Concise quantum associative memories with nonlinear search algorithm

J.-P. Tchapet Njafa* and S.G. Nana Engo

Received 11 August 2015, revised 14 November 2015, accepted 3 December 2015
Published online 11 January 2016

The model of Quantum Associative Memories (QAM) we propose here consists in simplifying and generalizing that of Rigui Zhou et al. [1] which uses the quantum matrix with the binary decision diagram put forth by David Rosenbaum [2] and the Abrams and Lloyd’s nonlinear search algorithm [3]. Our model gives the possibility to retrieve one of the sought states in multi-values retrieving scheme when a measurement is done on the first register in $O(c - r)$ time complexity. It is better than Grover’s algorithm and its modified form which need $O(\sqrt{\frac{m}{n}})$ steps when they are used as the retrieval algorithm.

The NonLinear Search Algorithm (NLSA) is based on the fact that it has been suggested that under some circumstances, the superposition principle of quantum theory might be violated. In other words, sometimes a quantum system might have temporal nonlinear evolution. Therefore, nonlinear quantum computer could solve NP-complete and even #P problems in polynomial time that Abrams and Lloyd argued in 1998 in their nowadays classic paper [3]. We recall that the NLSA of Abrams and Lloyd uses the Weinberg’s prescription and they based their argument on a general property of nonlinear evolutions in Hilbert spaces. This nonlinear evolution is the non-conservation of scalar products between nonlinearly evolving solutions of a nonlinear Schrödinger equation. We call this effect a mobility phenomenon.

With the intention of avoiding the fact that the Weinberg’s formalism implies a transmission faster than the light [8], Czachor [7] proposed another description based on the Polchinski-type one. In the present paper, we follow the Czachor’s description. Meyer and Wong give in [9] another reason which can justify the use of nonlinear formalism in quantum mechanics:

“[...] An obvious question is whether a modest, physically motivated nonlinearity can still produce a computational advantage. In particular, consider Bose-Einstein condensates (BECs). [...] In general, describing such many-body systems is difficult because of the many interaction terms. But under certain conditions, one can assume that only two-body contact interactions contribute and the s-wave scattering length

1 Introduction

Quantum Neural Networks are Artificial Neural Networks functioning according to quantum laws. One of the useful Neural Networks is the Associative Memory which is an important tool for pattern recognition, intelligent control, and artificial intelligence. Ventura and Martínez built a model of Quantum Associative Memory (QAM) where the stored patterns were considered as the basis states of the memory quantum state [4]. They used a modified version of the well-known Grover’s quantum search algorithm in an unsorted database as the retrieval algorithm. In order to overcome the limits of that model that solves only the completion problem by retrieving data from noisy data, Ezhov et al. used an exclusive method of quantum superposition and Grover’s algorithm with distributed queries [5]. However, their model still produces non-negligible probability of irrelevant classification. We have recently put forth an improved model of QAM with distributed query that reduces the probability of this irrelevant classification [6].

The NonLinear Search Algorithm (NLSA) is based on the fact that it has been suggested that under some circumstances, the superposition principle of quantum theory might be violated. In other words, sometimes a quantum system might have temporal nonlinear evolution. Therefore, nonlinear quantum computer could solve NP-complete and even #P problems in polynomial time that Abrams and Lloyd argued in 1998 in their nowadays classic paper [3]. We recall that the NLSA of Abrams and Lloyd uses the Weinberg’s prescription and they based their argument on a general property of nonlinear evolutions in Hilbert spaces. This nonlinear evolution is the non-conservation of scalar products between nonlinearly evolving solutions of a nonlinear Schrödinger equation. We call this effect a mobility phenomenon.

With the intention of avoiding the fact that the Weinberg’s formalism implies a transmission faster than the light [8], Czachor [7] proposed another description based on the Polchinski-type one. In the present paper, we follow the Czachor’s description. Meyer and Wong give in [9] another reason which can justify the use of nonlinear formalism in quantum mechanics:

“[...] An obvious question is whether a modest, physically motivated nonlinearity can still produce a computational advantage. In particular, consider Bose-Einstein condensates (BECs). [...] In general, describing such many-body systems is difficult because of the many interaction terms. But under certain conditions, one can assume that only two-body contact interactions contribute and the s-wave scattering length

Laboratory of Photonics, Department of Physics, University of Ngaoundere, POB 454 Ngaoundere, Cameroon
* Corresponding author E-mail: jptchapet@univ-ndere.cm
a is much less than the interparticle spacing. Then using mean field theory, one finds that the system is approximately described by a nonlinear Schrödinger equation [...]

Therefore, Meyer and Wong used Gross-Pitaevskii equation to build their nonlinear search algorithm.

Rigui et al. [1] recently proposed a model of Ventura’s associative memory which uses the Binary Superposed Quantum Decision Diagram (BSQDD) as a learning process. They also used the above nonlinear algorithm of Abrams and Lloyd as a retrieving process for multi-values retrieval. Although the learning process of their model is good, there are some ambiguities on how the memory evolves and how the multi-values retrieval arises. First of all, there is no exact description about operator \( U_i \) which links the first register denoted by \( |\psi_i\rangle \) to the second register denoted by \( |\gamma\rangle \). Secondly, the use of a simple binary decision diagram to represent the states \( \Phi \) (see step 2 on section 3.2) only shows the way to attain the needed state. Moreover, the nonlinear operator denoted by \( U_2 \) is used on a particular state, not on a superposed state (see step 3 on section 3.3 and section 4.2 in [1]). There is also no indication on how a measurement will give a needed state because the nonlinear search algorithm leaves the first register in a superposed state.

In this paper, as the primary innovation, we propose a concise NLSA for QAM with a method to retrieve one of the sought states especially in multi-values retrieving scheme when we do a measurement only on the first register. The time complexity is \( \mathcal{O}(c-r) \). The parameters \( c \) and \( r \) are obtained as it follows: if \( n \) is the number of qubits of the first register, \( p \leq 2^n \) is the number of stored patterns. If the values of \( t \) qubits are known (i.e., \( t \) qubits have been measured or have already been disentangled from others, or the oracle operator acts on a subspace of \( (n-t) \) qubits), then we have the number of stored patterns \( q \leq p \) and the number of values for which \( f(x) = 1 \) is \( m \leq q \). Therefore, \( c = \text{ceil}(\log_2 q) \), it is the least integer greater or equal to \( \log_2 q \), and \( r = \text{int}(\log_2 m) \) it is the integer part of \( \log_2 m \). Thus, our model simplifies and generalizes that of Rigui et al. [1].

However, if the strength of the nonlinearity provides a large computational advantage, it also makes the system highly susceptible to the noise which appears as a true bottleneck that may limit the usefulness of the NLSA. As the secondary innovation of this paper, we investigate the effects of the noise in the algorithm by considering the bit-flip quantum channel modeling the environmentally induced noise. We assume that at most a single complete bit-flip error occurs on one of the data qubits. It should be noted that the problem of the influence of noise on the Grover quantum search algorithm has been extensively studied by various researchers [10, 11].

This is the way we organise the paper: section 2 clearly describes the NLSA proposed by Abrams and Lloyd. Section 3 presents the QAM with NLSA hereafter noted QAM-NLSA with a new method to retrieve one of the sought states in multi-values retrieving scheme. In section 4, we introduce the single qubit noise channels model to the NLSA and we analyze its influence on our model of QAM. At the end, we provide a short conclusion in section 5. But, we start with a short description of parameters used in this paper.

We will use the following parameters:

- \( n \) is the number of qubits of the first register,
- \( |z\rangle \) is the initial state of the first register,
- \( x \) is the needed value while \( |x\rangle \) is the corresponding state,
- \( p \leq 2^n \) the number of stored patterns,
- \( q \leq p \) the number of stored patterns if the values of \( t \) qubits are known,
- \( c = \text{ceil}(\log_2 q) \), i.e. the least integer greater or equal to \( \log_2 q \),
- \( m \leq q \) the number of values \( x \) for which \( f(x) = 1 \), and
- \( r = \text{int}(\log_2 m) \) is the integer part of \( \log_2 m \).

2 Nonlinear search algorithm

Suppose there is an unitary transformation \( U_f \), which is the oracle or the black box, that acts as it follows: for a set of inputs between 0 and \( 2^n - 1 \), there is at most one \( x \) for which \( f(x) = 1 \) and the other values give 0. Let us consider two registers; the first register which is an \( n \)-qubit system is going to compute the inputs and the second which is a single-qubit system will be computing the answer of the oracle. We can define the function \( f \) as it follows:

\[
f : \mathcal{H}^{\otimes n} \rightarrow \mathcal{H}
\]

\[
|y\rangle \mapsto |\delta_{yf}\rangle
\]

(1)

where \( \mathcal{H}^{\otimes n} \) is a Hilbert space of \( 2^n \) dimensions.

