Grover's Algorithm for Data Lake Optimization Queries

Mohamed CHERRADI, Anass EL HADDADI
Data Science and Competitive Intelligence Team (DSCI)
ENSAH, Abdelmalek Essaadi University
Tetouan, Morocco

***Abstract*—Now-a-days, the use of No-SQL databases is one of the potential options for storing and processing big data lakes. However, searching for large data in No-SQL databases is a complex and time-consuming task. Further, information retrieval from big data management suffers in terms of execution time. To reduce the execution time during the search process, we propose a fast and suitable approach based on the quantum Grover algorithm, which represents one of the best-known approaches for searching in an unstructured database and resolves the unsorted search query in O ($\sqrt{n}$) time complexity. To assess our proposal, a comparative study with linear and binary search algorithms was conducted to prove the effectiveness of Grover's algorithms. Then, we perform extensive experiment evaluations based on ibm_qasm_simulator for searching one item out of eight using Grover's search algorithm based on three qubits. The experiments outcomes revealed encouraging results, with an accuracy of 0.948, well in accordance with the theoretical result. Moreover, a discussion of the sensitivity of Grover's algorithm through different iterations was carried out. Then, exceeding the optimal number of iterations *round* ($\frac{\pi}{4}\sqrt{N}$), induces low accuracy of the marked state. Furthermore, the incorrect selection of this parameter can outline the solution.**

Keywords*—Big data; data management; information retrieval; quantum computing

I. INTRODUCTION

In the last decades, database management systems have occupied a significant area in IT due to their efficiency in managing massive amounts of heterogeneous datasets. Indeed, the investigation of database research leads to the evolution of special concepts, processes, and algorithms. However, big data lake, recent big news, depicts a recommended solution for dealing with heterogeneous datasets in any format, structured, semi-structured, or unstructured. Thus, numerous contributions, such as No-SQL databases, have been offered for the optimization of processing times on the Big Data Lake [1] [2] [3]. Faced with this challenge, this paper aims to investigate the data lake optimization queries through an efficient and powerful approach based on the Grover algorithm. As the volume of data generated grows, the requirements for a large data processing supercomputer has attracted increasing interest due to their various applications. Therefore, using quantum computers as very fast calculators represents one of the hot topics for accelerating big data processing. It allows us to drastically reduce the execution time when searching for data in a large space. The Grover algorithm was introduced as one of the most beneficial algorithms for data lakes.

Although various classical data retrieval methods have been proposed, most of them remain heavy in query execution for a big data space, which is characterized by volume, variety, and veracity, among other v-properties. This constitutes a significant issue, as several applications are defined in large-scale environments with heterogeneous data in which the majority of the data is unstructured, almost 80%. Furthermore, searching in an unstructured database using quantum algorithms is one of the most widely used techniques to speed up classical search algorithms. It allows finding a more generic research solution to a very wide range of problems [4]. The search time required for a database depends on the size of the database and the quantum hardware. Therefore, it turns out that it is necessary to analyze the design of the quantum circuit.

To address the execution latency issue when searching in a challenging big data space. In this paper, we propose to investigate data lake optimization queries using an efficient and powerful approach based on the Grover algorithm, which is the fastest quantum algorithm for searching an unsorted database with a quandratic complexity of O ($\sqrt{(N)}$) time, as opposed to classical algorithms with a linear complexity of $O(N)$ time. Roughly speaking, a standard analogy for Grover's algorithm is to look up the name of a person in a phone book who only knows their phone number. The phone book remains an unsorted database, and a classical search algorithm appears tedious. On average, this would take N requests, or $N/2$ in the worst case, depending on the position of the desired element, with N denoting the number of entries in the telephone annuaire. Yet, if the correlation between phone name and number is encoded or embedded with quantum bits, the search phone number is reduced approximately to $\sqrt{(N)}$ requests. Thus, quantum computing is a fast-evolving domain, and it is reaching significant accelerations compared to classical algorithms [6] [7] [8] [9]. Considering the speed at which data is growing every day, it is necessary to think of powerful algorithms with the ability to process data quickly and efficiently. Based on this, the principal motivation of this research article is to propose the quantum design of the Grover algorithm and benefit from its speed-up to efficiently manage and extract the hidden relevant information from the heterogeneous data lake. Our proposed, implement IBM Quantum Composer to build the Grover quantum circuit. Indeed, IBM provides multiple quantum computers to the public through its IBM cloud service, accessible via the

application programming interface such as Qiskit [5]. The experiments prove Grover's algorithm as one of the most beneficial algorithms for data lakes.

The remainder of this article is organized as follows: Section II provides the necessary background for readers to fully understand our article. Section III presents the different stages of Grover's algorithm. Moreover, results and discussions are examined in Section IV. Finally, we conclude with a summary and some perspectives in Section V.

II. Related Work

In this section, we review some preliminary and necessary background information needed for the readers to fully understand the rest of our article. We start by examining the data lake concept as a storage space for heterogeneous data sources. Thus, we will give an overview of all the concepts related to quantum computing.

A. Data Lake Concept

In the last decades, the amount of data produced every day is absolutely horrible. So-called big data refers to the exponential growth of massive data. In this context, J. Dixon [10] introduced the data lakes concept to address the challenges and issues induced by big data. Among one of the principal issues studied in the literature is metadata management, proposed with the objective of avoiding the transformation of data lakes into data swamps, i.e., useless data [11] [12] [13] [14]. Thus, data lakes have evolved into data management solutions capable of meeting big data needs and producing a high level of advanced data analysis. They accept various data sources and can accommodate a resilient ecosystem for making creative, data-driven business determinations. Also, Data Lake has a data-centric approach, which refers to an architecture in which data is the primary and permanent asset. Therefore, the data lake has developed as a strong and adaptable concept better suited to data analytics, allowing enterprises to take advantage of this complicated data and generate new commercial industrial activities. While traditional ETL is used in data warehouses to prepare data for integration into a structured relational database, ELT (Extract, Load, and Transform) paradigms are used in data lakes to process unstructured data [15] [16] [17]. Data is loaded into the lake "as-is", with no data transformation. This makes it easier to set up jobs because all that is required is a declaration of the origin and destination locations. As a result, one can reduce the time spent on the data transformation phase, which is considered the most expensive stage in any data project, accounting for over 60% of the total time spent on the project.

