that aligns with their preferences and requirements. This manual process can be time-consuming and impractical, particularly when dealing with extensive datasets.

Quantum computing, on the other hand, offers a promising solution to expedite this laborious search process. Grover's algorithm, in particular, has the potential to significantly accelerate the task of finding flights that meet specific criteria. Its unique quantum properties allow for parallel processing and more efficient searching, offering a faster and more streamlined approach to flight selection.

This research aims to explore the practicality of implementing Grover's algorithm for real-world applications, such as flight data search, and assess its ability to provide a faster and more efficient alternative to classical search methods.

IV. QUANTUM CIRCUIT DESIGN

➢ *Data-Set Loading*

Initiating Grover's search algorithm for flight information necessitates loading datasets categorized as small (15Q), medium (17Q), and large (19Q). The small dataset is characterized by its dimensions of 32,768×6 data points, the medium dataset of 131,072 ×6, and the large dataset of 524,288 ×6.

Table 1 presents a small sample from the 15Q dataset, giving an overview of its characteristics and organization.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24AUG998

Table 1 Example Dataset

Flight Number	Departure Date	Arrival Date	Price	Airline	Halt Status
F193148	Aug 13	Aug 13	291	Delta Airlines	0
F92447	Nov 17	Nov 17	355	Qatar Airways	1
F27710	Feb 5	Feb 5	389	Qatar Airways	0
F235148	Aug 12	Aug 12	295	Delta Airlines	0
F51009	Sep 12	Sep 12	217	Air India	1
F56267	Nov 17	Nov 17	235	Emirates	0
F124359	Mar 17	Mar 17	227	Delta Airlines	1
F33161	May 17	May 17	250	Delta Airlines	0
F214542	Jun 16	Jun 16	387	Delta Airlines	0
F50879	Nov 23	Nov 23	229	Emirates	0

It's important to note that the datasets provided for this research are synthetic. They have been generated to demonstrate the functionality and efficiency of Grover's search algorithm in a controlled environment. While the data accurately represents the structure of flight information, it does not correspond to real-world flight data.

➢ *Oracle*

The oracle (sometimes known as the 'black box') in Grover's search algorithm is a fundamental component that is responsible for encoding the information of the target element(flight) into the quantum state. In our implementation it's work by selecting the criteria from the user, such as departure date, arrival date, price, airline, and halt status, to define the parameters of the search and then transform it into a binary string representation.

In the process of identifying a particular flight within the quantum computing framework, a series of critical steps are executed to select and manipulate the quantum state. An initial step involves the user specifying their preferences for a flight, including departure date, arrival date, price, airline, and halt status. Subsequently, the oracle identifies the optimal match for these criteria within the dataset and assigns it as the "target flight."

The pivotal index of the target flight in the dataset is then calculated and translated into binary representation. In this representation, each binary digit corresponds to whether the related condition aligns with the criteria for the target flight. For instance, an index of 7 corresponds to the binary '111,' signifying the fulfillment of all criteria.

The quantum state comprises qubits, which undergo manipulation according to this binary representation. Each digit in the binary string determines whether an X gate, a quantum gate that flips qubits, is applied to the corresponding qubit. This strategic flipping of qubits aligns them with the specified criteria. A flipped qubit corresponds to an unfulfilled condition, while an unflipped qubit signifies the fulfillment of that condition. (seen in fig. 2)

Following the conditional flipping, a quantum operation known as a controlled-MCT (Multicontrol Toffoli) gate is deployed. This gate amplifies the probability of measuring the desired state, specifically the state that corresponds to the target flight [9]. This controlled amplification is indispensable in the subsequent stages of the quantum search algorithm.

To ensure the quantum state returns to its original form after amplification, a meticulous process of restoration unfolds. Conditional X gates are applied once again, precisely reversing the prior adjustments and returning the qubits to their initial states. This meticulous orchestration of quantum operations forms a coherent and efficient pathway in the pursuit of identifying the target flight within the quantum computing framework.

For each flight in the dataset, the complete process of encoding the flight data, flipping qubits according on criteria, amplifying the target state, and restoring the quantum state is repeated. As a result, the quantum state is a superposition of all the flights, with highlighted states signifying flights that meet the user's criteria. This prepared state is then used in the sub-sequent step of Grover's algorithm called amplitude amplification, which increases the likelihood of measuring the target flight when the quantum circuit is executed.

