



Journal of Applied Sciences

ISSN 1812-5654

science
alert

ANSI*net*
an open access publisher
<http://ansinet.com>

The Quantum Approach Leading from Evolutionary to Exhaustive Optimization

Massimo Panella and Giuseppe Martinelli

Department of Information Engineering, Electronics and Telecommunications,
University of Rome “La Sapienza”, Via Eudossiana 18, 00184 Rome, Italy

Abstract: What bio-inspired algorithms mimic are natural mechanisms governing the macroscopic world for optimizing actual performances that are of vital importance. Neural and neurofuzzy networks, genetic, swarm-intelligence and other evolutionary algorithms are well-known results of this imitation. A completely different situation characterizes the microscopic world governed by quantum mechanics. All the possible solutions exist simultaneously in superposition and the problem is to extract the optimal one. In this case, basic mechanisms of quantum mechanics, i.e., superposition and entanglement, are necessary to mimic nature. Following the latter approach, in this paper a quantum architecture was proposed for determining the maximum/minimum in a set of positive integers which is a basic problem related to optimization. The proposed architecture is based on a suitable nonlinear quantum operator and it solves the said problem by an exhaustive search. This was illustrated in detail in the case of a typical NP-complete problem.

Key words: Microscopic nature-inspired optimization, nonlinear quantum processing, quantum search for maximum/minimum, exhaustive quantum optimization

INTRODUCTION

The biological aspects considered to imitate natural mechanisms governing the macroscopic world span from the human brain to the behavior of swarms, as for instance that of ant colonies. Taking as inspiration the macroscopic world, the solutions are individuals belonging to a population. Their number is only a small fraction of the entire set of possible individuals. These individuals can be represented by points in a suitable space, the solution space. The bio-inspired mechanisms of optimization move the population in this space by specific interactions among the individuals for exploitation and exploration of all the possible alternatives (Goldberg, 1989; Eberhart *et al.*, 2001; Hassan *et al.*, 2005; Panella, 2011).

A completely different situation characterizes the microscopic world governed by quantum mechanics. All the possible solutions exist simultaneously in a superposition and the determination of the optimal one is a problem of extracting it from the superposition. Hence, to mimic nature in this case requires applying basic mechanisms of quantum mechanics, i.e. the tools that nature exploits for attaining optimal solutions: superposition and entanglement. As a consequence of this, the evolutionary strategy must be replaced by a different one, passing from the macroscopic to the

microscopic world, in such a way that the quantum approach can be considered as a microscopic nature-inspired strategy. A proof is given in the paper that the new strategy corresponds to an exhaustive search of the optimal solution.

It should be emphasized that quantum computing is presently not supported by sufficient technological achievements and there are serious difficulties of physical feasibility. Emerging technologies in this regard are those based on superconducting loops, cavity quantum electrodynamics-QED and, above all, ion traps, nuclear magnetic resonance, silicon spintronic and phosphorous-in-silicon systems (Gershenfeld and Chuang, 1997; Stick *et al.*, 2006; Das, 2011a, b). In spite of this inconvenience, quantum processing is partially imitated since it is widely believed that the exploitation of quantum principles is very useful for increasing the efficacy of the algorithms to be applied to information processing. Consequently, the quantum inspiration has been frequently followed for improving classical stochastic algorithms. There are very numerous contributions in the technical literature in this regards, labeled either with the correct label ‘quantum-inspired’ or simply with the label ‘quantum’. The resulting algorithms however do not fully exploit the inherent power of quantum processing. Two clear examples in this direction are respectively the quantum neural (QNN) and quantum neurofuzzy networks

(QNF) (Ventura, 1998; Narayanan and Menneer, 2000; Ezhov and Ventura, 2000; Gupta and Zia, 2001; Ricks and Ventura, 2003) as well as Quantum Evolutionary Algorithms (QEA) (Han and Kim, 2000, 2002; Giraldi *et al.*, 2004). This partial exploitation of quantum processing power is still present in the recent technical literature, as clearly evidenced by Abs da Cruz *et al.* (2007), Platel *et al.* (2009), Huo *et al.* (2010), Yanguang *et al.* (2010), Mani and Patvardhan (2010), Nie *et al.* (2010), Abs da Cruz *et al.* (2010), Lu and Juang (2011), Mukherjee *et al.* (2011) and Ibrahim *et al.* (2011, 2012).