Since 2016, the contributions of data lakes in both industry and the academic community have been growing. But most of the data lake proposals are abstract and depend on a specific use case. In our case, we will try to project Grover's algorithm into the data lake as being an unstructured data search space. Since this algorithm applies to unstructured data, it adapts perfectly to the data lake to find crucial information stored in the lake.

B. Quantum Computing

Today's conventional computers are marked with "classical bits" (cbits), which are the basic units of data. With one bit, it takes either the value 0 or 1. Yet, this type of computer faces a limit when challenged with a multivariate problem. In this case, each calculation is a unique path to a unique result. Furthermore, classical computers are less efficient in terms of computation compared to quantum computers due to the limits of classical physics principles, which constitute the core of classical computer components [18] [19]. Thereby, due to recent hardware advancements, quantum computing is a rapidly evolving research field. The principles of quantum mechanics enable quantum computers to solve certain classes of problems very quickly compared to classical computers. Such as factorization and searching databases [21] [22] [23]. Moreover, quantum computers are classified as supercomputers because they exploit the strengths of quantum mechanics, including the quantum superposition principle and entanglement [20]. The superposition principle reflects the possibility of considering a quantum system to be in multiple states at the same time. While quantum entanglement defines the correlation between two (or more) quantum particles even though they are distantly separated.

Following the classical nature of the binary bit, the qubit tries to design a superposition of the states $|0\rangle$ and $|1\rangle$. Since a quantum system can be prepared in a superposition state, the quantum computer can perform 2^n calculations in a single physical step, where n represents the number of qubits used during this process [24]. Furthermore, the quantum computer can execute jobs in exponentially fewer steps than a conventional computer. A qubit can be expressed as a unit vector in a complex vector space, C^2. Constantly written in the form of ket and bra, which corresponds to the notation of Dirac [25]. Hence, the qubit at state zero is written as $|0\rangle$ and the qubit at state one is written as $|1\rangle$. $|0\rangle$ and $|1\rangle$ represent the basis vectors in the complex vector space of quantum states. A Bloch sphere, observed in Fig. 1, is used as a geometric representation of the qubit. The state $|1\rangle$ is represented by the south pole of the sphere, while the state $|0\rangle$ is represented by the north pole. A state $|\psi\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle$ is defined as the angle point (θ, ϕ), where α_0 and α_1 validate the normalization condition, i.e. $\alpha_0^{\ 2} + \alpha_1^{\ 2} = 1$. Thus, it is written in the geometric form with $\alpha_0 = cos\ \theta/2$ and $\alpha_1 = e^{i\phi} sin\ \theta/2$. Yet, the Bloch sphere can be very useful as a geometric representation to visualize the quantum state and its transformation.

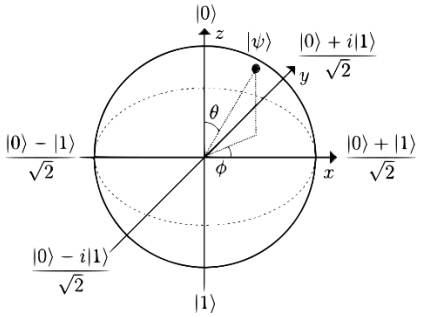

Fig. 1. Bloch Sphere [19].

Further, quantum computing, as one of the rapidly emerging research topics, has fundamentally altered the computing world. Quantum software development continues to be one of the most active investigative study fields [26]. This implies the proposal of new algorithms that adapt to a specific type of new information processing technology. Therefore, quantum computing fascinates the scientific community of researchers because it shows the computing power of big data in a reduced time. Thereby, quantum computing could stimulate scientific progress by leveraging quantum mechanical theories.

However, before we can assess the quantum computer's advantages, several restrictions must be addressed. The most famous one is the decoherence phenomenon, which is the major obstacle. Indeed, to calculate much faster than a conventional computer, the quantum computer uses superposition and entanglement of states that are significantly more sensitive to the environment than classical states [27]. The more qubits you add to a system, the more parallel operations you will increase. Then, since the environment interacts with qubits, quantum measurement uncontrollably changes quantum states. This is called decoherence and is caused by a variety of factors in the environment, including changes in magnetic and electric fields, radiation from nearby hot objects, or uncontrolled interactions between qubits, among others. Subsequently, decoherence affects the state of superposition and disrupts quantum information processing. It is the biggest barrier to the development of quantum technology. Furthermore, it is crucial to examine quantum computing technologies and algorithms.

1) Technologies of quantum computing: The quantum computing field has seen tremendous technological *advancement* over the past decades. Nonetheless, the state of the art related to quantum computing technologies [28] is provided by web giants, such as IBM, Google, Intel, and Microsoft. IBM is one of the major corporations that has made significant investments in quantum computing [29]. At the time of writing this article, IBM had almost 12 simulators, which had up to 5000 qubits, corresponding to a simulator called "simulator stabilizer". Thus, there is a simulator with only one qubit, which corresponds to a simulator called "ibmq armonk". IBM's simulators employ IBM QISKit, a highly handy python library, to process asynchronously run jobs [30] [31]. Qiskit is an open-source framework for quantum computing. It provides the necessary tools that can be used to create and manipulate quantum programs and run them on prototype quantum devices on the IBM Q Experience or simulators on a local computer. Furthermore, once a job process is completed, the user receives the results in the form of the job run time (seconds) and the measurement of each state. Moreover, IBM provides multiple quantum computers to the public through its IBM cloud service. The ibmqx5 is a 16-qubit superconductivity-based quantum computer, ready through an Application Programming Interface (Python-API) called QISKit.