Fig. 2 showcases the oracle circuit for a specific example state within a 4-qubit system, namely {"0101"}. This example state has been chosen to demonstrate the working principle of the oracle as defined above.

Remember here, indexing is from left to right, which means q0, q1, q2, and q3 correspond to the qubits from the most significant bit (left) to the least significant bit (right).

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24AUG998

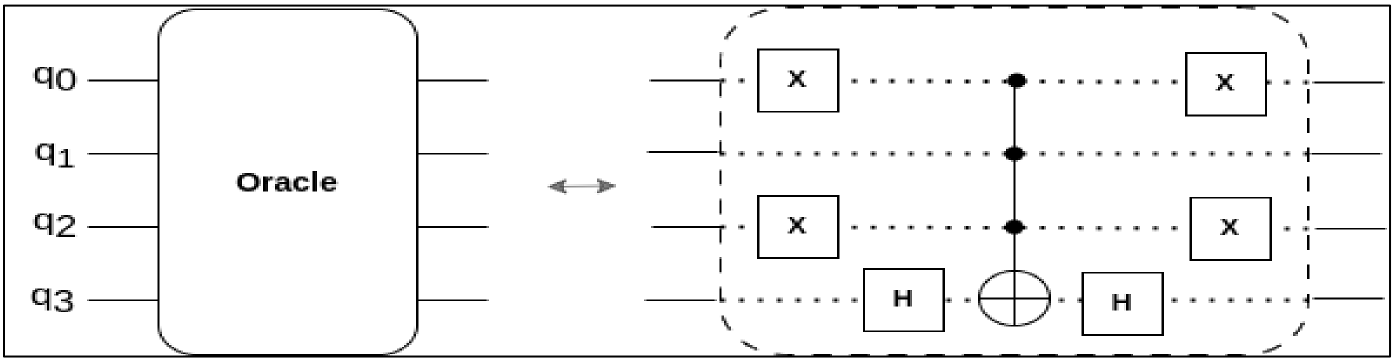

Fig 2 Oracle Circuit

➢ *Amplitude Amplification*

Amplitude amplification is a crucial step in Grover's algorithm, designed to enhance the probability of measuring the desired state while diminishing the likelihood of other states. It involves a sequence of quantum gates meticulously arranged to manipulate the quantum state in a manner that accentuates the presence of the target state.

Let's break down the amplification process step by step:

- *Hadamard Gates (H-gates):*

A Hadamard gate (H-gate) is applied to each qubit in the quantum register, iterating through all n qubits. This H-gate creates a superposition of basis states, which is a pivotal step in Grover's algorithm. It prepares the quantum state to explore multiple possibilities in parallel.

- *X-Gates (NOT-gates):*

Next, an X-gate (NOT-gate) is applied to each qubit. This operation flips the state of each qubit, effectively negating the superposition of states created by the Hadamard gates. The purpose of these X-gates is to isolate the target state and manipulate it separately from the other states.

- *Multicontrol Toffoli Gate (MCT):*

The heart of the amplification process is the Multicontrol Toffoli gate (MCT). This gate is applied to the qubits in a controlled manner. The target qubit is q[n-1], and the control qubits are all the qubits except the last one, q[:-1]. The mode 'noancilla' signifies that no additional ancillary qubits are used in this operation. The MCT gate performs a Toffoli operation on the control and target qubits [9], effectively amplifying the amplitude of the target state. This gate is a crucial part of Grover's algorithm, ensuring that the probability of measuring the target state is maximized.

- *Hadamard Gates (H-gates) - Reversal:*

After the MCT gate, Hadamard gates are applied to the qubits once again. This serves to reverse the superposition created by the initial Hadamard gates. This reversal operation ensures that only the target state has an amplified probability of being measured while the amplitudes of the non-target states decrease.

By applying these quantum gates in a specific sequence, the amplification process systematically enhances the probability of finding the desired state while minimizing the likelihood of measuring other states. This is a central feature of Grover's algorithm, allowing for a substantial speedup in search tasks.