However, also if quantum technology is still far from actual application, it is necessary to be ready to fully exploit its power. This remark motivates the development of the few contributions recently devoted to the fully exploitation of quantum computing (Panella and Martinelli, 2007, 2009, 2011; Fan *et al.*, 2008; Malossini *et al.*, 2008; Xiao *et al.*, 2010; Zhang, 2011; Dong *et al.*, 2012). Particularly interesting are the algorithms proposed in the case of QNN's and QNF's based on the use of both linear and nonlinear quantum operators. The use of the nonlinear operators is mandatory to an exhaustive optimization process. In fact, a new methodology is proposed in this paper in order to design a hardware architecture consisting of a suitable number of linear and nonlinear quantum gates which is able to solve the basic problems related to optimization (Goldberg, 1989; Panella, 2012).

QUANTUM ARCHITECTURES FOR THE EXHAUSTIVE OPTIMIZATION OF A PROBLEM

Exhaustive optimization of a problem is carried out by considering all its possible solutions and by determining the one(s) scoring the best performance. The number of possible solutions of a problem is however very large and hence, the consideration of all of them cannot be undertaken with a reasonable computational cost by applying the optimization procedures available outside the quantum world. Exhaustive optimization is instead feasible by pursuing the inherent properties of quantum mechanics where suitable quantum operators are involved, each corresponding to well-defined set of physical quantum gates. In the present study this possibility is proved by developing a suitable quantum hardware architecture; it consists of a set of linear and nonlinear quantum gates connected among them for achieving the said goal of exhaustive optimization. The structure of the proposed quantum architecture will be successively justified in the study.

It is important to notice that, in the case of a specific problem, there are several ancillary operations to be

necessarily undertaken in order to exploit the proposed architecture for the exhaustive optimization of the problem. These operations depend on the problem of interest and constitute a relevant portion of the entire computational procedure. No general rules can be devised in this regard because of the excessive diversity of situations arising in the field of information processing. In order to clarify this point, a specific problem is focused in the following by considering in this case the entire procedure.

Nonetheless, most of the optimization problems faced in the field of information processing can be eventually reduced to find the maximum (minimum) in a set of positive integers, each constituted by a suitable string of bits. These integers represent the performances obtained in relation with all the possible solutions of a problem of interest. Hence, the exhaustive optimization of the latter requires the determination of the maximum (minimum) of the said integers. Taking into account this remark, the following three points should be focused:

- The problem of interest must be tailored to the rules of quantum processing. Namely, the problem of interest must be formulated in terms of Boolean variables and Boolean functions by using suitable strings of bits. A particular attention should be addressed to the choice of the number of bits required for attaining an acceptable accuracy (Papageorgiou and Traub, 2005)
- All the solutions of the problem of interest along with their performances must be known simultaneously. This result is obtained by exploiting the superposition and entanglement properties of quantum processing. In this regard it is necessary:
 - To implement a Boolean function for determining the performance corresponding to a solution under the form of a positive integer
 - To map the previous Boolean function into the quantum field. This is always possible by means of linear quantum gates (Rieffel and Polak, 2000). A quantum operator is hence obtained for performing this task
 - To map the Boolean representation of solutions, i.e., M strings of a suitable number of bits each representing a solution of the problem of interest, in a quantum superposition $|S\rangle$ of a same number of qubit strings
 - To apply the previously obtained quantum operator to $|S\rangle$ so that a new superposition $|S, P\rangle$ is generated, being $|P\rangle$ the superposition entangled to $|S\rangle$ containing the performances scored by the M solutions present in $|S\rangle$. It is

important to point out that M is very large in order to explore exhaustively the solution space and that, if the M performances take on almost M distinct values, each performance should be represented by a string of λ qubits where λ is on the order of $\log_2 M$. However, this is not the practical case, since the performances do not need to be represented with high accuracy. Consequently, λ is usually much less than $\log_2 M$

