Optimized Methods for Inserting and Deleting Records and Data Retrieving in Quantum Database

Amor Gueddana*, Rihab Chatta and Noureddine Boudriga
Engineering School of Communication of Tunis (SUP'COM), Ghazala Technopark, 2083, Ariana, Tunisia
**e-mail: Amor.gueddana@gmail.com*

ABSTRACT
In this paper, we show how a quantum CNOT-based relational database is built, then we show how to query its quantum tables using the most used SQL-like queries, e.g. INSERT, UPDATE, DELETE and SELECT. We specify each time the evolution of the probability amplitude corresponding to the records before and after the query has been executed and we propose corresponding circuits implementation.
Keywords: Quantum Gates, Grover's algorithm, Quantum Database, Quantum Query.

1. INTRODUCTION

Quantum computation represents a challenge for engineering and computer science. Manipulating data on a quantum database is at the heart of quantum computing. Inserting, updating and deleting values from superposed states were proposed by A. Younes *et al*. [6]; however, circuit implementation and evolution of the probability amplitude were neglected in his work. Deleting a marked item from a database with a single query was better treated by Y. Liu *et al*. [4] since they provide useful information about the amplitude of probability but no circuit implementation provided to handle table management. Searching items with respect to specific oracle were proposed by I.Y.L. Ju *et al*. [7], they showed a circuit based implementation of the Grover's algorithm used for the database search process; but data basic operations like INSERT and DELETE have not been addressed.

In this paper, we show how a quantum database can be built from a relational multi-tables database and we show how querying operations can be handled through quantum circuits. Our contribution is three-fold: first, we show how a set of quantum circuits can be associated to a given relational database and can handle queries; second, we provide a circuit based implementation of generic SQL-like queries; third, we define a computation scheme for the determination of the amplitude of probability after every query implementation.

This paper is organized as follows: after a brief presentation of quantum basics in Section 2, Section 3 presents the steps necessary for constructing any quantum table of a quantum database, Section 4 treats in details the circuits necessary to realize different kinds of queries on those tables, the conclusion is presented in Section 5.

2. BASICS ON QUANTUM COMPUTING

In the following, we consider five major functions useful for building querying and managing quantum circuits associated to database [1][2]. They are: the identity (I), the negation (X), the phase shift of $\alpha\pi$ over the state $|1\rangle$ (T_α), the phase shift of $\beta\pi$ over all basis states (P_β), and the Hadamard transform (H). The matrix representations of these functions involving one-qubit gates transform, for $\{\alpha,\beta\} \in \mathbb{R}$, are given as follows:

$$\begin{gathered} I = |0\rangle\langle 0| + |1\rangle\langle 1| = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}; \quad X = |1\rangle\langle 0| + |0\rangle\langle 1| = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}; \quad T_\alpha = |0\rangle\langle 0| + e^{i\alpha\pi}|1\rangle\langle 1| = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\alpha\pi} \end{pmatrix} \\ P_\beta = e^{i\beta\pi}(|0\rangle\langle 0| + |1\rangle\langle 1|) = \begin{pmatrix} e^{i\beta\pi} & 0 \\ 0 & e^{i\beta\pi} \end{pmatrix}; \quad H = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\langle 0| + \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)\langle 1| = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \end{gathered} \tag{1}$$

The Grover's algorithm [3] [4] that allows searching or retrieving items from a table can be generalized to the following 5-step process, where $\phi = \eta\pi$, $\eta \in \mathbb{R}$:

Step 1 perform a conditional phase shift $e^{i\phi}$ to all states verifying an oracle $\{|i\rangle \mid i \in A\}$, we denote this action as I_c and it is described as follows:

$$I_c = I + \left(e^{i\phi} - 1\right)\sum_{i\in A}|i\rangle\langle i| \tag{2}$$

Step 2 realize the Hadamard transform to the n qubits (or applying $H^{\otimes n}$) constituting the superposition.

Step 3 perform a conditional phase shift $e^{i\phi}$ to the $|0\rangle$ state, only. This action is denoted by I_0.

Step 4 perform $W_n = H^{\otimes n}$ once again.