The nonlinear algorithm of Abrams and Lloyd aims at disentangling the flag qubit from the first register when we make a measurement on the flag qubit. That measurement can tell us if there is at most a value \( x \) for which \( f(x) = 1 \). They transform the part of the flag qubit that is \( |0\rangle \) to \( |1\rangle \) before a measurement. They claim that it is not possible to do so by using linear operators of quan-
tum information. Algorithm 1 summarizes the Abrams and Lloyd nonlinear algorithm.

**Algorithm 1 Nonlinear search algorithm (NLSA)**

1: Put the first register in the superposed state of all the $N$ values and the flag qubit to $|0\rangle$
2: Apply the oracle operator $U_f$
3: for each qubit of the first register with the flag qubit do
4: Apply the unitary operator $U$
5: 1. apply the nonlinear operator $\mathcal{NL}^-$
6: 2. apply the nonlinear operator $\mathcal{NL}^+$
6: Apply the Hadamard operator $\mathcal{H}$ on the qubit of the first register and the NOT operator $\mathcal{X}$ on the flag qubit
7: end for
8: Observe the flag qubit

Let $|\psi\rangle$ be the state which describes all the system and assume that the first register computes all $N = 2^n$ inputs with equal amplitude:

$$|\psi\rangle = \frac{1}{\sqrt{N}} \sum_{y} |y\rangle |0\rangle. \quad (2)$$

Applying the oracle operator produces

$$U_f |\psi\rangle = \frac{1}{\sqrt{N}} \sum_{y} |y\rangle |f(y)\rangle = \frac{1}{\sqrt{N}} \left( \sum_{y \neq x} |y\rangle |0\rangle + |x\rangle |1\rangle \right). \quad (3)$$

To describe the disentanglement algorithm, we consider the binary forms of values and assume that there is at most one value $x$ which gives $f(x) = 1$. Let $|j_0j_{n-1} \ldots j_1\rangle$ and $|i_0i_{n-1} \ldots i_1\rangle$ be the binary forms of states $|y\rangle$ and $|x\rangle$ respectively, with $j_k, i_k \in \{0, 1\}$. Then, we can rewrite equations (2) and (3) as it follows:

$$|\psi\rangle = \frac{1}{\sqrt{2^n}} \left[ \sum_{j_0j_{n-1} \ldots j_1} |j_0j_{n-1} \ldots j_1\rangle + |i_0i_{n-1} \ldots i_1\rangle |0\rangle + |i_0i_{n-1} \ldots i_1\rangle |1\rangle \right]. \quad (4)$$

and

$$U_f |\psi\rangle = |\Psi\rangle = \frac{1}{\sqrt{2^n}} \left[ \sum_{j_0j_{n-1} \ldots j_1} |j_0j_{n-1} \ldots j_1\rangle |0\rangle + |i_0i_{n-1} \ldots i_1\rangle |1\rangle \right]. \quad (5)$$

Highlighting the least significant qubit (LSQ) of the first register, Eq. (5) can be helpfully written as it follows:

$$|\Psi\rangle = \frac{1}{\sqrt{2^n}} \left[ \sum_{j_0j_{n-1} \ldots j_1} |j_0j_{n-1} \ldots j_1\rangle |0\rangle + |i_0i_{n-1} \ldots i_1\rangle |1\rangle \right]. \quad (6)$$

One must view state (6) as the general binary form of the system after the action of the oracle operator. It summarizes all particular states given by Czachor in [7] and removes ambiguities given by his notation. Indeed, his equation

$$\frac{1}{\sqrt{2^n}} \sum_{j_0j_{n-1} \ldots j_1} \left[ |j_0j_{n-1} \ldots 01\rangle |1\rangle + |j_0j_{n-1} \ldots 11\rangle |0\rangle \right], \quad (7)$$

suggests that there is $2^{n-1}$ values $x$ for which $f(x) = 1$, and not $s = 1$, as he claims.

Let us consider the subsystem of only the LSQ of the first register $|x\rangle$ and the flag qubit $|k\rangle$; $k, \ell \in \{0, 1\}$; the computer will be in one of the following states where we ignore the normalization constants:

$$|00\rangle + |10\rangle, \quad (8a)$$

$$|10\rangle + |01\rangle, \quad (8b)$$

$$|00\rangle + |11\rangle. \quad (8c)$$

The left part of Eq. (6) suggests that state (8a) occurs with the highest probability whereas state (8b) + (8c) to |01⟩ + |11⟩ while leaving the state (8a) unchanged. The NLE part of the algorithm then acts as it follows:
Step 4. Apply the 2-qubit operator

\[
U = \frac{1}{\sqrt{2}} \begin{pmatrix}
1 & 0 & 0 & 1 \\
0 & 1 & 1 & 0 \\
0 & 1 & -1 & 0 \\
1 & 0 & 0 & -1
\end{pmatrix},
\]

on states (8):

\[
u(|00\rangle + |10\rangle) = \frac{1}{\sqrt{2}}(|00\rangle + |01\rangle - |10\rangle + |11\rangle),
\]

(10a)

\[
u(|10\rangle + |01\rangle) = \sqrt{2}|01\rangle,
\]

(10b)

\[
u(|00\rangle + |11\rangle) = \sqrt{2}|00\rangle.
\]

(10c)

Step 5.1. Apply the nonlinear 1-qubit operator \(N_L^-\) on the flag qubit:

\[
N_L^- \left[ \frac{1}{\sqrt{2}}(|00\rangle + |01\rangle - |10\rangle + |11\rangle) \right] = \sqrt{2}|0\rangle (\alpha|0\rangle + \beta|1\rangle),
\]

(11a)

\[
N_L^- (\sqrt{2}|01\rangle) = \sqrt{2}|00\rangle,
\]

(11b)

\[
N_L^- (\sqrt{2}|00\rangle) = \sqrt{2}|00\rangle,
\]

(11c)

where \(\alpha, \beta \in \mathbb{C}, |\alpha|^2 + |\beta|^2 = 1\). As we see on state (11a), the action of the 1-qubit nonlinear operator \(N_L^-\) on state \(\frac{1}{\sqrt{2}}(|00\rangle \pm |11\rangle)\) is not specified. This gives some flexibility to choose the nonlinear gate \(N_L^-\) [3]. On the states (11b) and (11c), the operator \(N_L^-\) maps the two flag qubits \(0\) and \(1\) to the state \(0\). Thus, the operator \(N_L^-\) must be seen as the NOT gate \(\hat{x}\) in case of the state (11b) and the identity gate \(I\) in case of the state (11c).

Step 5.2. Apply the second nonlinear 1-qubit operator \(N_L^+\) on the flag qubit:

\[
N_L^+ (\sqrt{2}|0\rangle (\alpha|0\rangle + \beta|1\rangle)) = \sqrt{2}|01\rangle,
\]

(12a)

\[
N_L^+ (\sqrt{2}|00\rangle) = \sqrt{2}|00\rangle.
\]

(12b)

The nonlinear operator \(N_L^+\) acts as the identity gate \(I\) on the state \(0\). The general form of the unitary matrix \(N_L^+\) which maps the generic 1-qubit \(\alpha|0\rangle + \beta|1\rangle\) to \(|1\rangle\) is

\[
M = \begin{pmatrix}
\mp \gamma \beta^* & \pm \gamma \alpha \\
\alpha^* & \beta^*
\end{pmatrix},
\]

(13)

where \(\alpha, \beta \in \mathbb{C}, |\alpha|^2 + |\beta|^2 = 1\).

It is noteworthy that matrix (13) corrects that of Rigui et al. [1] who claim that matrix \(U\) must be

\[
v = \begin{pmatrix}
1 & \frac{1}{\beta} \\
0 & \frac{\alpha}{\beta}
\end{pmatrix}.
\]

(14)

That matrix \(v\) is unfortunately not a unitary matrix like the matrix (13) as the quantum information processing requires. Furthermore, matrix \(v\) produces a wrong result:

\[
v[\sqrt{2}|0\rangle (\alpha|0\rangle + \beta|1\rangle)] = \sqrt{2}|0\rangle [(\alpha + 1)|0\rangle - \alpha|1\rangle],
\]

(15)

and not \(\sqrt{2}|0\rangle\) as expected.