- The optimal solution corresponds to the maximum (minimum) item in the superposition $|P\rangle$. Its determination must be performed at a reasonable cost. The latter condition rules out previous quantum algorithms available in the technical literature, as those suggested in study of Durr and Hoyer (1998), which is based on linear quantum operators only. They in fact require a computational cost on the order of $O(\sqrt{M})$. This cost is too large for undertaking an exhaustive search of the optimum. A possible quantum architecture for carrying out this job is successively described in the paper; it has a computational cost equal to $O(\lambda)$ where, as said, it is also possible to obtain $\lambda \ll \log_2 M$ without actually impairing the results available by the exhaustive search of all the solutions.

SOME BASIC QUANTUM GATES AND OPERATORS

For what previously introduced, this study focuses on the algorithms that allow obtaining the solution of a problem by a suited sequence of logical (Boolean) operators. The corresponding architectures of quantum circuits are considered as well, in particular the methodology for determining a suited topology by using well-known mappings of Boolean functions into the quantum field. This is confirmed by the proposed quantum network successively illustrated.

By the way, it is worth summarizing some notations, definitions and properties, as they will be used in the following of the paper. They are regarding to the processed quantum data and the linear and nonlinear quantum gates and operators. More details about basic concepts can be found, for instance, in the study by Rieffel and Polak (2000):

- **Data:** The data to be handled are strings of qubits (quantum bits) which play the same role as bits of the classical logic circuits. A qubit is a unit vector in a two dimensional complex vector space. Unlike classical bits, qubits are in superposition of $|0\rangle$ and $|1\rangle$

as $(a|0\rangle + b|1\rangle)$, where a and b are complex numbers such that $|a|^2 + |b|^2 = 1$. If the qubit is measured, the probability that the measured value is $|0\rangle$ is $|a|^2$ and, similarly, the probability to be $|1\rangle$ is $|b|^2$. A string of n qubits has a state space of 2^n dimensions with 2^n basis vectors. The extraction of a specific state from the superposition is carried out in a probabilistic manner through measurement. After the measurement, the original superposition is destroyed since the process of measurement changes the state of the superposition to that measured. Only one result is obtained and worse still one cannot even choose which result one gets

- **Linear quantum gates and operators:** They are the ordinary components of a quantum architecture and they implement unitary transformations which can be thought as being rotations in a complex vector space. There are several gates, most of them are obtained by mapping into the quantum field a valid Boolean function. If $f(x)$ is the Boolean function to be handled, whose output is defined by q bits and the input x by m bits, then the corresponding quantum operator U is as follows:

$$U: |x, 0_q\rangle = |x, f(x)\rangle \quad (1)$$

where, $|x\rangle, |f(x)\rangle$ is an entangled pair and $|0_q\rangle$ represents a string of q qubits all equal to $|0\rangle$. The result $|x, f(x)\rangle$ is characterized by a string of $(q+m)$ qubits

In the following the well-known Toffoli gate will be used; it implements into the quantum field the AND operator (\wedge):

$$T: |x, y, 0\rangle = |x, y, x \wedge y\rangle \quad (2)$$

The inputs are the 3 qubits $|x\rangle, |y\rangle, |0\rangle$ and the output is constituted by the first 2 qubits and a third qubit, entangled with them which is coincident with their AND. Because of the entanglement, the three outputs are strictly connected and cannot be used separately. The previous operator is also valid in the case $|y\rangle$ is a string of qubits; in this case the third output is also a string.

The unitary property of linear quantum processing implies that quantum states cannot be copied or cloned by linear quantum operators. This property is a formidable drawback of linear quantum processing, taking into account the probabilistic nature of the extraction of a component from a superposition. More measurements should be required for obtaining the component of interest. However, one cannot repeat the measurement since it destroys the superposition. The only possibility

for repeating a measurement is to use different exemplars of a same superposition, obtained independently of each other.