Fig 3 shows the Amplification Circuit for a 4-qubit system, as per our definition of amplification

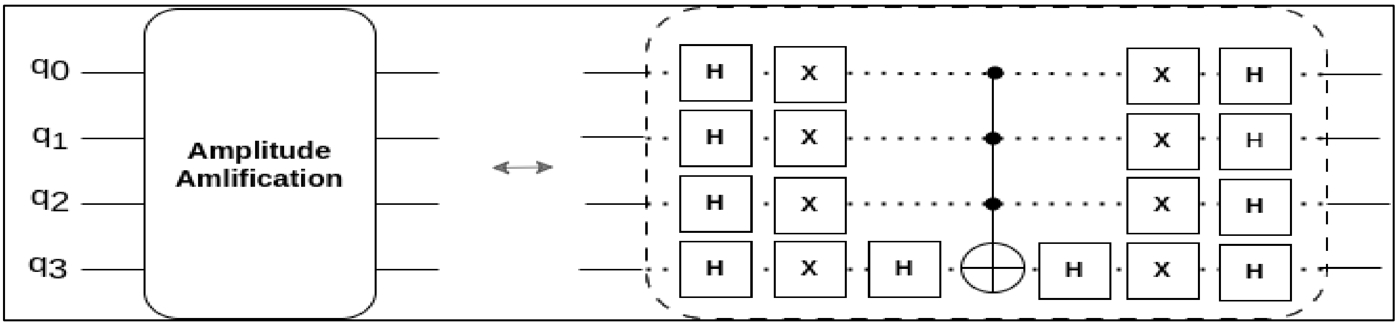

Fig 3 Amplification Circuit

➢ *Measurement*

After applying amplitude amplification, the quantum state is subjected to measurements. The obtained bitstring denotes the index of the flight that most closely aligns with the criteria specified by the user. By repeating this process multiple times and examining the measurement outcomes, it becomes possible to pinpoint the index of a flight that closely meets the user's preferences.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24AUG998

V. RESULT AND DISCUSSION

A comprehensive analysis of the results obtained from implementing Grover's search algorithm on flight data of varying sizes: small (15Q), medium (17Q), and large (19Q) is presented in this section (Table 2). The section discusses the execution times, correlated flight details, and effectiveness of the algorithm, along with the performance of the algorithm on different quantum simulators. Details regarding the computer resources used in this procedure are provided in APPENDIX

Using Grover's algorithm, the small (15Q) dataset was searched for specific flights, yielding accurate results. For example, a flight departing on Jan 10, priced at 394, operated by Qatar Airways, and with a halt status of 1, was in just 2.03 seconds. Similarly, a flight on Aug 6, priced at 250, operated by Delta Airlines, and with a halt status of 1, took 1.54 seconds to find. The effectiveness of Grover's algorithm continued in the medium (17Q) dataset, with searches for flights departing on Jan 1 and Aug 28 yielding results within 5.54 and 6.74 seconds, respectively. Despite a larger dataset size, the algorithm efficiently retrieved matching flights.

In the large (19Q) dataset, Grover's algorithm navigated efficiently, requiring 25.78 seconds to locate a flight departing and arriving on Jan 5, operated by Air India. Similarly, finding a flight departing and arriving on Aug 29, operated by Delta Airlines, took 41.73 seconds. These results highlight the algorithm's resilience and precision across a range of dataset sizes.

Table 2 provides a detailed summary of the performance of the quantum algorithm across different qubit datasets, further supporting the results discussed above. Each dataset's characteristics, search details, and corresponding execution times are presented, offering valuable insights into the algorithm's efficiency and effectiveness.

Table 2 Quantum Algorithm Performance

Qubits	Data Elements	Iterations	Search	Target Details					Execution Time
				Flight Name	Departure Date	Arrival Date	Price	Halt Status	
15	32,768	142	Search 1	F1	Jan 10	Jan 10	394	1	2.0274
15	32,768	142	Search 2	F32768	Aug 6	Aug 6	250	1	1.5404
17	131,072	284	Search 1	F1	Jan 1	Jan 1	232	0	5.5430
17	131,072	284	Search 2	F131072	Aug 28	Aug 28	244	0	6.7395
19	524,288	568	Search 1	F1	Jan 5	Jan 5	381	0	25.7756
19	524,288	568	Search 2	F524288	Aug 29	Aug 29	333	1	41.7278