Step 6. Apply the NOT gate \(\hat{x}\) on the flag qubit and the Hadamard gate \(\hat{w}\) on the first qubit.

We summarize below the nonlinear evolution of the states (8) and give their corresponding circuits:

\[
\begin{align*}
|00\rangle + |10\rangle & \xrightarrow{U} \frac{1}{\sqrt{2}}(|00\rangle + |01\rangle - |10\rangle + |11\rangle) \\
& \xrightarrow{N_L^-} \sqrt{2}|0\rangle (\alpha|0\rangle + \beta|1\rangle) \\
& \xrightarrow{\hat{w} \otimes \hat{x}} |00\rangle + |10\rangle.
\end{align*}
\]

(16a)

\[
\begin{align*}
|10\rangle + |01\rangle & \xrightarrow{U} \sqrt{2}|01\rangle \\
& \xrightarrow{N_L^+} \sqrt{2}|00\rangle \\
& \xrightarrow{\hat{w} \otimes \hat{x}} |01\rangle + |11\rangle.
\end{align*}
\]

(16b)

\[
\begin{align*}
|00\rangle + |11\rangle & \xrightarrow{U} \sqrt{2}|00\rangle \\
& \xrightarrow{N_L^-} \sqrt{2}|0\rangle \\
& \xrightarrow{\hat{w} \otimes \hat{x}} |01\rangle + |11\rangle.
\end{align*}
\]

(16c)
**Example 1.** For a better understanding, let us consider a simple case of a 4-qubit register in the superposition states of all the 16 possible values plus a flag qubit. The marked state is \( |2 = |0010\rangle \). We start with

\[
|\psi\rangle = \frac{1}{4} (|0000\rangle + |0001\rangle + |0010\rangle + |0011\rangle + |1000\rangle + |1011\rangle + |1100\rangle + |1110\rangle + |1101\rangle + |1111\rangle + |0100\rangle + |0101\rangle + |0110\rangle + |0111\rangle + |1000\rangle + |1010\rangle + |1100\rangle + |1101\rangle + |1110\rangle + |1111\rangle) |0\rangle.
\]

The action of the oracle operator \( \psi_f \) produces

\[
|\psi\rangle = \frac{1}{4} (|0000\rangle |0\rangle + |0001\rangle |0\rangle + |0010\rangle |1\rangle + |0011\rangle |0\rangle + |1000\rangle |0\rangle + |1011\rangle |0\rangle + |1100\rangle |0\rangle + |1101\rangle |0\rangle + |1110\rangle |0\rangle + |1111\rangle |0\rangle).
\]

Now, we will describe the process as in [7], but with details on how the system is when the \( \psi_f \) gate is applied. \( \psi_f \) acts on the significant qubit and \( x_f \) acts on flag qubit.

We start by considering the LSQ of the register

\[
|\psi\rangle = \frac{1}{4} (|0000\rangle |0\rangle + |0001\rangle |0\rangle + |1000\rangle |0\rangle + |0100\rangle |0\rangle + |1011\rangle |0\rangle + |1101\rangle |0\rangle + |1100\rangle |0\rangle + |1110\rangle |0\rangle + |1111\rangle |0\rangle).
\]

Applying the NL gate the first time produces

\[
|\psi_f\rangle = \frac{1}{4} \left( (|0000\rangle + |0010\rangle + |0011\rangle + |1000\rangle + |1010\rangle + |1011\rangle + |1100\rangle + |1101\rangle + |1110\rangle + |1111\rangle) (|0\rangle + |1\rangle) \right).
\]

Next, considering the second LSQ

\[
|\psi_{f2}\rangle = \frac{1}{4} (|0000\rangle |0\rangle + |0001\rangle |0\rangle + |0010\rangle |1\rangle + |0011\rangle |0\rangle + |1000\rangle |0\rangle + |1011\rangle |0\rangle + |1100\rangle |0\rangle + |1111\rangle |0\rangle + |1111\rangle |0\rangle + |1111\rangle |0\rangle).
\]

(18a)

(18b)

(18c)
Applying the NLE gate the second time produces

\[ |\Psi_2\rangle = \frac{1}{4} (W_x x_3)NLE^+ \{\langle 010 | (010) + |011 | + |100 | \\
+ |101 | + |110 | + |111 |\} \langle 00 \rangle + (000) + (001) + (111)\}
+ |01 - |10 | + |11 |\} + (000)(001)(\sqrt{2}(00))\}
\]

\[ = \frac{1}{4} (W_x x_3)NLE^+ \{\langle 010 | + |011 | + |100 | + |101 | + |110 | + |111 |\} \langle \sqrt{2}(00) \rangle
+ (000) + (001)(\sqrt{2}(00))\}
\]

\[ = \frac{1}{4} (W_x x_3)\{\langle 010 | + |011 | + |100 | + |101 | + |110 | + |111 |\} \langle \sqrt{2}(00) \rangle
+ (000) + (001)(\sqrt{2}(00))\}
\]

\[ = \frac{1}{4} \{\langle 010 | + |011 | + |100 | + |101 | + |110 | + |111 |\} \langle \sqrt{2}(00) \rangle
+ (000) + (001)(\sqrt{2}(00))\}
\]

\[ = \frac{1}{4} \{\langle 010 | + |011 | + |100 | + |101 | + |110 | + |111 |\} \langle \sqrt{2}(00) \rangle
+ (000) + (001)(\sqrt{2}(00))\}
\]

\[ = \frac{1}{4} \{\langle 010 | + |011 | + |100 | + |101 | + |110 | + |111 |\} \langle \sqrt{2}(00) \rangle
+ (000) + (001)(\sqrt{2}(00))\}
\]

\[ = \frac{1}{4} \{\langle 010 | + |011 | + |100 | + |101 | + |110 | + |111 |\} \langle \sqrt{2}(00) \rangle
+ (000) + (001)(\sqrt{2}(00))\}
\]

Now, considering the third qubit

\[ |\Psi_2\rangle = \frac{1}{4} (\langle 000 | + (001) + (010) + (011)\} \langle 010 \rangle
+ (011)\} \langle 100 \rangle
+ (101)\} \langle 000 \rangle
+ (100)\} \langle 001 \rangle
+ (011)\} \langle 010 \rangle
+ (010)\} \langle 011 \rangle
+ (100)\} \langle 100 \rangle
+ (101)\} \langle 101 \rangle
+ (110)\} \langle 110 \rangle
+ (111)\} \langle 111 \rangle
\]

Finally, we consider the most significant qubit

\[ |\Psi_3\rangle = \frac{1}{4} (\langle 000 | + (001) + (010) + (011)\} \langle 010 \rangle
+ (011)\} \langle 100 \rangle
+ (101)\} \langle 000 \rangle
+ (100)\} \langle 001 \rangle
+ (011)\} \langle 010 \rangle
+ (010)\} \langle 011 \rangle
+ (100)\} \langle 100 \rangle
+ (101)\} \langle 101 \rangle
+ (110)\} \langle 110 \rangle
+ (111)\} \langle 111 \rangle
\]
Applying the \(WLE\) gate the last time produces

\[
|\Psi_4\rangle = \frac{1}{4} (w_4 x_4) NL^+ NL^- U((000) + |001\rangle \\
+ |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle \\
+ |111\rangle) |(10) + |01)\rangle \\
= \frac{1}{4} (w_4 x_4) NL^+ NL^- |((000) + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle)(\sqrt{2}|01)\rangle) \\
= \frac{1}{4} (w_4 x_4) [(|000\rangle + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle) |(00)\rangle + |01)\rangle + |10)\rangle + |11)\rangle] \\
= \frac{1}{4} \left[ \begin{array}{c} 01000 \\ 01001 \\ 01010 \\ 01011 \\ 01100 \\ 01101 \\ 01110 \\ 01111 \\ 11000 \\ 11001 \\ 11010 \\ 11011 \\ 11100 \\ 11101 \\ 11110 \\ 11111 \end{array} \right] \\
\text{A measurement on the flag qubit tells us that there is a value (here it is 2) which gives } f(x) = 1. \\
\text{It appears that we need to apply } n \text{ times the } WLE \text{ gate. So, if we know the values of } t \text{ qubits of our register (i.e. either } t \text{ qubits have been measured or have already been disentangled from others, or the oracle operator acts on a subspace of } (n - t) \text{ qubits), we will apply the } WLE \text{ gate } (n - t) \text{ times. Let us see it with another example that uses the same conditions.}