- **Nonlinear quantum operators:** They are the basis of algorithms for overcoming the difficulties connected with the ordinary linear quantum components (Panella and Martinelli, 2009). The second (nonlinear) algorithm proposed by Abrams and Lloyd (1998), denoted as ‘NAL algorithm’, will be used in the following. It regards a superposition of the type:

$$|s\rangle = \frac{1}{\sqrt{M}} \sum_{k=1}^M |c^{(k)}, \Phi^{(k)}\rangle \quad (3)$$

where, $|\Phi^{(k)}\rangle$ is a string of qubits, their number can be any integer greater than or equal to 1, $|c^{(k)}\rangle$ is a single qubit entangled with $|\Phi^{(k)}\rangle$ and M is the number of elements in the superposition.

The NAL algorithm can ascertain with certainty if in the superposition (3) all $|c^{(k)}\rangle$ are equal to $|0\rangle$ or there is present at least one of them equal to $|1\rangle$. Namely, after the application of the NAL algorithm, the resulting superposition in the two cases is equal either to:

$$\frac{1}{\sqrt{M}} \sum_k |0, \Phi^{(k)}\rangle$$

or to:

$$\frac{1}{\sqrt{M}} \sum_k |1, \Phi^{(k)}\rangle$$

respectively. Therefore, by measuring the first qubit either $|0\rangle$ or $|1\rangle$ is obtained with certainty.

The NAL algorithm requires, for undertaking this job in polynomial time, the iterative application of ordinary unitary gates and a sequence of nonlinear operations partly disentangling the state of the quantum processor. This is obtained by transformations of type:

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

The nonlinear operator is moreover feasible by the help of ordinary unitary operations and much simpler and more natural single qubit nonlinear operators. Improved justifications and other versions of the NAL algorithm are available in study of Czachor (1998a, b), where the unphysical effects associated with the original version are eliminated.

SYNTHESIS OF A QUANTUM ARCHITECTURE FOR DETERMINING THE MAXIMUM/MINIMUM IN A SET OF POSITIVE INTEGERS

The procedure proposed in the following pertains to the design of a quantum architecture for determining the maximum value in a set of M positive integers and it is based on the NAL algorithm. The integers are denoted as $\xi^{(k)}$, $k = 1 \dots M$, and represented by a string of n qubits as follows:

$$|\xi^{(k)}\rangle = |b_1^{(k)} b_2^{(k)} \dots b_n^{(k)}\rangle \quad (5)$$

where, $|b_1^{(k)}\rangle$ is the most significant qubit. The set of $|\xi^{(k)}\rangle$ is stored in a superposition $|\Omega\rangle$:

$$|\Omega\rangle = \frac{1}{\sqrt{M}} \sum_{k=1}^M |\xi^{(k)}\rangle \quad (6)$$

where, $|\Omega\rangle$ is usually managed in the steps of the specific algorithm that is used for solving the problem of interest. When the problem is the search of the maximum in a set of integers, then they can be preliminarily stored following, for instance, the procedure suggested by Trugenberger (2001).

The reference algorithm: The determination of the maximum value $|\xi^{(max)}\rangle = |b_1^{(max)} b_2^{(max)} \dots b_n^{(max)}\rangle$ can be carried out by using the NAL algorithm in the following procedure. Its computational cost is polynomial since it requires n applications of the NAL algorithm and n measurements:

- **Initialization:** Let $|S_1\rangle$ be a superposition such that $|S_1\rangle = |\Omega\rangle$ as in Eq. 6. It can be represented in a compact form using the following general notation:

$$|S_1\rangle = \frac{1}{\sqrt{M}} \sum_{k=1}^M |b_{1,1}^{(k)} b_{1,2}^{(k)} \dots b_{1,n}^{(k)}\rangle \quad (7)$$

where, $|b_{1,j}^{(k)}\rangle = |b_j^{(k)}\rangle$ for $j = 1 \dots n$.

The application of the NAL algorithm with the measurements applied in the next steps destroys the processed superposition. Since, this inconvenience happens in all the n steps of the procedure, it is necessary to start the procedure with n exemplars of $|S_1\rangle$, obtained preliminarily and independently because of the no cloning property; they will be denoted as $|S_1^{(h)}\rangle$, $h = 1 \dots n$.