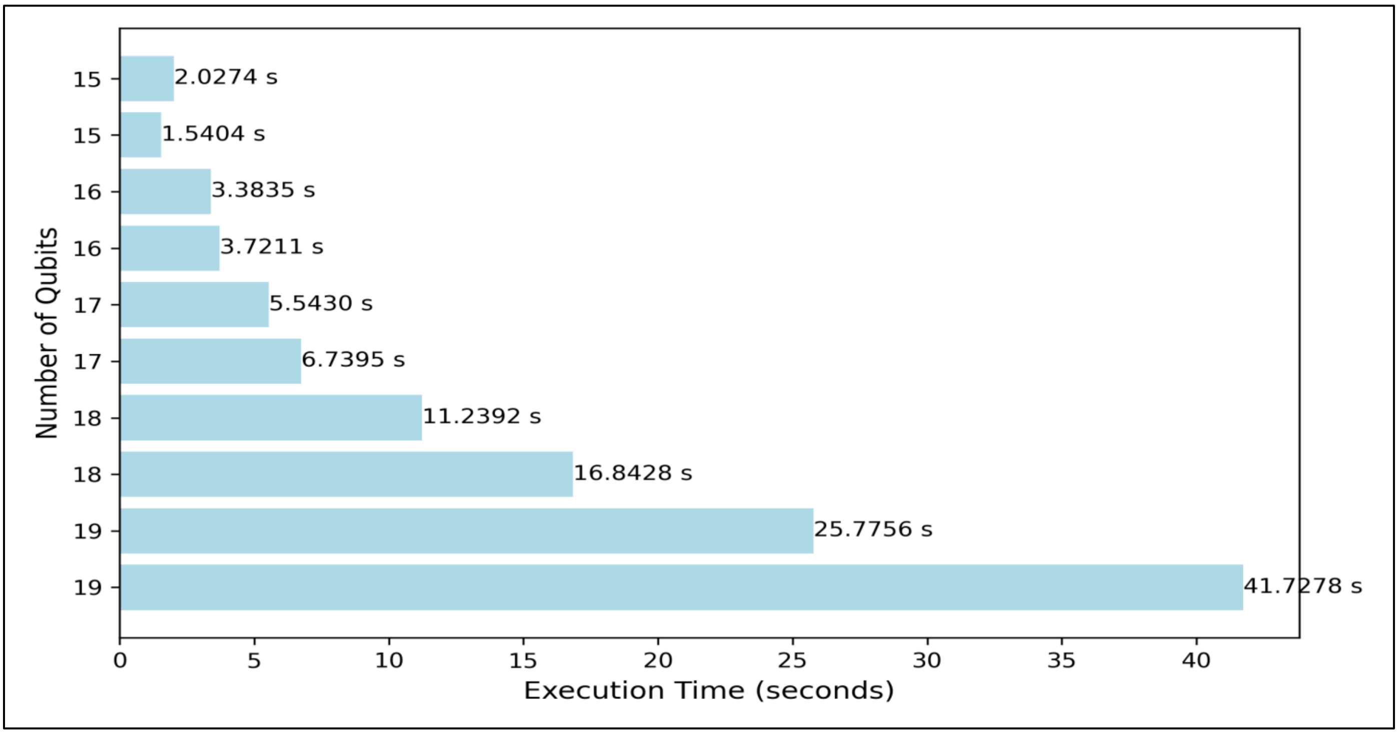

Fig 4 Quantum Flight Data Search.

The histogram plot visually Represents the algorithm's performance within the Qiskit framework across various search scenarios, showcasing execution times for each query and highlighting Grover's algorithm efficiency in handling different flight selection criteria within the 15Q, 17Q, and 19Q datasets (16Q and 18Q datasets also included)

A study on performance was carried out by executing the circuit on different simulators, as illustrated in Fig. 5. Each line on the graph represents a specific simulator, and markers highlight specific qubit counts, offering valuable insights. Notably, it was observed that the processing time for PENNYLANE (lightning.qubit) and Qiskit is quite similar

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24AUG998

compared to other simulators. This discovery indicates that both PENNYLANE (lightning.qubit) and Qiskit demonstrate comparable performance in terms of execution time across varying qubit numbers. The results also provide significant observations; specifically, CIRQ, ProjectQ, and PENNYLANE (default.qubit) exhibit longer execution times compared to other simulators across all qubit Counts-with PENNYLANE (default.qubit) registering the lengthiest time at 19 qubits.

Table 3 presents the data utilized to create Fig 5. For further information about this dataset, please refer to the GitHub link provided in Appendix.