\text{Example 2. We know the value of the most significant qubit (MSQ) and it is } |0\rangle \text{ (the MSQ has already been measured or the Hadamard gate has been applied on it). Our system collapses to}

\[
|\psi\rangle = \frac{1}{\sqrt{2}} (|0000\rangle + |0010\rangle + |0011\rangle + |0100\rangle + |0101\rangle + |0110\rangle + |0111\rangle + |1000\rangle + |1001\rangle + |1010\rangle + |1011\rangle + |1100\rangle + |1101\rangle + |1110\rangle + |1111\rangle) |0\rangle. \\
\text{Else the oracle operator acts on the 3-qubit and gives 1 for } |010\rangle \text{ which is a part of the values 2 and 10 in their binary forms (0010 and 1010). One must view the system as}

\[
|\psi\rangle = \left( \frac{1}{\sqrt{2}} |0\rangle + \frac{1}{\sqrt{2}} |1\rangle \right) \left[ \frac{1}{2\sqrt{2}} (|000\rangle + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle) |0\rangle. \right. \\
\text{If the MSQ standing for the fourth qubit is noted a } (|a\rangle = |0\rangle \text{ in Eq. (19a) and } |a\rangle = |\sqrt{2} |0\rangle + |\sqrt{2} |1\rangle \text{ in Eq. (19b), the application of the oracle gives}

\[
|\Psi_1\rangle = \frac{1}{\sqrt{2}} |a\rangle (|010\rangle |1\rangle + |011\rangle |1\rangle) \\
+ |000\rangle |0\rangle + |001\rangle |0\rangle \\
+ |100\rangle |0\rangle + |101\rangle |0\rangle \\
+ |110\rangle |0\rangle + |111\rangle |0\rangle). \\
\text{Next, proceeding like in Example 1 without as much as details as possible, each application of the } WLE \text{ gate on the system gives}

\[
|\Psi_2\rangle = \frac{1}{\sqrt{2}} |a\rangle (|001\rangle |1\rangle + |011\rangle |1\rangle) \\
+ |000\rangle |1\rangle + |010\rangle |1\rangle \\
+ |100\rangle |0\rangle + |101\rangle |0\rangle \\
+ |110\rangle |0\rangle + |111\rangle |0\rangle). \\
|\Psi_3\rangle = \frac{1}{\sqrt{2}} |a\rangle (|111\rangle |1\rangle + |111\rangle |1\rangle) \\
+ |100\rangle |1\rangle + |100\rangle |1\rangle \\
+ |101\rangle |1\rangle + |101\rangle |1\rangle \\
+ |110\rangle |1\rangle + |101\rangle |1\rangle \\
= \frac{1}{2\sqrt{2}} |a\rangle (|000\rangle + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle) |1\rangle. \right.
\]
3 Concise algorithm for quantum associative memories

3.1 Principles of algorithm

We briefly describe here the whole process of the QAM-NLSA. Like in the Rigui et al. paper’s [1], we do the process of learning or storing patterns of our memory by using an operator named BDD. The use of the Binary Superposed Quantum Decision Diagram (BSQDD) proposed by Rosenbaum [2] allows to obtain the operator BDD. We can use any basis states $|z\rangle$ of Hilbert space of $2^n$ dimensions (not only $|00\ldots0\rangle$) to compute the BSQDD. According to Rosenbaum, the idea behind BSQDD is to represent a quantum superposition as a decision diagram where each node corresponds to a gate. The path, used to reach the gate which corresponds to the node on each branch of the BSQDD by starting from the root of the decision diagram, controls the said gate. Thereby we need three steps to construct a BSQDD.

1. **Finding the unsimplified BSQDD by using the Hadamard gates, Feynmann gates, and inverters** (see Fig. 4 for a case of a register with fourth qubits). The number of nodes of this unsimplified BSQDD represents the upper bound on the number of the gates that the quantum array generated by the BSQDD needs for being constructed.

2. **Reducing the BSQDD to obtain the final BSQDD.** The goal is to have the lower bound on the number of quantum gates. To attain this goal, one needs to merge some nodes (gates) according to the links that can occur between the qubits (like control qubit and target qubit). Fig. 5 shows the BSQDDs with merged nodes (Fig. 5a and Fig. 5b) and the final BSQDD (Fig. 5c). The three BSQDDs of Fig. 5 are equivalents.

3. **Converting the BSQDD to a quantum array which generates the desired quantum state (see Fig. 6).**

**Example 3.** Fig. 7 gives the three steps which allow to construct the state $\sqrt{1/5}(|000\rangle + |010\rangle + |110\rangle + |001\rangle + |101\rangle)$ from the starting state $|000\rangle$. The elementary gates used are respectively $R(\theta) = \begin{pmatrix} \sqrt{3/5} & \sqrt{2/5} \\ \sqrt{2/5} & -\sqrt{3/5} \end{pmatrix}$, $R(\alpha) = \begin{pmatrix} \sqrt{2/3} & 1/\sqrt{3} \\ -1/\sqrt{3} & \sqrt{2/3} \end{pmatrix}$, the Hadamard gate $W$, and the NOT gate $X$.

Then $|\psi_3\rangle = \sqrt{2/5}|000\rangle + \sqrt{1/5}|100\rangle$ and $|\psi_2\rangle = \sqrt{2/5}|000\rangle + \sqrt{1/5}|010\rangle + \sqrt{1/5}|100\rangle + \sqrt{1/5}|110\rangle$.

Fig. 8 presents the two first steps needed to compute the state used in Example 1. The corresponding quantum array is the first dashed box of Fig. 10.

The quantum NLSA that allows us to have the information we want after a measurement on the flag qubit not on the first register does the retrieving data process. However, it can be useful to measure the register especially in case of multi-values which satisfy $f(x) = 1$. But, as it appears in the previous section, we will get each $2^n$ values with the same probability. In the method proposed by Rigui et al. [1] there are some ambiguities on

---

**Figure 4** Unsimplified BSQDD. Each layer corresponds to a set of gates which act on a specific qubit. For example, gate $A$ acts on the most significant qubit while gates $J$ to $W$ act on the least significant qubit. The state $|z\rangle$ is the non-normalized state obtained when a specific gate acts on a specific qubit. The state $|\psi\rangle$ being the sum of non-normalized states, is the normalized state obtained after the gates of the layer $l$ act on qubit $i$. The state $|z\rangle$ is the basis state used for starting and state $|\psi_1\rangle$ is the desired quantum superposed state.
J.-P. T. Njafa and S. G. N. Engo: Concise quantum associative memories with nonlinear search algorithm.

Figure 5 The merging nodes to obtain the final BSQDD. The following two rules have been used to obtain the final BSQD. The first rule states that in two different branches of different nodes which correspond to the same next node, that same nodes merge. The second rule states that in different branches of different nodes which generate the same branch, that same branches merge.

| $|z\rangle$ | $|\psi_4\rangle$ | $|\psi_3\rangle$ | $|\psi_2\rangle$ | $|\psi_1\rangle$ |
| --- | --- | --- | --- | --- |
| $|j_4\rangle$ | $G_4$ | $|j_3\rangle$ | $G_3$ | $|j_2\rangle$ | $G_2$ | $|j_1\rangle$ | $G_1$ |

(a) First merging nodes

Figure 6 The quantum array generated by the final BSQDD. The array is obtained by adding the gates that correspond to the nodes in each layer of the final BSQDD. The starting point is the last layer and we always place the new gates to the right of the previously placed gates in the quantum array. Therefore, the first gate is $G_4$ while the last one is $G_1$.

(b) The conditional operator $C(BDD)^1$ which acts on the first register and brings it back to its initial state $|z\rangle$ when the flag qubit is $|1\rangle$;

(c) The operator

$$CS = I_{2^n} \otimes |0\rangle \langle 0| + (I_{2^n} - (|z\rangle \langle z| + |x\rangle \langle x|)) + |x\rangle \langle z| + |z\rangle \langle x| \otimes |1\rangle \langle 1|$$

which is a $(2^{n+1}) \times (2^{n+1})$ conditional operator which maps the first register to the sought state $|x\rangle$ when the flag qubit is $|1\rangle$. Put differently,

- if the flag qubit is $|0\rangle$ we do nothing;
- if the flag qubit is $|1\rangle$ we apply the $2^n \times 2^n$ operator $S$ on the first register.