2) Quantum algorithms: In the quantum computing era, a quantum algorithm is a quantum computation solution that works on a practical quantum model [32]. It is often designed as a quantum circuit. Moreover, a classical (or non-quantum) algorithm is a sequence of instructions for solving a problem. One of the most well-known classical search algorithms is that of sequential and interval search.

a) Sequential search: One of the most basic and simplest search algorithms that fall under the category of searches is linear. This type of algorithm works sequentially (without jumping) through a list by comparing each element with the value we want to find [33]. In the worst case, the time complexity corresponds to the order of N, indicated as O(N), where N represents the number of elements in the list. This algorithm has the advantage of not requiring the list to be sorted because it works regardless of the order in which the list's elements appear. However, finding the element you're seeking takes a long time. As long as the number of elements in the list is large, the algorithm takes a lot of time.

b) Interval search: One of the frequently used implementations in interval search is the binary search. In fact, the search space must be ordered. Furthermore, this sort of algorithm divides the collection of elements that make up the search space into intervals [34], such that if the search value is smaller than the value in the middle of the interval, in this case, the search is not performed only at a level of less than half the interval. Otherwise, the search is carried out at the upper level. This process is repeated until the element marked is found in logarithmic time.

III. GROVER'S ALGORITHM

The study of quantum algorithms has recently been one of the most difficult scientific issues that has radically transformed the way people think about computers. Indeed, quantum computing remains a part of worldwide reality, and its advancement cannot be overlooked. Working on this research topic was also the goal of former researchers [35]. The principles of quantum physics, like, for example, the use of superconducting quantum processors, have a major peculiarity [36]. Exercising superconducting quantum circuit technology enables the researchers to contribute a list of contributions related to the quantum algorithms. In 2016, IBM introduced the Quantum Experience program, which provides a set of online quantum simulators that allow anyone interested to execute their quantum circuit [37]. The IBM Quantum Experience handbook gives users a hands-on experience with all of the criteria for building a quantum circuit that solves a specific problem perfectly.

Grover's algorithm offers a quick search through a mass of unstructured data to find the desired information. It has proven a significant speedup compared to the classical algorithm and produced a promising result, motivating extensive investigation into the viability of applying Grover's algorithm to a variety of domains. In this paper, we examine a search space of size N with no prior knowledge of how the data will be presented. This problem has a polynomial complexity with classical solutions, whereas the quantum search algorithm has a quadratic complexity $O(\sqrt{(N)})$ [38]. Through this paper, we have proposed an overview in algorithmic form, summarizing

the different stages of Grover's algorithm. As shown in algorithmic prototype 1.

Algorithm 1 Grover's Algorithm for data lake

Input: Heterogeneous datasets form a data lake $DL = \{x_0, x_1, x_{N-1}\}$

Output: Get the index of the marked element $x^* \in N$

Step 1: The quantum register's initialization:

Set the state of all qubits $x \otimes^n$ to the state $|0\rangle$ and set the oracle qubit to $|1\rangle$ state : $|\psi_0\rangle = |0\rangle \otimes^n |1\rangle$

Step 2: Deploy the register in a distributed uniform superposition:

Apply the Hadamard gate H:

$$|\psi_1\rangle = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} |i\rangle \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

Step 3: Repeat Grover's iterations:

for round($\frac{\pi}{4}\sqrt{N}$) times; do

a- Apply the oracle:

$|x\rangle \rightarrow (-1)^{f(x)} |x\rangle$

b- Execute Grover operator (reflection about the mean)

1. Apply $H \otimes^n$
2. Conditionally shift phase
3. Apply $H \otimes^n$

End for

Step 4: Quantum register measurement

Finding the index of the target element in a list of $N = 2^n$ entries is the problem of searching in an unordered list. With n denoting the number of qubits and N denoting the list's length. Moreover, an unstructured search is commonly expressed as a database search query in which we want to find an item that meets a set of criteria specified in the query. We refer to the problem search as "unstructured" because we have no control over how the data is organized in the database. If we have an ordered database, we can use a binary search to find the predicted element in logarithmic time. However, if we don't know the sequence of the database items, the task remains difficult to complete in terms of execution, and we can't get better results with the conventional approach. If there is no indication of where the desired item might be found, in this case, any classical algorithm must examine each element individually. Furthermore, the number of tries required to find the sought item equals the number of items in the list. As we can see, using quantum mechanics principles, only $O(\sqrt{(N)})$ attempts are required. To meet this requirement, Grover's algorithm uses two registers, the first one linked with quantum qubits, in which we shall create a superposition of all 2^n basis states $\{|0\rangle, ..., |2N-1\rangle\}$. This can be done by applying the Hadamard gate to all the initial qubits. While the second register is linked to classical bits to persist the measurement results, it takes either the value of *0* or *1*. For the sake of precision, we describe the different stages of Grover's algorithm:

A. Initialization

The Grover algorithm starts by initializing the qubits in the state $|0\rangle$ by performing a uniform superposition of all basic inputs. A Hadamard quantum gate, given by the matrix (1), is implemented to create a superposition of the set of quantum states [39].

$$H=\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix} \quad (1)$$

By applying the Hadamard gate to state $|0\rangle$, we obtain the following state.

$$\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}\begin{bmatrix}1\\ 0\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\ 1\end{bmatrix}$$

Which has mapped:

$$|0\rangle \rightarrow \frac{|0\rangle + |1\rangle}{\sqrt{2}}$$

If we instead initialize the qubit to $|1\rangle$ and apply a Hadamard gate:

$$\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}\begin{bmatrix}0\\ 1\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\ -1\end{bmatrix}$$

Which has mapped:

$$|1\rangle \rightarrow \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

As a result, to generalize the Hadamard gate application to the initial state $|0\rangle$, we obtain the following formula:

$$H^{\otimes n} |0\rangle = \sum_{i=0}^{N-1} \alpha_i |i\rangle$$

Where α_i represents the amplitude probability. Indeed, all quantum states have the same amplitude, i.e., $\alpha_i = \frac{1}{\sqrt{N}}$.

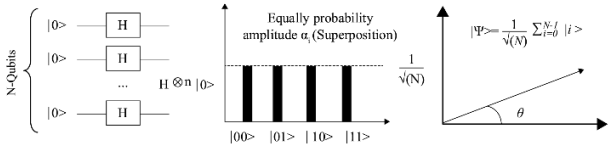

Fig. 2. The Grover Algorithm's Initialization Step.