- **Step p, p = 1...(n-2):** Apply the NAL algorithm to the first instance of $|S_p\rangle$ that is $|S_p^{(1)}\rangle$. Considering the

notations of Eq. 3, it is $|c^{(k)}\rangle = |b_{p,1}^{(k)}\rangle$ and $|\Phi^{(k)}\rangle = |b_{p,2}^{(k)}\rangle$. The choice of a single qubit for $|\Phi^{(k)}\rangle$, thus not considering the remaining qubits of the superposition, is suggested by the convenience of reducing the number of qubits of $|\Phi^{(k)}\rangle$ to a minimum without impairing the result. The NAL algorithm ascertains the existence of one of the following two cases for $|S_p^{(1)}\rangle$:

- The value of $|b_{p,1}^{(k)}\rangle$ in all the components of $|S_p^{(1)}\rangle$ is equal to $|0\rangle$ and hence, it results $|b_p^{(max)}\rangle = |0\rangle$
- There are some values of $|b_{p,1}^{(k)}\rangle$ equal to $|1\rangle$, then it results $|b_p^{(max)}\rangle = |1\rangle$

The superposition to be examined in the successive step of the procedure is different in the two previous cases. Namely, using the notation $|S_{p+1}\rangle$, it is defined as follows:

- **First case:** all the integers under investigation have $|b_{p,1}^{(k)}\rangle$ equal to $|0\rangle$. Hence, the determination of the maximum should be addressed to the successive most significant qubits. Since, the superposition $|S_p^{(1)}\rangle$ is destroyed by the previous measurement, it must be replaced by another exemplar of $|S_p\rangle$. For instance, it is considered as $|S_p^{(2)}\rangle$ and hence, the procedure continues by determining the maximum of the integers constituted by (n-p) qubits stored in $|S_{p+1}\rangle$. It will be represented in a compact form as follows:

$$|S_{p+1}\rangle = \frac{1}{\sqrt{M}} \sum_{k=1}^M |b_{p+1,1}^{(k)} b_{p+1,2}^{(k)} \dots b_{p+1,n-p}^{(k)}\rangle \quad (8)$$

where, $|b_{p+1,j}^{(k)}\rangle = |b_{p,j+1}^{(k)}\rangle$ for $j = 1 \dots (n-p)$.

- **Second case:** some of the integers under investigation have $|b_{p,1}^{(k)}\rangle$ equal to $|1\rangle$. Hence, the maximum value to be found is among them. Also in this case, the determination of the maximum in the next step is addressed to the successive most significant qubits of $|S_p^{(2)}\rangle$. However, the investigation should be limited only to the components of $|S_p^{(2)}\rangle$ having $|b_{p,1}^{(k)}\rangle = |1\rangle$. This objective is attained by setting to $|0\rangle$ the qubit $|b_{p,2}^{(k)}\rangle$ of the components of $|S_p^{(2)}\rangle$ having $|b_{p,1}^{(k)}\rangle$ equal to $|0\rangle$. This result is obtained by applying the Toffoli gate to the qubits $|b_{p,1}^{(k)}\rangle$ and $|b_{p,2}^{(k)}\rangle$ as in Eq. 2, i.e.:

$$T: |b_{p,1}^{(k)} b_{p,2}^{(k)} 0\rangle = |b_{p,1}^{(k)} b_{p,2}^{(k)} b_{p,1}^{(k)} \wedge b_{p,2}^{(k)}\rangle \quad (9)$$

By applying the Toffoli gate to such specific qubits of the whole superposition $|S_p^{(2)}\rangle$, the following one is obtained:

$$T(|S_p^{(2)}, 0\rangle) = \frac{1}{\sqrt{M}} \sum_{k=1}^M |b_{p+1,1}^{(k)} b_{p+1,2}^{(k)} \dots b_{p+1,n-p}^{(k)} w_{p+1}^{(k)}\rangle \quad (10)$$

where, $w_{p+1}^{(k)} = b_{p+1,1}^{(k)} \wedge b_{p+1,2}^{(k)}$. In this case, $|S_{p+1}\rangle$ will be represented in a compact form as follows, taking only the qubits to be used in the successive steps and using a suited permutation of the qubit order in Eq. 10:

$$|S_{p+1}\rangle = \frac{1}{\sqrt{M}} \sum_{k=1}^M |b_{p+1,1}^{(k)} b_{p+1,2}^{(k)} \dots b_{p+1,n-p}^{(k)}\rangle \quad (11)$$

where, $|b_{p+1,1}^{(k)}\rangle = |w_{p+1}^{(k)}\rangle$ and $|b_{p+1,j}^{(k)}\rangle = |b_{p,j+1}^{(k)}\rangle$ for $j = 2 \dots (n-p)$.