Table 3 Exection Times for Grover's Algorithm using Different Quantum Frameworks and Datasets in Seconds

Framework	15Q	17Q	19Q
CIRQ	3.67	17.60	114.77
Qiskit	2.03	6.74	41.73
PennyLane (default.qubit)	3.23	13.75	136.79
PennyLane (lightning.qubit)	1.12	5.31	39.17
ProjectQ	9.72	30.79	126.83

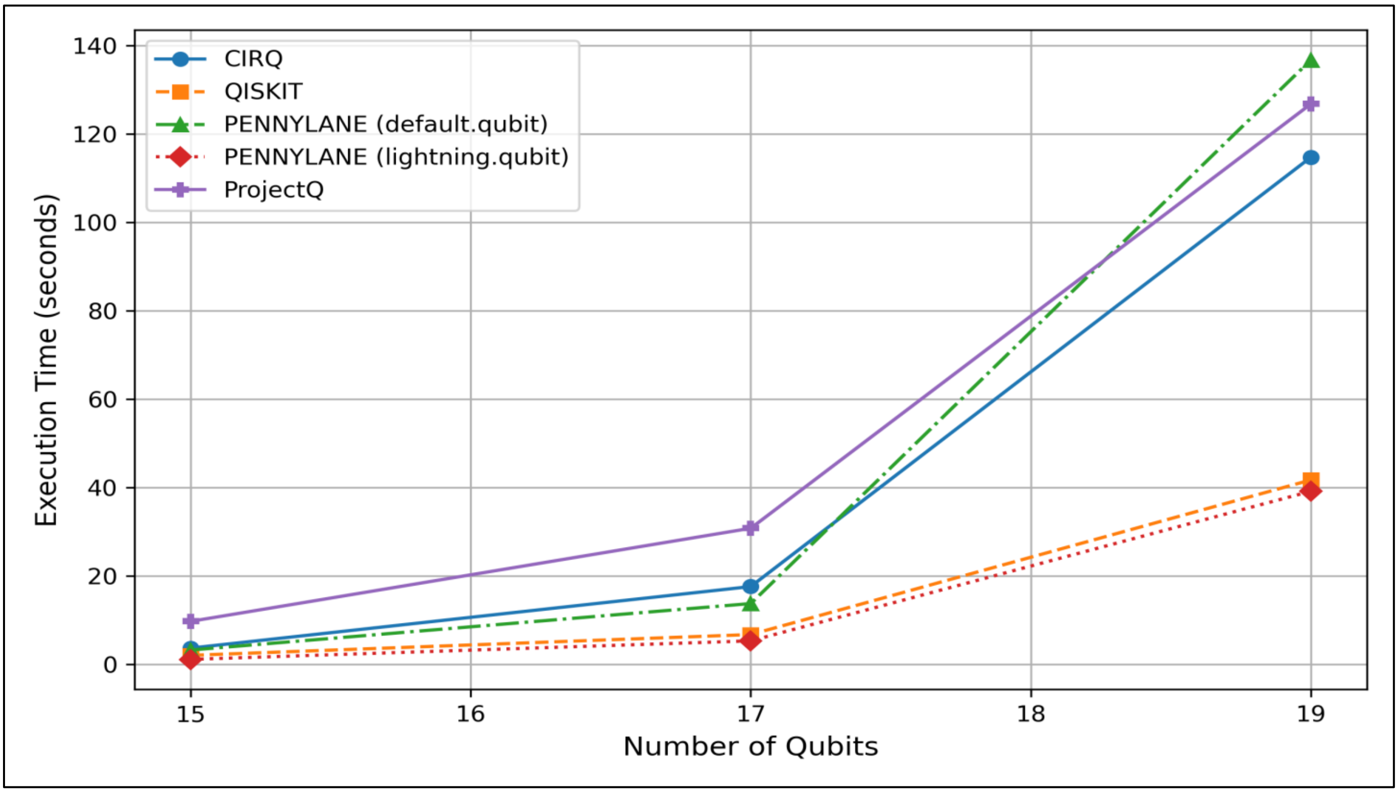

Fig 5 Execution Time vs Qubit Count across Quantum Frameworks

A brief comparison was also carried out between the results from a simulator and those from a quantum device. In this analysis, the '7Q' flight data containing 128 elements were used along with the IBM Quantum computer 'ibm_lagos' [12].