Step 5 realize a phase shift of $e^{i\pi}$ to the resulting state.

The Grover operator G is defined by $G = e^{i\pi} W_n I_0 W_n I_c$. It is worth to notice that G is a retrieval, when $\eta = 1$ [3][7], and a deletion when $\eta < 1$. In particular, when $\eta = 1/3$, the deletion is optimal since it is made with certainty for high data size [4].

Finally, based on the S_j operator presented in [6] for small values of j, we define an insertion operator denoted by INS_g, permitting to insert S states ($S \in \mathbb{N}$) writable over the same number of qubits n, where $g = \{|j_s\rangle \mid s = 1,2,\ldots,S\}$ is a group of entries to be inserted simultaneously in a quantum table. To this end, for every entry $|j_s\rangle$ to be inserted, let $k_s = \lfloor \log_2(j_s) \rfloor$, $n = n_s = k_s + 1$ and $l_s = j_s - 2^{k_s}$ represented over n qubits, the general operator INS_g is given by:

$$INS_g = \left(H \otimes \sum_{s=1}^{S} |l_s\rangle\langle l_s| \right) + \left(I \otimes \sum_{i=0; i \neq \{l_s\}}^{2^n - 1} |i\rangle\langle i| \right) = \prod_{s=1}^{S} INS_{|j_s\rangle} \tag{3}$$

INS_g allows the insertion of the group of entries in g simultaneously to the existing superposition of states leaving fixed all of the states $\{|i\rangle \mid i \in \{0,1,\ldots,2^n-1\}; i \neq l_s\}$ and superposing it using a Hadamard gate controlled by the states $\{l_s \mid s = 1,2,\ldots,S\}$. Inserting the elements of g one by one can be achieved by successive applications of $INS_{|j_s\rangle}$ operators but $S-1$ additional Hadamard gates should be used. It is obvious to notice that $INS_g \neq INS_{j_1}.INS_{j_2}\ldots INS_{j_s}$.

3. QUANTUM DATABASE DESIGN

Let DB be a relational database. Roughly speaking, DB is a finite set of relations (tables). A quantum database associated to DB may be viewed as a set of quantum circuits, each one is associated to a table. Getting a CNOT-based circuit from a table is obtained using the transforms described in [5]. Informally speaking, these transforms are obtained by defining a classical truth table from a related Boolean function and then transforming it to a quantum computing version of the truth table. For a given table T composed of t attributes, where each attribute R_i $(i = 1,2,\ldots,t)$ contains N_i different values, the number of qubits necessary for the circuit realization for each attribute, denoted by n_i, is $\log_2(N_i)$ if N_i is a power of 2, else it is $\lfloor \log_2(N_i) \rfloor + 1$. After circuit realization of T, applying Hadamard gates to the n_k qubits, constituting its primary key, results in equally distributed amplitude of probability of $1/\sqrt{2^{n_k}}$ to each state contained in T including NULL values.

For the sake of clarity, let DB be the database reduced to Table 1. Table 1 links department name (DName) and department number (DNum). Figure 1 depicts the circuit built from the Table 1 (the number of entries is 8 and two of them are unused). The circuit of Fig. 1 contains 16 negation gates (represented by $\oplus$ in stage 2), three Hadamard gates (Drawn as $\boxed{\mathrm{H}}$ in stage 1) and 22 full dots representing the control entry qubits set to $|1\rangle$.

Table 1. Digitization of the table Department.

DNum	A	DName	B
35	0	Sales	7
46	1	Engineering	4
49	2	NULL	0
27	3	Clerical	2
29	4	Finance	5
42	5	Service	1

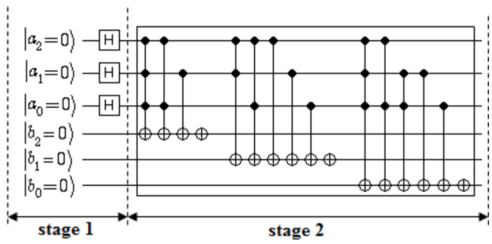

Figure 1. Quantum circuit for the table Department.