As illustrated in Fig. 2, for the case of the number of qubits $N = 4$. Therefore, the number of possible states corresponds to $N = 2^n = 2^2 = 4$. Each state is associated with equiprobable amplitudes, $\alpha_i = \frac{1}{\sqrt{4}} = \frac{1}{2}$.

B. Oracle

After having initialized the circuit with the Hadamard gate to create a superposition of quantum states, Grover's algorithm will proceed through its first iteration, which corresponds to what is known as the quantum oracle. The oracle, also known as a "black-box" function, modifies the quantum state of the item's index we're seeking [40] [41]. The change of the quantum state by the oracle Grover was performed without transforming it into a classical state. If the system is located in the right state, then the oracle will turn the phase by the angle π. Otherwise, no action will be taken. The function f corresponds to the oracle expressed as follows:

$$f(x)=\begin{cases}1 & if\ x\ is\ the\ correct\ state\ ,\\ 0 & Otherwise\ .\end{cases} \quad (2)$$

The quantum circuit implements the function *f* described by the unitary operator denoted by *O*.

$$O|\psi\rangle = \sum_i \propto_i (-1)^{f(i)} |i\rangle \quad (3)$$

The function *f* verifies the searched item by transforming the sign of its probability amplitude if *f(x) = 1*. Otherwise, nothing happens. To illustrate the operation of the oracle, we take an example of two qubits. To create a superposition state, we apply the Hadamard gate.

$$|\phi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |01\rangle + |10\rangle + |11\rangle) \quad (4)$$

Suppose that the item we're looking for is the index marked by $|i\rangle = |i^*\rangle = |10\rangle$. By applying the oracle to the state $|\phi\rangle$, we get the state $|10\rangle$ signed by a phase of factor -1. Grover's algorithm oracle step is depicted in Fig. 3.

$$|\phi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |01\rangle - |10\rangle + |11\rangle) \quad (5)$$

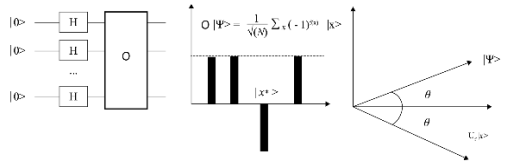

Fig. 3. The Grover Algorithm's Oracle Step.

C. Amplification

The amplification step performs a reflection around the average of the amplitudes. It flips the target state by increasing its amplitude probability and decreasing other states. Yet, this step can be implemented by a combination of the following gates: *HRH*. Here, *H* designates the Hadamard gate, and and *R* designates a phase shift transform [21]. Fig. 4 depicts Grover's algorithm amplification step.

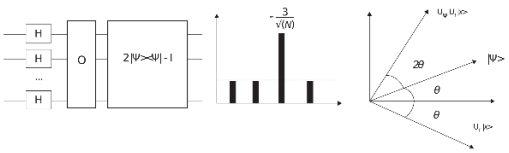

Fig. 4. The Grover Algorithm's Amplification Step.

D. Measurement

The measurement of each qubit in a quantum circuit is performed as the last step of the calculation to produce an output in the classical bit [42]. Indeed, we cannot say that a qubit has an actual value, but rather that it contains a probability of being found in a particular state when measured. Moreover, the measurement step is necessary to derive a result from the quantum state computation. The quantum gate associated with this step (the measurement gate) represents the only non-reversible quantum gate.

IV. Results and Discussions

Suppose we want to find the name of a well-described article with a set of metadata (Fig. 5). Each article that exists is indexed by an integer belonging to segment $\{0, \ldots, N-1\}$. Given that, we're looking for an article with an index $i = I^*$. The articles are not ordered, and we need to get a particular record from the list of articles. If we use the classical algorithm, we may be lucky and find the article we are looking for in the first index, i.e., $i = I^* = 0$, or we may not find the article until the last index $i = I^* = N-1$. Furthermore, for a search in the unstructured database, an average of approximately *N/2* (or *N* in the case where we found the article in the last index) of queries is required to find the article that adapts to the search criteria. It is important to point out that in the case of a uniform probability, we have a probability of *1/N* of finding an article among the *N* articles. Then, we can prove the average number of queries needed to find the right article, according to the equations below.

$$\sum_{i=1}^{N} i\frac{1}{N} = \frac{1}{N}\sum_{i=1}^{N} i,$$

$$\sum_{i=1}^{N} i = \frac{N(N+1)}{2},$$

$$\frac{1}{N}\sum_{i=1}^{N} i = \frac{1}{N}\frac{N(N+1)}{2} = \frac{(N+1)}{2} \approx \frac{N}{2} \quad (6)$$

Grover's algorithm bettered the classical search method by a quadratic speedup. The computer scientist, Grover, found a quantum search algorithm that requires only $O(\sqrt{(n)})$ steps. Suppose, for example, N=1000; the classical search algorithms do 1000 iterations (or 1000/2 = 500 in the worst case) to find the search record. However, the Grover algorithm will only perform $\sqrt{(1000)}$=100 iterations. Consequently, Grover's algorithm exhibits a significant acceleration. We cannot do great than a quadratic speedup with a complexity of order $\sqrt{(N)}$. The *N* articles are numbered from 0 to *N−1*, requiring *n* qubits to represent the list of articles (with $N = 2^n$). We can represent all *N* articles using only the principle of superposition with *n* qubits. A quantum state, $|\psi\rangle$ is designed by a column vector of size $(2^n, 1)$ whose values are probability amplitudes. Each probability amplitude is associated with a well-defined article that is identified by an index *i*.

$$|\psi\rangle = \begin{bmatrix}\alpha_0\\ \vdots \\ \alpha_m\end{bmatrix} \rightarrow \alpha_0|0\rangle + \alpha_1|1\rangle + \quad (7)$$

The article of index *i* is linked to the probability amplitude α_i. As a result, the probability of finding the article we're seeking is extremely close to 1, and the amplitude of all other probabilities is close to 0.

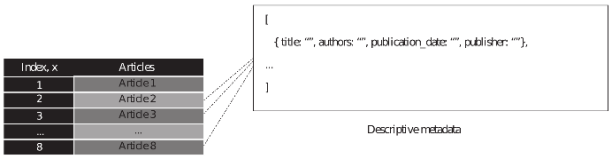

Fig. 5. Searching a List of Articles.