The IBM Quantum Simulator runs the circuit 1020 times to simulate state vectors on a classical computer without factoring in noise. From these measurements, the correct response is obtained 1016 times, with the '1111111' state representing the accurate flight information.

Furthermore, when running the same circuit on a quantum device, no input shows a noticeably high probability, as depicted in Fig. 7 and 8.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24AUG998

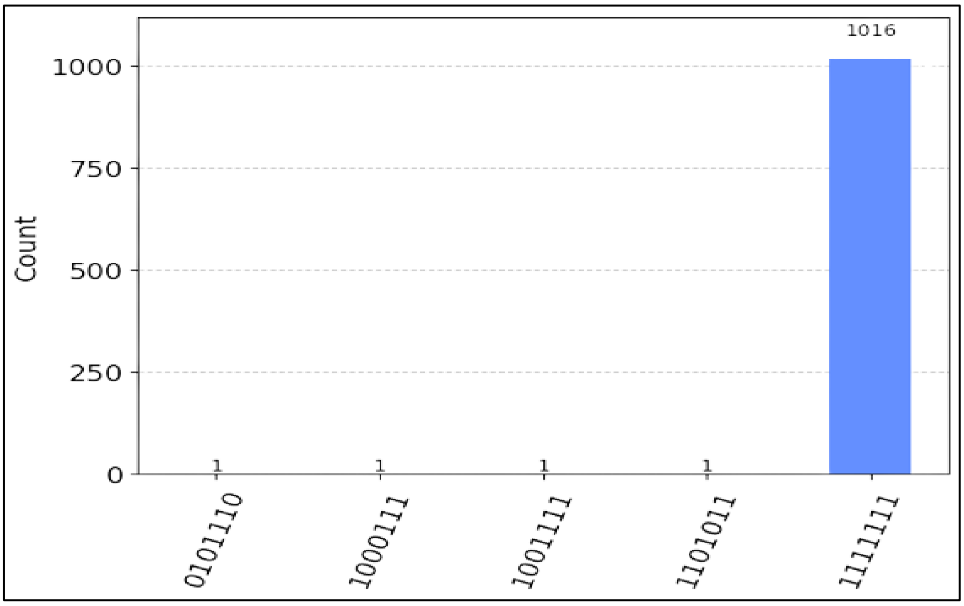

Fig 6 Results from the Simulator

➢ *Fidelity*

When a real quantum device is used, the quantum circuit does not yield consistent results. The results shown in Fig. 7 and 8 indicate clear differences between the output of the simulator and that of the quantum device when running the same circuit. In an actual quantum device, the main challenge preventing the circuit from getting the correct answer is unavoidable noise. Additionally, a study of noise inside IBM's quantum computers shows that two-qubit operators cannot directly act on two distant qubits because of the physical distance between qubits, which reduces connectivity. To finish the operations, they require the assistance of intermediary qubits. It increases the depth of the circuit and the number of gates significantly. As a result, additional noise gets in,