Table 1 is unsorted, field B represents a random order of field DName. Field A represents the unique ordering of the primary keys (DNum), assuming that NULL are ordered first and encoded by the state $|0\rangle$.

The Boolean functions associated to table department and used to construct the circuit of Fig. 1, are given by:

$$b_2 = \overline{a_1}\left(\overline{a_2} + \overline{a_0}\right); b_1 = \overline{a_2}\left(\overline{a_1 \oplus a_0}\right); b_0 = \overline{a_1}\left(a_2 + \overline{a_0}\right) \tag{4}$$

Where a_2, a_1, a_0 and b_2, b_1, b_0 are the qubits necessary to represent A and B, respectively. $\overline{a_i}$ and a_i represents the qubits a_i in states $|0\rangle$ and $|1\rangle$, respectively.

The probability associated to Table 1 is given by:

$$|Dep\rangle = \frac{1}{\sqrt{8}} \sum_{i=0}^{7} |A_i, B_i\rangle \tag{5}$$

4. QUERYING QUANTUM DATABASES

4.1 Inserting Records

Inserting records into a quantum database can be reduced to the insertion of different entries to some of the quantum tables constituting the quantum database. We consider a quantum table containing N superposed records:

$$|T\rangle = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} |k,i\rangle |q\rangle , \tag{6}$$

where k refers to the key value of the record, i represents the rest of the columns values and $|q\rangle$ is an auxiliary qubit set initially to $|0\rangle$. Inserting a record $|x, j\rangle$ into the quantum circuit is done through three cases, depending on the number n of qubits available in the circuit to represent the key values, the key value $|x\rangle$ and the number N of states existing in the superposition. These cases are given as follows:

(1) If $N < 2^n$ and $x < 2^n$, two sub-cases appear: (a) the value $|x\rangle$ exists in the superposition but is unused, meaning that the superposition contains the state $|x\rangle$ entangled with $|q\rangle = |1\rangle$, in this case, the inserting process becomes a matter of updating $|x,i\rangle$ to $|x, j\rangle$ by applying an ad hoc update operator to $|i\rangle$, that we denote by $u_{i\to j}$. The realization of $u_{i\to j}$ is based on a transformation of the current circuit of the table by replacing qubits states representing $|i\rangle$, using NOT gates controlled by qubits representing the selected unused key value $|x\rangle$, until obtaining the desired state $|j\rangle$; (b) the value $|x\rangle$ doesn't exist in the superposition, which means that all key values in the current superposition are used. Here, we begin by creating sufficiently enough states in the superposition that are unused in order to reduce the search complexity with future insertion and then insert $|x\rangle$ the way it was inserted in (a). Inserting simultaneously a set of key values, $g = \{|x_s\rangle \mid s = 1,2,...,S\}$, where S is a positive integer, is obtained by applying the INS_g operator as described by equation (3), the state of $|T\rangle$ as described by equation (6) becomes:

$$|T\rangle = INS_g\,|T\rangle = \frac{1}{\sqrt{2N}} \sum_{s=1}^{S} |l_s,i\rangle|1\rangle + \frac{1}{\sqrt{N}} \sum_{k=0;k\neq\{l_s,x_s|s=1,2,...,S\}}^{N-1} |k,i\rangle|0\rangle + \frac{1}{\sqrt{2N}} \sum_{s=1}^{S} |x_s,i\rangle|1\rangle . \tag{7}$$

As described by equation (7), both $\{|l_s,i\rangle\}$ and $\{|x_s,i\rangle\}$ are entangled with $|q\rangle = |1\rangle$. To keep this latter entangled only with states supporting insertion $(\{|x_s,i\rangle\})$, we apply the same set of gates that has been used to create entanglement of $\{|l_s,i\rangle\}$ with $|q\rangle = |1\rangle$, the state of $|T\rangle$ as described by equation (7) becomes:

$$|T\rangle = \frac{1}{\sqrt{2N}} \sum_{s=1}^{S} |l_s,i\rangle|0\rangle + \frac{1}{\sqrt{N}} \sum_{k=0;k\neq\{l_s,x_s|s=1,2,...,S\}}^{N-1} |k,i\rangle|0\rangle + \frac{1}{\sqrt{2N}} \sum_{s=1}^{S} |x_s,i\rangle|1\rangle . \tag{8}$$

After insertion of $|x\rangle$, a reset of the qubit $|q\rangle$ to the state $|0\rangle$ is required. To achieve this objective, we apply a NOT gate to the qubit controlled by the inserted key value $|x\rangle$.