We also point out the fact that as the BSQDD method can compute any sought state, it can be useful to compute the operator $CS$. Indeed, in the case of a complex sought state $|x\rangle$ where the Hadamard gates or other methods are inadequate, the BSQDD method can allow us to have the appropriate form of the operator $S$.

We also point out the fact that as the BSQDD method can compute any sought state, it can be useful to compute the operator $CS$. Indeed, in the case of a complex sought state $|x\rangle$ where the Hadamard gates or other methods are inadequate, the BSQDD method can allow us to have the appropriate form of the operator $S$.

1. The operator $BDD$ makes the learning process.
2. The following operators make the retrieving process:
   (a) The operator $NL$ which marks the sought states with $\psi$ computes repeatedly the nonlinear evolution $NLE$ on the system and disentangles the first register from the flag qubit;
   (b) The conditional operator $C(BDD)^1$ which acts on the first register and brings it back to its initial state $|z\rangle$ when the flag qubit is $|1\rangle$;
   (c) The operator

$$CS = I_{2^n} \otimes |0\rangle \langle 0| + (I_{2^n} - (|z\rangle \langle z| + |x\rangle \langle x|) + |x\rangle \langle z| + |z\rangle \langle x| \otimes |1\rangle \langle 1|$$

which is a $(2^{n+1}) \times (2^{n+1})$ conditional operator which maps the first register to the sought state $|x\rangle$ when the flag qubit is $|1\rangle$. Put differently,

- if the flag qubit is $|0\rangle$ we do nothing;
- if the flag qubit is $|1\rangle$ we apply the $2^n \times 2^n$ operator $S$ on the first register.

We also point out the fact that as the BSQDD method can compute any sought state, it can be useful to compute the operator $CS$. Indeed, in the case of a complex sought state $|x\rangle$ where the Hadamard gates or other methods are inadequate, the BSQDD method can allow us to have the appropriate form of the operator $S$. It is noteworthy that in the case of multi-patterns retrieving scheme, the sought state $|x\rangle$ can be a superposed state of all the sought states (for example $|x\rangle = \frac{1}{\sqrt{2}}(|0010\rangle + |1010\rangle)$ in Example 2).

(d) At the end, to observe the system we make a measurement on the first register and/or on the flag qubit to erase any ambiguity.
Figure 7 BSQDD to obtain state $|\psi_1\rangle = \sqrt{\frac{1}{5}}(|000\rangle + |010\rangle + |110\rangle + |001\rangle + |101\rangle)$.

Figure 8 Two first steps needed to compute state used in Example 1.

Figure 9 Schematic structure of QAM-NLSA. The BDD computes the learning process. The gates $U_f$, NLE, C$(BDD)^\dagger$ and CS make the retrieving process. Gate $U_f$ marks the sought states. Gate NLE repeatedly computes the nonlinear evolution. Conditional gate C$(BDD)^\dagger$ brings back the first register to its initial state $|\bar{z}\rangle$. Conditional gate CS maps the first register to the sought state $|x\rangle$. 
Example 4. Example 1 suggests that the operator $S = I_2 \otimes I_2 \otimes \chi \otimes I_2$. Therefore, the $\chi$ gate acts only on the second qubit, while the other qubits remain unchanged. Fig. 10 gives the evolution of the system.

3.2 Analysis of the complexity of the nonlinear evolution algorithm

We made all the above description with the assumption that there is at most one value $x$ for which $f(x) = 1$. Let us now consider the case where there can be more than one value satisfying $f(x) = 1$. In the simple case where there are at most two values satisfying $f(x) = 1$, the state (6) must be rewritten as

$$|\Psi\rangle = \frac{1}{\sqrt{2^n}} \left[ \left( \sum_{j_1, \ldots, j_{n-1}, j_{n+1}, \ldots, j_n = 0, 1} \right| j_1 j_n \ldots j_{n-1} |0\rangle \right. + |i_1 i_{n-1} \ldots (1-i_1) |0\rangle + |i_1 i_{n-1} \ldots i_1 |1\rangle + |e_n e_{n-1} \ldots e_1 |0\rangle + |e_n e_{n-1} \ldots e_1 |1\rangle \right].$$

Highlighting the LSQ of the first register and the flag qubit, the second part of state (22) must be in one of the following states:

- if $i_1 i_{n-1} \ldots i_2 \neq e_n e_{n-1} \ldots e_2$,
  $$|00\rangle + |11\rangle + |10\rangle + |01\rangle,$$  (23a)
  $$|00\rangle + |11\rangle + |00\rangle + |11\rangle.$$  (23b)

- or if $i_1 i_{n-1} \ldots i_2 = e_n e_{n-1} \ldots e_2$,
  $$|01\rangle + |11\rangle.$$  (24)

because there is no repetition of value.

The action of the $NLE$ gate on states (23) is the same as the one described in section 2. Taking a careful look at the states (23), it seems that the $NLE$ was already applied one time and that suggests the $NLE$ gate will be repeated $(n-1)$ times. State (24) also supposes that the $NLE$ gate was already applied thus the $NLE$ gate will begin on the second LSQ and will be repeated $(n-1)$ times. Therefore, if there are at most two values $x$ for which $f(x) = 1$, the number of steps of the QAM-NLSA is

$$\mathcal{O}(n-1),$$  (25a)

as the $NLE$ gate starts on the second LSQ. It is easy to find that in the case where there are at most three values $x$ satisfying $f(x) = 1$, when we start the repeated action of the $NLE$ on the second LSQ of the first register, the number of steps of the QAM-NLSA is also

$$\mathcal{O}(n-1).$$  (25b)

According to the above observation, if there are at most $m$ values $x$ for which $f(x) = 1$, the $NLE$ gate action starts on the $(r+1)^{th}$ LSQ of the first register. It will be repeated $(n-r)$ times. $r = \lfloor \log_2 m \rfloor$ is the integer part of $\log_2 m$. Thus, the number of steps of the QAM-NLSA is

$$\mathcal{O}(n-r).$$  (26)

Example 5. In order to highlight result (26), let us consider the parameters of Example 1. The marked states are $|2\rangle = |0010\rangle$, $|5\rangle = |0101\rangle$, $|8\rangle = |1000\rangle$, $|10\rangle = |1010\rangle$, $|11\rangle = |0110\rangle$. $|0\rangle$ remains unchanged.
\[ |11\rangle = |1011\rangle, \ |13\rangle = |1101\rangle, \text{ and } |15\rangle = |1111\rangle. \] The number of marked states is \( m = 7 \) and \( \log_2 m = 2.807. \) Consequently, \( r = 2 \) and the \( \text{XLE} \) gate action starts on the third \( \text{LSQ} \) of the register. Let us check that in detail. As in Example 1, we start with

\[ |\Psi\rangle = \frac{1}{4} (|0000\rangle + |0001\rangle + |0010\rangle + |0011\rangle + |0100\rangle + |0101\rangle + |0110\rangle + |0111\rangle + |1000\rangle + |1001\rangle + |1010\rangle + |1011\rangle + |1100\rangle + |1101\rangle + |1110\rangle + |1111\rangle) |0\rangle. \]  

(27a)

The action of the oracle operator \( \text{U}_f \) produces

\[ |\Psi\rangle = \frac{1}{4} (|0000\rangle |0\rangle + |0001\rangle |0\rangle + |0010\rangle |1\rangle + |0011\rangle |0\rangle + |0100\rangle |0\rangle + |0101\rangle |1\rangle + |0110\rangle |0\rangle + |0111\rangle |0\rangle + |1000\rangle |1\rangle + |1001\rangle |0\rangle + |1010\rangle |1\rangle + |1011\rangle |1\rangle + |1100\rangle |0\rangle + |1101\rangle |1\rangle + |1110\rangle |0\rangle + |1111\rangle |1\rangle) |0\rangle. \]  

(27b)

Highlighting the third qubit,

\[ |\Psi\rangle = \frac{1}{4} (|0000\rangle |0\rangle + |0001\rangle |0\rangle + |0010\rangle |0\rangle + |0011\rangle |0\rangle + |0100\rangle |0\rangle + |0101\rangle |1\rangle + |0110\rangle |0\rangle + |0111\rangle |0\rangle + |1000\rangle |1\rangle + |1001\rangle |0\rangle + |1010\rangle |1\rangle + |1011\rangle |1\rangle + |1100\rangle |0\rangle + |1101\rangle |1\rangle + |1110\rangle |0\rangle + |1111\rangle |1\rangle) |0\rangle. \]  