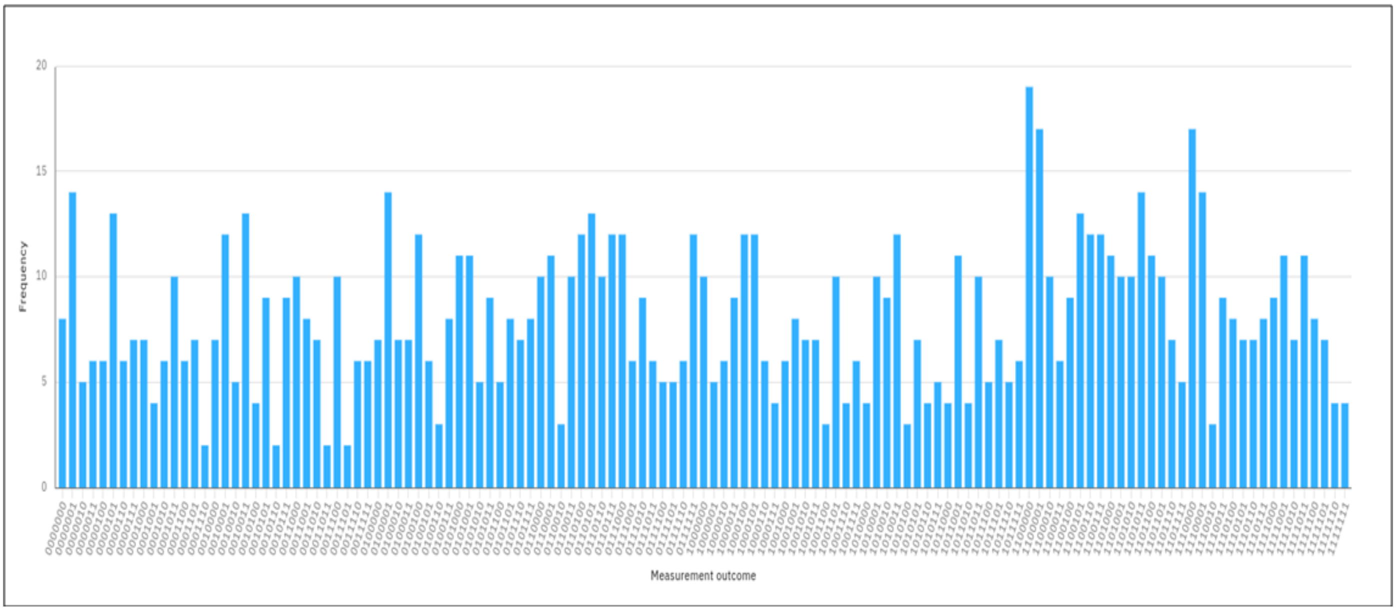

Fig 7 Results from the Ibm _Lagos and the Results become Inconsistent [17]. While Trying to Find a Particular Flight, Utilizing Multiple Control X Gates Extensively Deepens the Circuit and Distorts the Result.

VI. SUMMARY AND CONCLUSION

In this paper, the effectiveness and adaptability of Grover's search algorithm in practical scenarios were assessed using flight data of different sizes: small (15Q), medium (17Q), and large (19Q). The resulting discoveries and perspectives can be succinctly outlined:

- Grover's search algorithm efficiently located flights that matched user-defined criteria within the flight datasets.
- The algorithm demonstrated consistent performance across different sizes of datasets (small, medium, large)
- Despite the increase in dataset dimensions, Grover's algorithm maintained its effectiveness, indicating its potential to handle larger and more complex databases.

- The time taken for execution is found to increase in proportion to the dataset size and is significantly impacted by the construction of an oracle to identify the target item or data entry. Developing a highly effective oracle can be challenging and resource-intensive in certain instances, potentially diminishing the practical benefits obtained from the algorithm.

Additionally, this study offers important perspectives on the potential for quantum computing to revolutionize conventional computational procedures. The reliable capacity of Grover's algorithm to pinpoint data points across different qubit capacities highlights its practicality and effectiveness. With advancements in quantum technologies, it is expected that further enhancements and optimizations will boost its capabilities.

In conclusion, this research underscores the importance of quantum computing as a revolutionary solution for addressing challenges related to data search and optimization. The effective implementation of Grover's algorithm in conducting searches for flight data illustrates the potency and adaptability of quantum computing, which has the potential to be extended to various scenarios requiring efficient data search and retrieval. This makes it a highly appealing choice for industries.

The findings of this study encourage further exploration of quantum computing's potential, and they set the stage for the continued evolution of quantum algorithms and their real-world applications.