(2) If $x \geq 2^n$, a new qubit set initially to $|0\rangle$ is added to the circuit and then inserting $|x\rangle$ is done the way it was inserted in (1-b) for $n' = n + 1$.

For a quantum table containing N records, the number of extra unused key values permitted to be present in the superposition, which we have denoted by S, is estimated by the following lemma.

Lemma. The maximum number of unused states S, able to be present in the superposition with N used records represented over n qubits and permitting to keep unchanged near future search operation complexity is:

$$S = \min(N_+, 2^n - N); \text{ where } N_+ = \left({}^4\!/_\pi \left(\left\lfloor \left(\frac{\pi}{4}\sqrt{N} \right) \right\rfloor + 1 \right) \right)^2 - N . \tag{9}$$

Proof. The complexity of a search operation is associated to the number of iterations realized to retrieve a unique item from a table. Practically, this number is $M = \left\lfloor {}^\pi\!/_4 \sqrt{N} \right\rfloor$ for a table containing N superposed records represented over n qubits. Adding S unused key values to the superposition should neither exceed 2^n values nor alter M, say $M = {}^\pi\!/_4 \sqrt{N + N_+}$. For that purpose, M should remain equal to $\left\lfloor {}^\pi\!/_4 \sqrt{N} \right\rfloor + 1$ which is equivalent to say ${}^\pi\!/_4 \sqrt{N + N_+} = \left\lfloor {}^\pi\!/_4 \sqrt{N} \right\rfloor + 1$. This means that $\sqrt{N + N_+} = {}^4\!/_\pi \left(\left\lfloor {}^\pi\!/_4 \sqrt{N} \right\rfloor + 1 \right)$, then $N_+ = \left({}^4\!/_\pi \left(\left\lfloor \left(\frac{\pi}{4}\sqrt{N} \right) \right\rfloor + 1 \right) \right)^2 - N$. Finally, we choose $S = \min(N_+, 2^n - N)$.

Let us notice that near future in the lemma means "the next S^{th} search operations ".

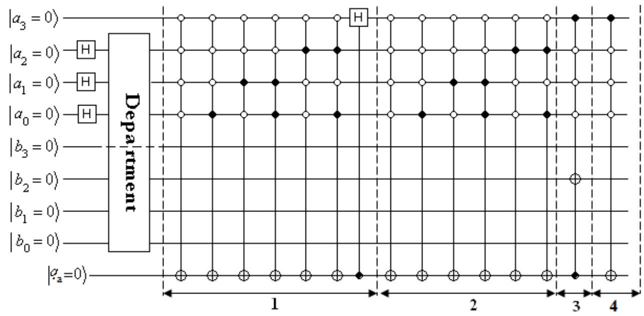

Figure 2. Inserting $|A, j\rangle = |8, 3\rangle$ to the table $|Dep\rangle$.

As an example, suppose we want to insert the state $|A, j\rangle = |8, 3\rangle$ to the table $|Dep\rangle$ where we suppose also that all key are used. We compute $S = \min(6,8) = 6$ and we begin by applying INS_g for $g = \{|x\rangle \mid x = 8, 9, \ldots, 13\}$. The Hadamard gate used is controlled by the states $\{|l\rangle \mid l = 0, 1, \ldots, 5\}$ that are entangled with $|q_a\rangle = |1\rangle$ (stage 1 of Fig. 2), we apply then the same set of gates (stage 2 of Fig. 2). Inserting $|8, 3\rangle$ is reduced to apply $u_{7\to 3}$ (stage 3 of Fig. 2), $|q_a\rangle$ is then reset to the state $|0\rangle$ for the key value $|A\rangle = |8\rangle$ (stage 4 of Fig. 2).