A. Use Case Application

Considering Data Lake (*DL*) as a database that stores heterogeneous data regardless of its format. Then, *DL*, which is made up of *N = 8* data sources, is derived from scientific

databases (Fig. 5). Each data source represents a scientific article, which is identified by a collection of descriptive metadata such as title, authors, and publication date, as well as the path where the paper is placed, which provides the paper's unique identity (*ident*). While $ident \in \{0, \ldots, N-1\}$, we want to find the article titled *X*, which contains an identifier id_k. To meet this requirement, we must first meet the prerequisite:

$f(id_k) = 1$.

Then,

$$O_f |X\rangle = -|X\rangle . \quad (8)$$

Let us express the different quantum states at each step shown in the circuit. Let's start with the first state $|\psi_0\rangle$, and end with the last state $|\psi_f\rangle$.

$$|\psi_0\rangle = |000\rangle \quad (9)$$

Subsequently, by applying the Hadamard gate, the state $|\psi\rangle$ becomes.

$$|\psi_0\rangle = H^{\otimes 3} |000\rangle = \frac{1}{\sqrt{8}} \sum_{i=0}^{7} |i\rangle = \frac{1}{2\sqrt{2}} \sum_{i=0}^{7} |i\rangle . \quad (10)$$

Consider that we are looking for the element which has the index $i = 5$. Then, $|i\rangle = |5\rangle = |101\rangle$ (Fig. 6 depicts the quantum circuit that corresponds to determining the quantum state $|101\rangle$). At this point, we need to specify the oracle operator that will be used in our use case. Indeed, when solving an NP problem, the defined oracle operator can mark the corresponding state. Therefore, the oracle operator must mark the element with the index 101 that we are looking for. Then, we have.

$$O_f |101\rangle|-\rangle = -|101\rangle|-\rangle.$$

$$O_f |i\rangle |-\rangle = |i\rangle |-\rangle \ if \ i \neq 5. \quad (11)$$

After specifying the sought state, we need to define a vector orthogonal to it, denoted by $|u\rangle$ as expressed below:

$$|u\rangle = \frac{1}{\sqrt{7}} \sum_{\substack{i=0, \\ i \neq 5}}^{7} |i\rangle, \quad (12)$$

$$|u\rangle = \frac{|000\rangle+|001\rangle+|010\rangle+|011\rangle+|100\rangle+|110\rangle+|111\rangle}{\sqrt{7}}$$

Then, we have

$$|\psi\rangle = \frac{\sqrt{7}}{\sqrt{8}} |u\rangle + \frac{1}{\sqrt{8}} |101\rangle = \frac{\sqrt{7}}{2\sqrt{2}} |u\rangle + \frac{1}{2\sqrt{2}} |101\rangle. \quad (13)$$

With this equality, one can determine the value of the angle θ as follows:

$$\theta = 2 \arccos\left(\frac{\sqrt{7}}{2\sqrt{2}}\right) \approx 41.4° \quad (14)$$

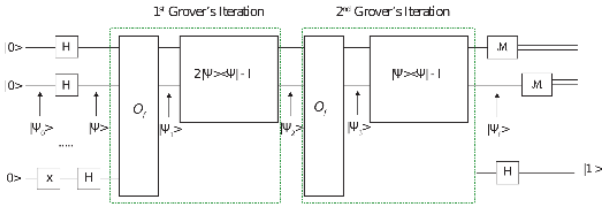

Fig. 6. Grover's Algorithm Quantum Circuit over an Unstructured Database of Eight Elements.

The next step is intended to apply the oracle operator $|\psi_1\rangle |-\rangle$. As a result, we get.

$$|\psi_1\rangle|-\rangle = O_f (|\psi 1\rangle|-\rangle) ,$$

$$= \frac{|000\rangle+|001\rangle+|010\rangle+|011\rangle+|100\rangle-|101\rangle+|110\rangle+|111\rangle}{\sqrt{7}} . \quad (15)$$

Note that quantum state $|101\rangle$ is the only one that has a minus sign. It is now suitable to rewrite $|\psi_1\rangle$ as follows:

$$|\psi_1\rangle = |\psi\rangle - \frac{1}{2\sqrt{2}} |101\rangle. \quad (16)$$

Or it can be expressed as

$$|\psi_1\rangle = \frac{\sqrt{7}}{2\sqrt{2}} |u\rangle - \frac{1}{\sqrt{2}} |101\rangle \quad (17)$$

Eq. (16), is useful for the next step's calculation. Since we are going to apply the formula $(2 |\psi\rangle \langle\psi| - I)$. The formula of Eq. (17), is useful to schematize the quantum state $|\psi_1\rangle$. Yet, the $|\psi_1\rangle$ state represents the reflection of $|\psi\rangle$ respecting the state $|u\rangle$. In the next step, we will apply the reflection again around the average.

$$|\psi_2\rangle = (2|\psi\rangle \langle\psi| - I) |\psi_1\rangle \quad (18)$$

By using Eq. (16), we get

$$|\psi_2\rangle = \frac{1}{2} |\psi\rangle + \frac{1}{\sqrt{2}} |101\rangle \quad (19)$$

Therefore, by using Eq. (13), we get

$$|\psi_2\rangle = \frac{\sqrt{7}}{4\sqrt{2}} |u\rangle + \frac{5}{4\sqrt{2}} |101\rangle \quad (20)$$

To assert that the angle between $|\psi\rangle$ and $|\psi_2\rangle$ is θ, note that

$$cos(\theta) = \langle\psi_2|\psi\rangle = \frac{1}{2} \langle\psi|\psi\rangle + \frac{1}{\sqrt{2}} \langle\psi|101\rangle = \frac{3}{4} \quad (21)$$

Which conforms with equality (14). This completes the first iteration of the Grover application designated by *G*. The second and final application of the Grover operator is similar to the first one. The next step in our examination is the analysis of the state $|\psi_3\rangle$, which is found by applying the oracle operator, as shown below:

$$|\psi_3\rangle = \frac{\sqrt{7}}{4\sqrt{2}} |u\rangle - \frac{5}{4\sqrt{2}} |101\rangle \quad (22)$$