(28a)

and applying the \( \text{XLE} \) gate the last time produces

\[ |\Psi_1\rangle = \frac{1}{4} (|0000\rangle |0\rangle + |1000\rangle |1\rangle + |0001\rangle |0\rangle + |1001\rangle |1\rangle + |0010\rangle |0\rangle + |1010\rangle |1\rangle + |0011\rangle |0\rangle + |1011\rangle |1\rangle + |0100\rangle |0\rangle + |1100\rangle |1\rangle + |0101\rangle |0\rangle + |1101\rangle |1\rangle + |0110\rangle |0\rangle + |1110\rangle |1\rangle + |0111\rangle |0\rangle + |1111\rangle |1\rangle) |0\rangle. \]  

(28b)

Finally, considering the \( \text{MSQ} \), we have:

\[ |\Psi_1\rangle = \frac{1}{4} (|0000\rangle |0\rangle + |1000\rangle |1\rangle + |0001\rangle |0\rangle + |1001\rangle |1\rangle + |0010\rangle |0\rangle + |1010\rangle |1\rangle + |0011\rangle |0\rangle + |1011\rangle |1\rangle + |0100\rangle |0\rangle + |1100\rangle |1\rangle + |0101\rangle |0\rangle + |1101\rangle |1\rangle + |0110\rangle |0\rangle + |1110\rangle |1\rangle + |0111\rangle |0\rangle + |1111\rangle |1\rangle) |0\rangle. \]  

(28c)

and applying the \( \text{XLE} \) gate the first time produces

\[ |\Psi_2\rangle = \frac{1}{4} (|0000\rangle + |1000\rangle + |0001\rangle + |1001\rangle + |0010\rangle + |1010\rangle + |0011\rangle + |1011\rangle + |0100\rangle + |1100\rangle + |0101\rangle + |1101\rangle + |0110\rangle + |1110\rangle + |0111\rangle + |1111\rangle) |0\rangle. \]  

(28d)

It effectively appears that we repeat the \( \text{XLE} \) gate \( (n - r) = 4 - 2 = 2 \) times.

If we know the values of \( t \) qubits of our first register (i.e. \( t \) qubits have been measured or have already been disentangled from others, or the oracle operator acts on a subspace of \( (n - t) \) qubits) and there is at most \( m \) values \( x \) for which \( f(x) = 1 \), the \( \text{XLE} \) gate will act repeat-
edly \((n - t) - r\) times that starts on the \((r + 1)^{th}\) LSQ. As the already known \(t\) qubits will be ignored, it is clear that \(m \leq 2^{n-t}\). Consequently, the number of steps of the QAM-NLSA is

\[
\mathcal{O}(n - t) - r).
\]

(29)

Now, if in the first register, which is an \(n\)-qubit system, the computed patterns are \(p \leq 2^n\) the NLE gate will act repeatedly \((b - r)\) times. Here we state \(b = \text{ceil}(\log_2 p)\) (i.e. the least integer greater or equal to \(\log_2 p\)) and \(r = \text{int}(\log_2 m)\) where \(m\) is the number of values \(x\) for which \(f(x) = 1\). Therefore, the number of steps of the QAM-NLSA is

\[
\mathcal{O}(b - r),
\]

(30)

for which the upper bound is Eq. (26). Note that the starting point of the NLE gate action will always be the \((r + 1)^{th}\) LSQ.

If we know the values of \(t\) qubits of our first register, it supposes that we must view the system in terms of \(q \leq p \leq 2^n\) patterns. Consequently, the number of values \(m\) for which \(f(x) = 1\) is \(m \leq q\). For \(c = \text{ceil}(\log_2 q)\), the NLE gate will act repeatedly \((c - r)\) times. Therefore, the number of steps of the QAM-NLSA takes the general form

\[
\mathcal{O}(c - r).
\]

(31)

Example 6.

• If we consider again the parameters of Example 1, we find that \(p = 16\). The number of the known qubits is \(t = 0\). Consequently, \(q = p = 16\) and \(\log_2 q = \log_2 16 = 4.0\). Thus \(c = 4\), \(m = 1\), and \(r = 0\). The NLE gate will act repeatedly \((c - r) = 4 - 0 = 4\) times.

• In Example 2, \(t = 1\). Consequently, \(q = 8\) according to the assumption taken in this example. Thus \(c = 3\), \(m = 1\), and \(r = 0\). The NLE gate will act repeatedly \((c - r) = 3 - 0 = 3\) times.

• In Example 5, \(t = 0\) and \(q = 16\) but \(m = 7\). Then, \(\log_2 m = 2.807\). That is, \(r = 2\). The NLE gate will act repeatedly \((c - r) = 4 - 2 = 2\) times.

It is noteworthy that when state \(|01) + |11\rangle\) appears the last time, the NLE gate acts repeatedly. The state \(|01) + |11\rangle\) then does not evolve. Its nonlinear evolution must be like that of the state (8a). Such we describe the nonlinear evolution of the state \(|01) + |11\rangle\) as it follows:

Step 4. Apply the operator \(\mathcal{U}\):

\[
\mathcal{U}(|01) + |11\rangle) = \frac{1}{\sqrt{2}}(|00) + |01) + |10) - |11\rangle).
\]

(32)

Step 5.1. Apply the nonlinear operator \(\text{NL}^-\):

\[
\text{NL}^-\left[\frac{1}{\sqrt{2}}(|00) + |01) + |10) - |11\rangle)\right] = \sqrt{2}|0) (|0) + \epsilon |1\rangle),
\]

(33)

where \(\delta, \epsilon \in \mathbb{C}, |\delta|^2 + |\epsilon|^2 = 1\). As we see on state (33), the action of the nonlinear operator \(\text{NL}^-\) is not also specified like on the state (11a).

Step 5.2. Apply the second nonlinear operator \(\text{NL}^+\):

\[
\text{NL}^+\left[\sqrt{2}|0) (|0) + \epsilon |1\rangle\right] = \sqrt{2}|00\rangle.
\]

(34)

The general form of the unitary matrix \(\text{NL}^+\) which maps the generic 1-qubit \(\delta |0) + \epsilon |1\rangle\) to \(|0\rangle\) is

\[
\Pi = \left(\begin{array}{cc}
\delta^* & \epsilon^* \\
\mp \gamma \epsilon & \pm \gamma \delta
\end{array}\right), \gamma = \pm i \text{ or } \pm 1 (\dot{i}^2 = -1),
\]

(35)

where \(\delta, \epsilon \in \mathbb{C}, |\delta|^2 + |\epsilon|^2 = 1\).

Step 6. Apply the NOT gate \(\hat{x}\) on the flag qubit and the Hadamard gate \(\hat{w}\) on the first qubit.

We summarize below this nonlinear evolution of state \(|01) + |11\rangle\) and give its corresponding circuit (see Fig. 11)

\[
|01) + |11\rangle \xrightarrow{\mathcal{U}} \frac{1}{\sqrt{2}}(|00) + |01) + |10) - |11\rangle) \xrightarrow{\text{NL}^-} \sqrt{2}|0) (|0) + \epsilon |1\rangle) \xrightarrow{\text{NL}^+} \sqrt{2}|00\rangle \xrightarrow{\hat{x}} |01) + |11\rangle.
\]

(36)

Figure 11 Equivalent circuit of the nonlinear evolution (36).

Finally, Algorithm 2 describes the QAM-NLSA where

• \(n\) is the number of qubits of the first register,
• \(p \leq 2^n\) the number of stored patterns,
$\eta \in [0, 1]$ after the action of each gate of the NLE gate. We assume that:

- each gate operates before the error proceeds and it is the same error;
- the first register is an $n$-qubit system and that the probability $\eta$ of a state to be affected by the noise is independent of the total number of network qubits;
- and errors are located at each time step in the network affecting $|\ell\rangle$ and $|k\rangle$.

4.1 Single qubit quantum noise channels

If $\rho_{in} = |\psi\rangle \langle \psi|$ is the density matrix of state $|\psi\rangle$, the effect of the environment leads in the Kraus representation to

$$\rho_{out} = \mathcal{K} (\rho_{in}) = \sum_i E_i \rho_{in} E_i^\dagger, \quad \sum_i E_i^\dagger E_i = I, \quad (37)$$

where $E_i$ are the error operators or Kraus operators which completely describe here the single qubit quantum noise channels briefly presented in Table 1.