4.2 Deleting Records from a Quantum Table

For the table described by equation (6), suppose that $|d, i\rangle$ is the state to be deleted from the table:

$$|T\rangle = \frac{1}{\sqrt{N}} \sum_{k=0;k\neq d}^{N-1} |k, i\rangle|0\rangle + \frac{1}{\sqrt{N}} |d, i\rangle|0\rangle \qquad (10)$$

Since the state $|d, i\rangle$ is entangled with $|q\rangle = |0\rangle$, the deletion process consists of flipping the state of $|q\rangle$ to $|1\rangle$. This is done by applying a NOT gate to qubit $|q\rangle$ controlled by the value $|d\rangle$. When reaching $2S$ unused key values in the superposition, the state of table $|T\rangle$ as described by equation (10) becomes:

$$|T\rangle = \frac{1}{\sqrt{N}} \sum_{k=0;k\neq\{d_s|s=1,2,\ldots,2S\}}^{N-1} |k, i\rangle|0\rangle + \frac{1}{\sqrt{N}} \sum_{s=1}^{2S} |d_s, i\rangle|1\rangle \qquad (11)$$

To reduce the complexity of future searches, we need to limit the number of unused states. Thus, S states has to be totally removed from the superposition in order to keep only S unused states entangled with $|q\rangle = |1\rangle$, as stated by the lemma of the previous section. This is done by applying a DEL_g operator, where $g = \{|d_s\rangle \mid s = [1, 2, \ldots, S]\}$ are states to be deleted corresponding to the odd values of the $2S$ unused key.

It is worth to notice that DEL_g is the G operator corresponding to the generalized Grover's algorithm when applied with parameter $\phi = \pi/3$ and an oracle evaluating to true the states not to be deleted and to false the states given by g, however, only states having the least significant qubit value set to $|1\rangle$ and entangled with $|1\rangle$ should evaluate the oracle to false. The table given by equation (11), after applying one time DEL_g becomes:

$$|T\rangle = \frac{e^{-i\pi/3}}{\sqrt{N-S}} \left(\sum_{k=0;k\neq\{d_s|s=1,2,\ldots,S\}}^{N-S-1} |k, i\rangle|0\rangle + \sum_{s=1}^{S} |d_s, i\rangle|1\rangle \right) \qquad (12)$$

As shown by equation (12), except a global phase factor of $e^{-i\frac{\pi}{3}}$, the group g of unused key values has been deleted, this phase factor is eliminated by a phase rotation of $e^{i\frac{\pi}{3}}$ to all basis states realized by a $P_{1/3}$ gate (gate P_β of equation 1 for $\beta = 1/3$).

4.3 Retrieving Records from Joined Quantum Tables

Retrieving records from a single table is achieved by applying the generalized Grover's algorithm with parameters $\phi = \pi$ [3][7]. In this section, we focus our interest on record selection from natural joined quantum tables. Other types of join operation can be addressed in a similar way.

We consider two tables T_A and T_B, realized by two quantum circuits C_A and C_B, containing N_A and N_B equally superposed states described as $\sum_{N_A} \left| k_A^p, i, k_A^f \right\rangle$ and $\sum_{N_B} \left| k_B^p, j \right\rangle$,respectively , where k_A^p, k_A^f and k_B^p stands for primary and foreign key of T_A and primary key of T_B, represented over n_A^p, n_A^f and n_B^p qubits, respectively. The quantum circuit corresponding to the natural join between C_A and C_B is obtained by connecting the n_A^f qubits at the output of C_A to the input n_B^p qubits of C_B. Superposition of the resulting joined states is obtained by applying a Hadamard gate to each qubit constituting the n_A^p qubits. The resulting joined circuit contains N_A equally superposed states described as $|J\rangle_{T_A,T_B}^{Natural} = \sum_{N_A} |k_A^p, i, k_A^f, j\rangle$.