Using Eq. (16), we get

$$|\psi_3\rangle = \frac{1}{2} |\psi\rangle - \frac{3}{2\sqrt{2}} |101\rangle \quad (23)$$

It is important to note that the state $|\psi_3\rangle$ represents the reflection of the state $|\psi_2\rangle$ with the state $|u\rangle$. Finally, the last step is to apply the inversion around the mean.

$$|\psi_f\rangle = 2(\langle\psi|\psi\rangle - I) |\psi_3\rangle \quad (24)$$

Using the two equations (13) and (23), we get

$$|\psi_f\rangle = \frac{-\sqrt{7}}{8\sqrt{2}} |u\rangle + \frac{11}{8\sqrt{2}} |101\rangle \quad (25)$$

It is self-evident θ that is the angle formed by the two quantum states, $|\psi_f\rangle$ and $|\psi_2\rangle$. Note that the amplitude of the marked state $|101\rangle$ is greater than the other quantum states $|i\rangle$, with $i \neq 101 = 5$. Subsequently, measuring the state based on

the computation will project it into quantum state 101 with the following probability:

$$p = \left|\frac{11}{8\sqrt{2}}\right|^2 = \left|\frac{121}{128}\right| \approx 0.945 \quad (26)$$

Therefore, after two iterations of applying Grover's operator, the chance of getting the sought result, which corresponds to the state $|101\rangle$ achieves an accuracy of nearly 94.5%. In the rest of this use case, we will show how important it is to know the major impact of the number of iterations of the Grover algorithm on accuracy. Suppose the number of iterations is unknown in advance. In this case, we will perform additional Grover iterations as follows:

$$|\psi_5\rangle = \frac{-\sqrt{7}}{8\sqrt{2}}|u\rangle - \frac{11}{8\sqrt{2}}|101\rangle = \frac{-1}{4}|\psi\rangle - \frac{5}{4\sqrt{2}}|101\rangle \quad (27)$$

The stage of the inversion around the mean induces the state $|\psi_6\rangle$, which is represented.

$$|\psi_6\rangle = 2(\langle\psi|\psi\rangle - I)\,|\psi_5\rangle$$

$$= 2(\langle\psi|\psi\rangle - I)\,(\frac{-1}{4}|\psi\rangle - \frac{5}{4\sqrt{2}}|101\rangle)$$

$$= \frac{-7}{8}|\psi\rangle + \frac{5}{4\sqrt{2}}|101\rangle$$

$$= \frac{-7}{8}(\frac{\sqrt{7}}{2\sqrt{2}}|u\rangle + \frac{1}{2\sqrt{2}}|101\rangle) + \frac{5}{4\sqrt{2}}|101\rangle$$

$$= \frac{-7\sqrt{7}}{16\sqrt{2}}|u\rangle + \frac{13}{16\sqrt{2}}|101\rangle \quad (28)$$

The measurement of the state $|\psi 6\rangle$ turns out to be us.

$$p = \left|\frac{13}{16\sqrt{2}}\right|^2 = \left|\frac{169}{512}\right| \approx 0.336 \quad (29)$$

Now, if we perform a measurement on the other states, the corresponding probability is calculated as below:

$$p = \left|-\frac{7\sqrt{7}}{16\sqrt{2}}\right|^2 = \left|\frac{343}{512}\right| \approx 0.67 \quad (30)$$

Table I shows the performance of the Grover algorithm according to the number of iterations. We notice that the probability of finding a solution for a search space of a specified size varies according to the number of iterations.

TABLE I. PERFORMANCE MEASUREMENT OF THE DIFFERENT ITERATIONS OF GROVER'S ALGORITHM

Simulator	*No. of Grover Iterations*	*Accuracy*
ibm_qasm_simulator	1	0.78
	2	0.945
	3	0.67

Therefore, if we continue the number of Grover iterations after the optimal number of *round*$(\frac{\pi}{4}\sqrt{N})$, the probability of finding the sought state decreases while the probability of error increases more and more. In the event of exceeding the number of iterations, which in our instance is two, the accuracy decreases by a percentage of 0.275. Thus, we report the empirical implementation of Grover's quantum search algorithm on the IBM quantum simulator with three qubits. Fig. 7 illustrates well the theoretical results that we have carried out. The QISKit code for the implementation can be found on my GitHub under the link: https://github.com/cherradii/Grover_Quantum_Search_Alg.

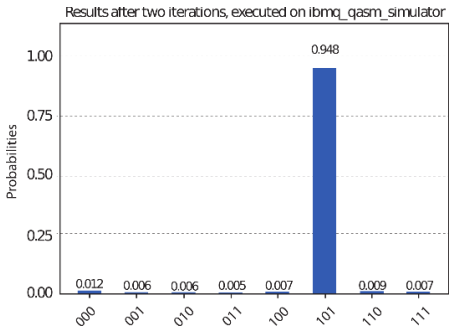

Fig. 7. Searching for Quantum State $|101\rangle$.

B. Iteration of Grover's Algorithm

Grover's algorithm is made up of a quantum subroutine named Grover's iteration, noted *G*, which is broken down into two steps:

- Apply the oracle U_f
- Apply the diffusion operator *G* on the first *n* qubits.

The iterations of Grover's algorithm are seen from a geometric point of view as a rotation in the two-dimensional space wrapped by the two vectors $|\alpha\rangle$ and $|\beta\rangle$. $|\alpha\rangle$ denotes normalized states of the sum of all targets, and $|\beta\rangle$ denotes normalized states of the sum of non-targets. The initial state $|S\rangle$ can be written as follows:

$$|S\rangle = sin\,(\theta)|\alpha\rangle + cos\,(\theta)|\beta\rangle \quad (31)$$

When looking in a search space of $N = 2^n$ items, there are *M* targets for searching ($0 \le M \le N$). Since $sin\,(\theta) = \sqrt{\frac{M}{N}}$, Apply Grover's operator (*G*) to states $|S\rangle$ for *k* times.

$$G^k|S\rangle = sin\,((2k+1)\theta)|\alpha\rangle + cos\,((2k+1)\theta)|\beta\rangle \quad (32)$$

When this appears, the target state will be explored with the probability of success *P*, formulated as follows:

$$p = sin^2((2k+1)\theta)) \quad (33)$$

Set $k = \frac{\pi}{4}\sqrt{MN}$, The Fig. 8 corresponds to the probability of success according to the proportion of target states in Grover's algorithm. To make things easier, let us set the proportion of the target as $\gamma = M/N$.