4.2 Quantum associative memories with noise - bit flip model

We suppose that during the nonlinear evolution (NLE), step 4 to step 6 in Algorithm 1, the quantum noise occurs with the probability $\eta \in [0, 1]$ after the action of each gate of the NLE gate. We assume that:

- each gate operates before the error proceeds and it is the same error;
- the first register is an $n$-qubit system and that the probability $\eta$ of a state to be affected by the noise is independent of the total number of network qubits;
- and errors are located at each time step in the network affecting $|\ell\rangle$ and $|k\rangle$.

4 Taking into account the quantum noise

We will briefly analyze in present section how our QAM-NLSA evolves in the presence of the quantum noise. As the NLSA evolves qubit per qubit, we will consider only the single qubit quantum noise channels as described in [12, 13]. The quantum states to be considered will be the density operators instead of the state vectors.

Figure 12 Schematic structure of the NLSA with quantum noise.

Now, we analyse the effect of each quantum noise channels during the evolution of the state (6). Thus, while we reduce the system to the two qubits $|\ell\rangle$ and $|k\rangle$ we have

$$\rho_{out} = \sum_i E_i \left( w \otimes x \left[ \sum_i E_i \left( \mathcal{NL}^+ \left[ \sum_i E_i \right] \left[ \mathcal{NL}^- \left[ \sum_i E_i \right] \right] \right) \right] \right)$$

$$\times \left( \mathcal{NL}^- \left[ \sum_i E_i \left( u \rho_{in} u^\dagger \right) E_i^\dagger \left( \mathcal{NL}^- \right)^\dagger \right] \right) E_i^\dagger.$$

(38)
Because for each step of NLE we consider a two qubits system, the error operators to apply must be a tensor product of operators of single quantum noise models.

For the bit flip model, the tensor product \( |I_r, X_f \rangle \otimes |I_f, X_f \rangle \) gives below the set of error operators where \( r \) indexes the qubit of first register and \( f \) indexes the flag qubit:

\[
E_1 = \sqrt{1 - \eta} \sqrt{1 - \eta} I_r \otimes I_f \\
E_2 = \sqrt{1 - \eta} \sqrt{\eta} I_r \otimes X_f \\
E_3 = \sqrt{\eta} \sqrt{1 - \eta} X_r \otimes I_f \\
E_4 = \sqrt{\eta} \sqrt{\eta} X_r \otimes X_f.
\] (39)

Notice that according to Eq. (38), \( \rho_{\text{out}} \) is the sum of 256 matrix for the bit flip model.

Due to the entanglement the system must be observed completely. So, without the constant normalization, the input matrix is

\[
\rho_{\text{in}} = |\psi\rangle \langle \psi| = \left( \sum |j_0j_{n-1} \ldots j_f 0 \rangle \langle j_0j_{n-1} \ldots j_f 0| + |j_0j_{n-1} \ldots j_f 1 \rangle \langle j_0j_{n-1} \ldots j_f 1| \right) \\
+ |i_ni_{n-1} \ldots (1 - i) \rangle \langle i_ni_{n-1} \ldots i| \right)
\] (39)

The sought output density matrix is

\[
\Omega = \left( \sum |j_0j_{n-1} \ldots j_1 \rangle \langle j_0j_{n-1} \ldots j_1| \right) \otimes |1\rangle \langle 1|.
\] (41)

To evaluate the influence of the quantum noise on the effectiveness of the algorithm, we will compute the fidelity between the sought output and the obtained output. Let \( \sigma \) be the density matrix of the sought output. The fidelity then is

\[
\mathcal{F}(\sigma, \Gamma) = tr \sqrt{\sqrt{\sigma} \Gamma \sqrt{\sigma}},
\] (42)

with \( 0 \leq \mathcal{F}(\sigma, \Gamma) \leq 1 \). \( \mathcal{F}(\sigma, \Gamma) = 0 \) if \( \sigma \) and \( \Gamma \) are orthogonal and \( \mathcal{F}(\sigma, \Gamma) = 1 \) if \( \sigma = \Gamma \).

If we only focus on the effectiveness of the quantum noise on the NLE, that is before the retrieving process, we can consider that the sought output is a pure state \( |\psi\rangle = \sum |x\rangle = |\phi_1\rangle \otimes |\phi_2\rangle \otimes \cdots \otimes |\phi_n\rangle \), |\phi_i\rangle =
\[ \alpha |0\rangle + \beta |1\rangle \quad (|\alpha|^2 + |\beta|^2 = 1), \]
and then we can write Eq. (42) as

\[
\mathcal{F}_0(|\psi\rangle, \rho_{out}) = \sqrt{\langle \psi | \rho_{out} | \psi \rangle}.
\] (43)

### 4.3 Simulation

We suppose that

\[
NL^- = CX_f^{1}, CW_f^{0}, CW_f^{1}
\]

\[
= \frac{1}{\sqrt{2}} \begin{pmatrix}
1 & 1 & 0 & 0 \\
0 & 0 & 1 & -1 \\
0 & 0 & 1 & 1 \\
1 & -1 & 0 & 0
\end{pmatrix},
\] (44)

and according to equations (44) we can choose \( \mathbf{X} = \mathbf{X}' \). We also consider that \( \eta_r = \eta_f = \epsilon \) because the error operator is applied on states \( |\ell\rangle \) and \( |k\rangle \) simultaneously (i.e. the error arises on these qubits at the same time). Thus, we can consider that the probability of error is identical for both qubits. The set of operators is now

\[
\begin{align*}
E_1 &= (1 - \epsilon)I_r \otimes I_f, \\
E_2 &= \sqrt{\epsilon(1 - \epsilon)}I_r \otimes x_f, \\
E_3 &= \sqrt{\epsilon(1 - \epsilon)}x_r \otimes I_f, \\
E_4 &= \epsilon x_r \otimes x_f.
\end{align*}
\] (45)

Because we make a measurement on the flag qubit, we only extract its density matrix \( \rho_{out} \) to evaluate fidelity.

### 4.3.1 Case where the sought state does not exist in the first register

Here the sought output of the flag qubit is \( |0\rangle \).

If the first register has \( n \) qubits,

\[
\rho_{out} = \begin{pmatrix}
1 + \frac{1}{2}[(1 - 2\epsilon)^3 - 1] & 0 \\
0 & -\frac{1}{2}[(1 - 2\epsilon)^3 - 1]
\end{pmatrix},
\] (46a)

then

\[
\mathcal{F}_0(|0\rangle, \rho_{out}) = \sqrt{1 + \frac{1}{2}[(1 - 2\epsilon)^3 - 1]}.
\] (46b)

Indeed,

- if the first register has only one qubit,

\[
\rho_{out} = \begin{pmatrix}
1 + \frac{1}{2}[(1 - 2\epsilon)^3 - 1] & 0 \\
0 & -\frac{1}{2}[(1 - 2\epsilon)^3 - 1]
\end{pmatrix}.
\] (47)

- If the first register has three qubits,

\[
\rho_{out} = \begin{pmatrix}
1 + \frac{1}{2}[(1 - 2\epsilon)^3 - 1] & 0 \\
0 & -\frac{1}{2}[(1 - 2\epsilon)^3 - 1]
\end{pmatrix}.
\] (48)

As we see on Fig. 13 and according to Eq. (46b), while \( \epsilon < 0.5 \) the fidelity is higher than 0.7. But it decreases for \( \epsilon \geq 0.5 \) if \( n \) is an odd number and increases if \( n \) is an even number. For \( n \) as an odd number (see Fig. 13a, one qubit), the area between 0.4 and 0.6 can be viewed as a stability area. That means the area where fidelity is maintained around 0.7. For \( n \) as an even number (see Fig. 13b, six qubits), this stability area grows with the number of qubits. Therefore, if the sought state does not exist in the first register, the QAM-NLSA is more resistant to the noise when the register has an even number of qubit.

### 4.3.2 Case where the sought state exists in the first register

Here, the sought output of the flag qubit is \( |1\rangle \).