For the sake of clarity and due to space constraints, here after an example of natural join. Let the table "Employee" be as described by Table 2, the corresponding quantum CNOT-based circuit is realized the way it was done for Table1. The quantum circuit realization of the natural join between Table 1 and 2 is depicted by stage 2 of Fig. 3. Applying Hadamard to c_1 and c_0 (stage 1 of Fig. 3) gives the employees with the attached departments. However, the states composing the resulting superposition are contained in Table 3.

Table 2. Digitization of the table Employee.

ENum	C	EName	D	DNum	E
123	0	Albert	3	27	3
119	1	Brian	0	46	1
128	2	Craig	1	42	5
120	3	Oliver	2	27	3

Table *3. Result of natural join between Table 1 and Table 2.*

ENum	C	EName	D	DNum	E	DName	B
123	0	Albert	3	27	3	Clerical	2
119	1	Brian	0	46	1	Engineering	4
128	2	Craig	1	42	5	Service	1
120	3	Oliver	2	27	3	Clerical	2

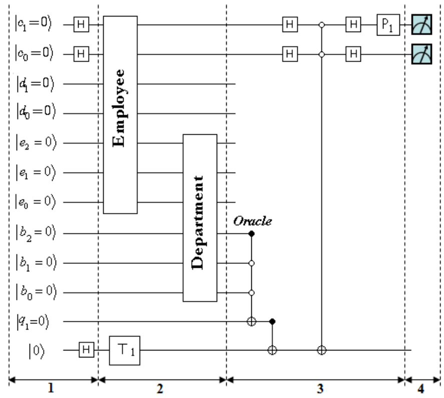

Figure 3. Retrieve of ENum=119 $(|C_1\rangle = |1\rangle)$.

Searching the number of the employee belonging to the department whose name is "Engineering" is realized by the quantum circuit given by stage 3 of Fig. 3. After $M = \pi/4\sqrt{4} \simeq 2$ repetitions of stage 3 (Fig. 3), the state of the joined tables becomes:

$$|J\rangle_{Emp,Dep}^{Natural} = \frac{1}{2\sqrt{3}} \sum_{i=0;i\neq 1}^{3} |C_i, D_i, E_i, B_i\rangle + \sqrt{\frac{3}{4}} |1,\ 0,\ 1, 4\rangle \tag{13}$$

A final measurement (stage 4 of Fig. 3) is necessary to observe the result of the query, the measurement implies a collapse of the superposed states of equation (13) to one of the basis states, with higher probability to observe the searched state. This measure destructs the superposition and a reconstruction of the database is necessary.

5. CONCLUSIONS

We have defined a scheme that takes a relational database and transform it in a quantum DB allowing an optimized query processing of SQL-based queries. Our approach has been applied to a simple example to show how our scheme performs.

A future work will address the realization of more complex queries and build an appropriate environment for quantum transaction execution.

REFERENCES

[1] T.C. Ralph, N.K. Langford and A.G. White: Linear optical controlled not gate in the coincidence basis, *Phys Rev A*, vol. 65, pp. 24-28, 2002

[2] M. Nakahara and T. Ohmi: *Quantum Computing from Linear Algebra to Physical Realizations*, L. Taylor & Francis Group, 2008.

[3] L.K. Grover: A fast quantum mechanical algorithm for database search, in *Proc. 28th Ann. ACM Symp. Theor. Comput.*, pp. 212-219, 1996.

[4] Y. Liu and G.L.Long: Deleting a marked item from an unsorted database with a single query, *ArXiv Quantum Physics e-Prints*, Cornell Univ., Ithaca, NY, Oct. 2007

[5] A. Younes and J.Miller: An automated method for building CNOT based quantum circuits for Boolean function, *ArXiv Quantum Physics e-Prints*, Cornell Univ, Ithaca, NY, Apr. 2003

[6] A. Younes: Database manipulation on quantum computers, *ArXiv Quantum Physics e-Prints*, Cornell Univ., Ithaca, NY, May. 2007

[7] I.Y.L. Ju and S.Y. Kuo: Quantum circuit design and analysis for database search application, *IEEE Transaction on Circuits and Systems-I: Regular Papers*, vol. 54, no. 11, pp. 2552-2562, Nov. 2007.