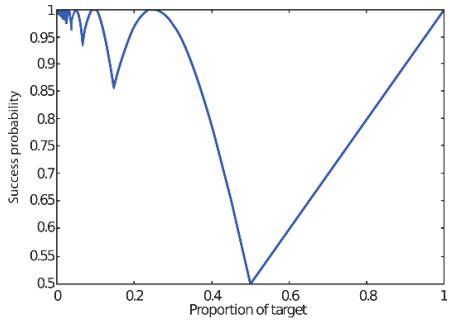

Fig. 8. The Success Probability of Grover's Algorithm.

To make things easier, let us set the proportion of the target as γ = M/N. Analyzing Fig. 8, we notice that the minimum probability that Grover's algorithm can reach is about 50% when γ = 0.5. Therefore, when 1/4 ≤ γ ≤ 1/2 the success probability of the proportion target declines rapidly. In return, when γ ≥ 1/2 the success probability of the proportion target gradually increases until it reaches 100% full accuracy when γ = 1.

C. Comparison of Grover's with Classical Algorithms

The practical implementation of Grover's search algorithm proved the efficiency in terms of its accuracy. After analyzing the different iterations, we found that the algorithm's effectiveness is influenced by the number of iterations. Moreover, applying the Grover algorithm iterations for a total number of *round* ($\frac{\pi}{4}\sqrt{N}$) times is the best choice to maximize the success probability of Grover's quantum search algorithm. Further, the quadratic reduction complexity of the quantum search Grover algorithm presents a major advantage over classical algorithms and exceeds any known classical algorithm of sub-exponential complexity. As shown in Fig. 9, Grover's quantum algorithm complexity time and classical counterpart algorithms.

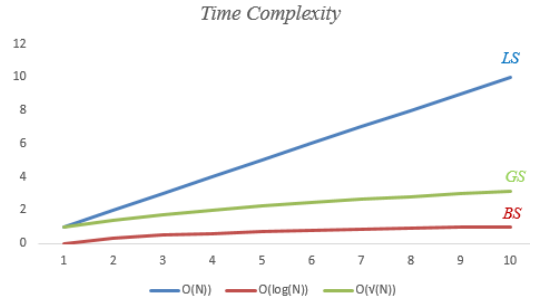

Fig. 9. The Average Number of Steps Needed to Find a Solution.

Therefore, a benchmark examination comparing the conventional search algorithms like sequential and interval search with the quantum Grover algorithm is still required. Table II shows a benchmark study between classical search algorithms and their counterparts, Grover's quantum search.

TABLE II. COMPARISON BETWEEN THREE DIFFERENT SEARCH ALGORITHMS

	Binary Search	***Linear Search***	***Grover Search***
Time complexity	*O(log(N))*	*O(N)*	*O(√(N))*
Database requirements	Database must be sorted	No requirements	No requirements
Algorithm type	Divide and conquer	Iterative	Iterative and parallel
Implementation	Medium	Easy	Hard

According to the comparison in Fig. 9, the binary search is the most sophisticated search, but it requires that the data be sorted, which is no longer possible with unstructured data. Linear search can override binary search if the targeted element exists at the beginning. However, being a search request for one or more elements in the heterogeneous database that contains data in different formats (structured, semi-structured, and unstructured), like the case of a data lake, Grover's algorithm remains the most efficient compared to the classical searching algorithms. Consequently, quantum algorithms are more prominent and highly recommended thanks to their quadratic acceleration, which is very fast compared to exponential acceleration, which corresponds to classical algorithms.

V. CONCLUSION

In this paper, an interesting algorithm is used to solve the search problem for unstructured datasets. We have investigated a clear procedure for making use of the potential of the quantum search Grover algorithm by proposing the design and implementation of the algorithm, including the prevalence effect of the number of iterations to decrease data processing time in unsorted databases. Based on this solution, our experimental results are very encouraging, and demonstrate the usefulness of Grover's algorithm to be applied efficiently to solve the search problem with high accuracy. Thus, from the benchmark search algorithms discussed in Section IV.C, we have retained that Grover's algorithm appears the best solution to the search problem in an unstructured data space. An important future perspective consists of moving to a higher dimension to solve the larger space search challenge with a large number of qubits.

REFERENCES

[1] Dabbèchi H, Nahla Z, Haytham E, Kais H. NoSQL Data Lake: A Big Data Source from Social Media. In: International Conference on Hybrid Intelligent Systems, pp. 93-102. Hybrid Intelligent Systems (2020).

[2] Oussous A, Benjelloun F, Lahcen A, Belfkih S. NoSQL databases for big data. In: International Journal of Big Data Intelligence. A (2017). https://doi.org/10.1504/IJBDI.2017.085537.

[3] Dabbèchi H, Nahla Z, Haytham E, Kais H. Social Media Data Integration: From Data Lake to NoSQL Data Warehouse. In: International Conference on Intelligent Systems Design and Applications. A (2021). https://doi.org/10.1007/978-3-030-71187-0.64.

[4] Ashley M. Quantum algorithms: An overview. In: npj Quantum Information. A (2016). https://doi.org/10.1038/npjqi.2015.23.

[5] Qiskit. https://qiskit.org/. Accessed 01 December (2021).

[6] Huai-Chun C, Hsiu-Chuan H. Digital quantum simulation of dynamical topological invariants on near-term quantum computers. In: Journal of Quantum Information Processing. A (2022).

[7] Aimeur E, Gilles B, Sèbastien G. Machine Learning in a Quantum World. In: Conference of the Canadian Society for Computational Studies of Intelligence, pp. 431-442. (2006).

[8] Songfeng L, Braunstein L. Quantum decision tree classifier. Quantum Information Processing. A (2013). https://doi.org/10.1007/s11128-013-0687-5.