REFERENCES

[1]. Ladd, T. D., Jelezko, F., Laflamme, R., Nakamura, Y., Monroe, C., & O'Brien, J. L. (2010). Quantum computers. Nature 2010 464:7285, 464(7285), 45–53. https://doi.org/10.1038/nature08812
[2]. Szalay, A. S., Gray, J., & Vandenberg, J. (n.d.). Petabyte Scale Data Mining: Dream or Reality?
[3]. de Wolf, R. (2017). The Potential Impact of Quantum Computers on Society.
[4]. Harrow, A. W., Hassidim, A., & Lloyd, S. (2009). Quantum Algorithm for Linear Systems of Equations. Physical Review Letters, 103(15), 150502. https://doi.org/10.1103/PhysRevLett.103.150502
[5]. Shor, P. W. (2006). Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. 41(2), 303–332. https://doi.org/10.1137/S0036144598347011
[6]. Rivest, R. L., Shamir, A., & Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, 21(2), 120–126. https://doi.org/10.1145/359340.359342
[7]. Grover, L. K. (1996). A fast quantum mechanical algorithm for database search.
[8]. Yanofsky, N. S. (2007). AN INTRODUCTION TO QUANTUM COMPUTING.
[9]. Nielsen, M. A. ., & Chuang, I. L. . (2023). Quantum computation and quantum information. Cambridge University Press.
[10]. Adedoyin, A., Ambrosiano, J., Anisimov, P., et al. (2022). Quantum Algorithm Implementations for Beginners. ACM Trans. Quantum Comput., 3(4), Article 18. https://doi.org/10.1145/3517340
[11]. Szabłowski, P. J., Paweł, B., Szabłowski, J., & Szabłowski, P. J. (2021). Understanding mathematics of Grover's algorithm. Quantum Information Processing, 20, 191. https://doi.org/10.1007/s11128-021-03125-w
[12]. IBM, Ibm quantum experience, 2023. Available: https: //quantum-computing.ibm.com/
[13]. Google, Cirq, 2023. Available: https://quantumai.google/cirq.
[14]. Bergholm, V., Izaac, J., Schuld, M., et al. (2022). PennyLane: Automatic differentiation of hybrid quantum-classical computations. PennyLaneAI. https://github.com/PennyLaneAI/pennylane/
[15]. Steiger, D. S., Häner, T., & Troyer, M. (2018). ProjectQ: An Open Source Software Framework for Quantum Computing. https://doi.org/10.22331/q-2018-01-31-49
[16]. Mandviwalla, A., Ohshiro, K., & Ji, B. (2018). Implementing Grover's Algorithm on the IBM Quantum Computers.
[17]. Johnstun, S., & van Huele, J.-F. (2021). Understanding and compensating for noise on IBM quantum computers. American Journal of Physics, 89(10), 935–942. https://doi.org/10.1119/10.0006204

APPENDIX

➢ *Computer System :*

The quantum circuit described in this paper was implemented on a computational platform, the specifications of which are outlined subsequently.

- Computer Model: Acer Aspire A515-52G
- Processor: Intel® Core™ i5-8265U CPU @1.60GHz × 8
- RAM: 8 GB
- Operating System: Ubuntu 22.04.2 LTS
- Python Environment: Python 3.10, Qiskit 0.44.1

This information provides transparency regarding the computing resources used and can help readers understand the computational context of the work. It's especially important when working with quantum computing, as hardware and software environments can significantly impact the results and execution times of quantum algorithms.

➢ *Supplementary Materials:*

For additional insights and access to the raw data used for plotting Fig. 6, refer to the GitHub repository link: [https://github.com/Jayesh1211/Result-analysis-]

➢ *Additional Outputs :*

- *For 15 Qubits:*

✓ Target Departure Date: Dec 17
✓ Target Arrival Date: Dec 17

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24AUG998

- ✓ Target Price: 205
- ✓ Target Airline: Qatar Airways
- ✓ Target Halt Status: 0
- ✓ Execution Time: 1.6980 seconds
- ✓ Matching Flight Found at Index: 14759
- ✓ Flight Details:

  - Flight Number: F14760
  - Departure Date: Dec 17
  - Arrival Date: Dec 17
  - Price: 205
  - Airline: Qatar Airways
  - Halt Status: 0

- ✓ Result: 111001011001110
- ✓ Measured Index: 14759

- *For 17 Qubits:*

- ✓ Target Departure Date: May 28
- ✓ Target Arrival Date: May 28
- ✓ Target Price: 209
- ✓ Target Airline: Emirates
- ✓ Target Halt Status: 0
- ✓ Execution Time: 6.4555 seconds
- ✓ Matching Flight Found at Index: 66652
- ✓ Flight Details:

  - Flight Number: F66653
  - Departure Date: May 28
  - Arrival Date: May 28
  - Price: 209
  - Airline: Emirates
  - Halt Status: 0

- ✓ Result: 00111010001000001
- ✓ Measured Index: 66652

- *For 19 Qubits:*

- ✓ Target Departure Date: Nov 5
- ✓ Target Arrival Date: Nov 5
- ✓ Target Price: 206
- ✓ Target Airline: Air India
- ✓ Target Halt Status: 0
- ✓ Execution Time: 30.9212 seconds
- ✓ Matching Flight Found at Index: 249754
- ✓ Flight Details:

  - Flight Number: F249755
  - Departure Date: Nov 5
  - Arrival Date: Nov 5
  - Price: 206
  - Airline: Air India
  - Halt Status: 0

- ✓ Result: 0101100111110011110
- ✓ Measured Index: 249754