If the first register has \( n \) \((n > 1)\) qubits,

\[
\rho_{out} = \begin{pmatrix}
\frac{1}{2} f(\epsilon^{4n}) & 0 \\
0 & 1 + \frac{1}{2} f(\epsilon^{4n})
\end{pmatrix},
\] (49a)

where \( f \) is a polynomial function which grows with \( \epsilon^{4n} \). Then,

\[
\mathcal{F}_0(|1\rangle, \rho_{out}) = \sqrt{1 + \frac{1}{2}[(1 - 2\epsilon)^4 - 1]}, \quad \text{for } n = 1,
\] (49b)

or

\[
\mathcal{F}_0(|1\rangle, \rho_{out}) = \sqrt{1 + \frac{1}{2} f(\epsilon^{4n})}, \quad \text{for } n > 1.
\] (49c)

Indeed,

- if the first register has only one qubit,

\[
\rho_{out} = \begin{pmatrix}
\frac{1}{2}[(1 - 2\epsilon)^3 - 1] & 0 \\
0 & 1 + \frac{1}{2}[(1 - 2\epsilon)^4 - 1]
\end{pmatrix}.
\] (50)
Figure 13 Evolution of the fidelity \( F_0(\rho_0, \rho_{\text{out}}) \) at the end of the NLE steps in the case where the sought state does not exist in the first register. (a) for 1, 3, 5 and 1001 qubits and (b) for 2, 4, 6 and 1000 qubits. The dashed blue and green rectangles highlight the stability area.

- If the first register has two qubits,
  \[
  \rho_{\text{out}} = \begin{pmatrix}
  \frac{1}{2}(&(1-2\epsilon)^6 - 1) + 5((1-2\epsilon)^{15} - 1) \\
  +15((1-2\epsilon)^{14} - 1) + 13((1-2\epsilon)^{13} - 1) \\
  +\frac{1}{2}0((1-2\epsilon)^{12} - 1) - 6((1-2\epsilon)^{11} - 1) \\
  +5((1-2\epsilon)^{10} - 1) - ((1-2\epsilon)^{9} - 1) \\
  -((1-2\epsilon)^{8} - 1) + ((1-2\epsilon)^{7} - 1)
  \\
  0
  \end{pmatrix}
  \begin{pmatrix}
  0 \\
  1 + \frac{1}{2}((1-2\epsilon)^{8} - 1) + ((1-2\epsilon)^{7} - 1)
  \end{pmatrix}.
  \]
  (51)

- If the first register has four qubits,
  \[
  \rho_{\text{out}} = \begin{pmatrix}
  \frac{1}{2}((1-2\epsilon)^{16} - 1) + 5((1-2\epsilon)^{15} - 1) \\
  +15((1-2\epsilon)^{14} - 1) + 13((1-2\epsilon)^{13} - 1) \\
  +\frac{1}{2}0((1-2\epsilon)^{12} - 1) - 6((1-2\epsilon)^{11} - 1) \\
  +5((1-2\epsilon)^{10} - 1) - ((1-2\epsilon)^{9} - 1) \\
  -((1-2\epsilon)^{8} - 1) + ((1-2\epsilon)^{7} - 1)
  \\
  0
  \end{pmatrix}
  \begin{pmatrix}
  0 \\
  1 + \frac{1}{2}((1-2\epsilon)^{8} - 1) + ((1-2\epsilon)^{7} - 1)
  \end{pmatrix}.
  \]
  (52)

As we see in Fig. 14 and according to Eq. (49c), whatever the value of \( \epsilon \) the fidelity is greater than 0.7. In other words, if the sought state exists in the first register, the QAM-NLSA is resistant to the noise whatever the number of qubits. We also see that the stability area highlighted by the dashed green rectangle on Fig. 14 grows with the number of qubits. That stability area is the same as those shown on Fig. 13.

From the above simulations, it appears that the noise affects the QAM-NLSA during its implementation. In the particular case of the bit flip channel, the fidelity between the unaffected and affected systems is about 70% and this value does not change even if the number of qubits in the first register grows.

5 Conclusion

We have proposed a model of the QAM-NLSA similar to that of Rigui et al. [1]. However, the model we
propose differs with the possibility of retrieving one of the sought states in the multi-values retrieving when a measurement on the first register is done.

Firstly, we have described the NLSA put forth by Abrams and Lloyd in [3] with notations that overcome some ambiguities due to the notations of Rigui et al. and Czachor [7] and by summarizing each step of the nonlinear evolution with an equivalent circuit. A general form of the unitary matrix NL⁺ which acts on the generic flag qubit α|0⟩ + β|1⟩ was given thereby correcting the wrong one that Rigui et al. gave. Secondly, we have described our model of the Quantum Associative Memory (QAM), where we have introduced a (2ⁿ+1) × (2ⁿ+1) conditional operator CS which maps the first register to the sought state |x⟩ when the flag qubit is |1⟩ and n being the number of qubits of the first register. If n is the number of qubits of the first register, p ≤ 2ⁿ the number of stored patterns, q ≤ p the number of the stored patterns if the values of t qubits are known (i.e. t qubits have been measured or have already been disentangled from others or the oracle operator acts on a subspace of (n − t) qubits), m ≤ q the number of values x for which f(x) = 1, c = ceil(log₂ q) the least integer greater or equal to log₂ q, and r = int(log₂ m) the integer part of log₂ m, then the time complexity of our algorithm is $\mathcal{O}(c − r)$. It is better than Grover’s algorithm and its modified forms which need $\mathcal{O}(\sqrt{\frac{q}{m}})$ steps when they are used as the retrieval algorithm. We have done an example to illustrate the results given by our analysis. It is noteworthy that our algorithm also allows to measure the flag qubit to erase any ambiguity on the result given by a measurement on the first register. This possibility is introduced by the use of two conditional gates which do not affect the flag qubit after the nonlinear evolution. Finally, we have briefly analysed the influence of the quantum noise namely the bit flip channel on our model of the QAM-NLSA. We found that the bit flip channel leaves the QAM unaffected fully at 70% if the sought state is present in the first register or if the register has an even number of qubits when the sought state does not exist. However, when the first register has an odd number of qubits and the sought state does not exist, the bit flip channel is extremely destructive when the probability $\epsilon > 0.5$. Our analysis shows the robustness of the QAM-NLSA against the noise, particularly when the sought states exist in the first register.

Further work will be undertaken in another study to detail the influence of the quantum noise related to the quantum network construction through errors characterizing the qubit time evolution and gate application in both the first register and the flag qubit.

Acknowledgments. Authors are very grateful to Dr D.E. Houpa Danga for useful discussions and remarks. We are thankful to M.A.L. Kamga (PhD. Student Literature and Media, University of Bayreuth, Germany) for proofreading the work.

References

[1] R. Zhou, H. Wang, Q. Wu, and Y. Shi, Quantum Associative Neural Network with Nonlinear Search Algorithm, International Journal of Theoretical Physics 51(3), 705–723 March (2012).
[2] D. Rosenbaum, Binary superposed quantum decision diagrams, Quantum Information Processing 9(4), 463–496 August (2010).
[3] D. S. Abrams and S. Lloyd, Nonlinear Quantum Mechanics Implies Polynomial-Time Solution for NP-Complete and # P Problems, Phys. Rev. Lett. 81, 3992–3995 November (1998).
[4] D. Ventura and T. R. Martinez, Quantum Associative Memory, Inf. Sci. Inf. Comput. Sci. 124(1-4), 273–296 (2000).
[5] A. A. Êzhov, A. V. Nifanova, and D. Ventura, Quantum associative memory with distributed queries, Inf. Sci. Inf. Comput. Sci. 128(3-4), 271–293 October (2000).
[6] J.-P. Tchapet Njafa, S. G. Nana Engo, and P. Woafou, Quantum Associative Memory with Improved Distributed Queries, International Journal of Theoretical Physics 52(6), 1787–1801 June (2013).
[7] M. Czachor, Notes on nonlinear quantum algorithms, Acta Phys. Slov. 48, 157 February (1998).
[8] J. Polchinski, Weinberg’s Nonlinear Quantum Mechanics and the Einstein-Podolsky-Rosen Paradox, \textit{Physical Review Letters} Volume 66, Number 4 (1991).

[9] D. A. Meyer and T. G. Wong, Nonlinear quantum search using the Gross–Pitaevskii equation, \textit{New Journal of Physics} 15(6), 063014 (2013).

[10] P. J. Salas, Noise effect on Grover algorithm, \textit{The European Physical Journal D} 46(2), 365–373, February (2008).

[11] P. Gawron, J. Klamka, and R. Winiarczyk, Noise effects in the quantum search algorithm from the viewpoint of computational complexity, \textit{Applied Mathematics and Computer Science} 22(2), 493–499 (2012).

[12] M. A. Nielsen and I. L. Chuang, Quantum Computation and Quantum Information (Cambridge Series on Information and the Natural Sciences) (Cambridge University Press, New York, NY, USA, 10 edition, January (2000).

[13] C. P. Williams and S. H. Clearwater, Explorations in Quantum Computing (Springer-Verlag TELOS, Santa Clara, CA, USA, (1998).