[9] Ouedrhiri O, Banouar O, El hadaj S, Raghay S. Intelligent recommender system based on quantum clustering and matrix completion. In: Concurrency and Computation Practice and Experience; 2022.

[10] Dixon, J (CTO of Pentaho). Hadoop, and Data Lakes. In: Dixons Blogs. https://jamesdixon.wordpress.com/2010/10/14/pentaho-hadoop-and-data-lakes/. Accessed 21 Sep 2021.

[11] Sawadogo P, Darmont J. On data lake architectures and metadata management. In: Journal of Intelligent Information Systems. A (2021).

[12] Inmon B. Data Lake Architecture: Designing the Data Lake and Avoiding the Garbage Dump. Bill Inmon - Google Livres. (2016).

[13] Khine P, Wang Z. Data lake: a new ideology in big data era. In: ITM Web of Conferences. A (2018).

[14] Cherradi M, El Haddadi A, Routaib H. Data Lake Management Based on DLDS Approach. In: Networking, Intelligent Systems and Security, pp. 679-690. (2022).

[15] Hellerstein et al. Ground: A Data Context Service. CIDR (2017).

[16] Hegazi O M, Saini K D, Zia K. Moving from Heterogeneous Data Sources to Big Data: Interoperability and Integration Issues. In: International Journal of Advanced Computer Science and Applications(IJACSA), Volume 9 Issue 10, 2018.

[17] Cherradi M, EL HADDADI A. Data Lakes: A Survey Paper. In book: Innovations in Smart Cities Applications Volume 5. January (2022).

[18] Mavroeidis V, Vishi K, Zych D M, Jøsang A. The Impact of Quantum Computing on Present Cryptography. In: International Journal of Advanced Computer Science and Applications(IJACSA), Volume 9 Issue 3, 2018.

[19] Anton F. Quantum optics with artificial atoms: Thesis for: PhD. (2014).

[20] Brian R, Classical emulation of a quantum computer. In: International Journal of Quantum Information. Vol. 14, No. 04, 1640004 (2016).

[21] Lov K. A fast quantum mechanical algorithm for database search. Computer Science, Physics. A (1996). https://doi.org/10.1145/237814.237866.

[22] Peter W. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. Quantum Physics. A (1996). https://doi.org/10.1137/S0097539795293172.

[23] Shor P. Algorithms for quantum computation: discrete logarithms and factoring. In: Proceedings 35th Annual Symposium on Foundations of Computer Science. (1994).

[24] Mandviwalla A, Keita O, Bo J. Implementing Grover's Algorithm on the IBM Quantum Computers. In: IEEE International Conference on Big Data, pp. 701-710. Springer (2018).

[25] Kyongyob M. Dirac Bra-ket Notation for Interpreting Regional Distribution of Pulmonary Ventilation-Perfusion. International Journal of Teaching and Case Studies. (2015).

[26] Tin T. The Efficiency of the Quantum Search : How effective is Grover's algorithm (quantum search) opposed to classical computer searching algorithms in terms of time complexity. A (2020). https://doi.org/10.13140/RG.2.2.18744.57604.

[27] Zubairy M. Quantum Superposition and Entanglement. Quantum Mechanics for Beginners. A (2020). https://doi.org/10.1093/oso/9780198854227.003.0010.

[28] Ryan L. Overview and Comparison of Gate Level Quantum Software Platforms. Computer Science, Mathematics, Physics. (2018).

[29] Ian M. IBM ups the stakes for Quantum Computing. https://www.enterprisetimes.co.uk/2017/11/13/ibmups-stakes-quantum-computing/. Accessed 07 October 2021.

[30] IBM, Corporation: IBM Quantum Experience. https://quantum-computing.ibm.com/. Accessed 12 August 2021.

[31] IBM, Corporation.: IBM Quantum Documentation. https://quantum-computing.ibm.com/docs/. Accessed 3 September. 2021.

[32] Marco M, Enrico P.A continuous rosenblatt quantum perceptron . In: International Journal of Quantum InformationVol. 19, No. 04, 2140002 (2021).

[33] Komal S. An Indexed Sequential Search and its Comparative Analysis with basic Searching Techniques. IJEAST. A (2020). https://www.ijeast.com/papers/559-564,Tesma504,IJEAST.pdf.

[34] Kostakis O, Gionis A.Subsequence Search in Event-Interval Sequences. ACM. SIGIR. A (2015). https://10.1145/2766462.2767778.

[35] Arkadiusz L. Rafa l R. Quantum Digital Signatures for Unconditional Safe Authenticity Protection of Medical Documentation. HIGHER SCHOOL'S PULSE. A (2015). https://doi.org/10.5604/2081-2021.1191752.

[36] You Q, Franco N. Superconducting Circuits and Quantum Information. Quantum Physics. A (2005). https://10.1063/1.2155757.

[37] Arkadiusz L, Laurentiu N. The Research of Grover's Quantum Search Algorithm with Use of Quantum Circuits QX2 and QX4: Part I. In: Information Systems Architecture and Technology: Proceedings of 39th International Conference on Information Systems Architecture and Technology – ISAT, pp 146-155. ISAT (2019).

[38] Panjin K, Daewan H, Kyung C. Time–space complexity of quantum search algorithms in symmetric cryptanalysis: applying to AES and SHA-2. In: Journal of Quantum Information Processing. A (2018).

[39] Gernot S. Quantum algorithm for optical template recognition with noise filtering. Physical review A, Atomic, molecular, and optical physics. A (2006). https://doi.org/10.1103/PHYSREVA.74.012303.

[40] Ulyanov S, et al. Modelling of Grover's quantum search algorithms: implementations of Simple quantum simulators on classical computers. In: Computer Science Journal. A (2020).

[41] Riccardo F. Quantum Amplitude Amplification Algorithm: An Explanation of Availability Bias. In: Proceedings of the 3rd International Symposium on Quantum Interaction, pp. 84-96. (2009).

[42] Akanksha S, Arko C. Grover's Algorithm. Architecture Design and Implementation of the Quantum search algorithm. A (2018). https://doi.org/10.13140/RG.2.2.30860.95366.