Finally, it is important to notice that the NAL algorithm will be applied similarly also in the successive n-p steps. Therefore, n-p superpositions $|S_{p+1}\rangle$ must be determined as illustrated in the previous operation. More precisely, they are obtained using n-p exemplars of $|S_p\rangle$ that is $|S_p^{(h+1)}\rangle$, $h = 1 \dots (n-p)$, and they will be denoted as $|S_{p+1}^{(h)}\rangle$.

- **Step n-1:** The same operations of the previous generic step are applied, considering in this case the specific value $p = n-1$. In particular Eq. 10 becomes:

$$T(|S_{n-1}^{(2)}, 0\rangle) = \frac{1}{\sqrt{M}} \sum_{k=1}^M |b_{n-1,1}^{(k)} b_{n-1,2}^{(k)} w_n^{(k)}\rangle \quad (12)$$

where, $w_n^{(k)} = b_{n-1,1}^{(k)} \wedge b_{n-1,2}^{(k)}$. However, in this case, the next superposition $|S_n\rangle$ will be still formed by two qubits instead of only one. More precisely:

$$|S_n\rangle = \frac{1}{\sqrt{M}} \sum_{k=1}^M |b_{n,1}^{(k)} b_{n,2}^{(k)}\rangle \quad (13)$$

where, $|b_{n,2}^{(k)}\rangle = |b_{n-1,1}^{(k)}\rangle$ and:

- $|b_{n,1}^{(k)}\rangle = |b_{n-1,2}^{(k)}\rangle$ in the first case, when $|b_{n-1}^{(max)}\rangle = |0\rangle$
- $|b_{n,1}^{(k)}\rangle = |w_n^{(k)}\rangle$ otherwise that is in the second case when $|b_{n-1}^{(max)}\rangle = |1\rangle$

At the end of this step, there is only one exemplar of $|S_n\rangle$.

- **Step n:** Apply the NAL algorithm to $|S_n\rangle$ considering, with the notations of Eq. 3, $|c^{(k)}\rangle = |b_{n,1}^{(k)}\rangle$ and $|\Phi^{(k)}\rangle = |b_{n,2}^{(k)}\rangle$. The two cases obtained by applying the NAL algorithm and then measuring $|b_{n,1}^{(k)}\rangle$ will determine the value of the nth qubit of the solution. Namely, in the first case it is $|b_n^{(max)}\rangle = |0\rangle$ while in the second case it is $|b_n^{(max)}\rangle = |1\rangle$

Determination of the quantum architecture in a toy case:
The methodology to determine the quantum architecture

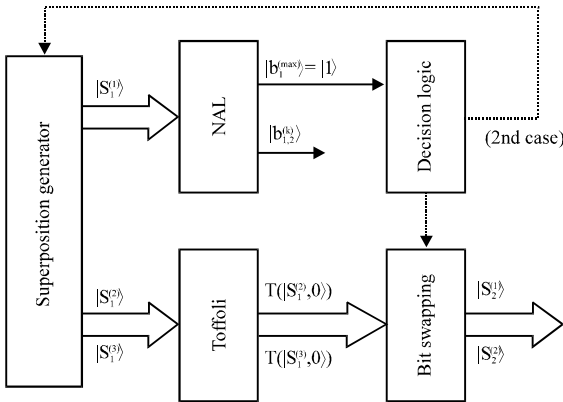


Fig. 1: Quantum architecture obtained for the toy case: the figure refers to its operation in the step 1, Wide arrows: A set of entangled qubits, Tiny arrows: Single disentangled qubits, Dashed lines: Signals of control logic

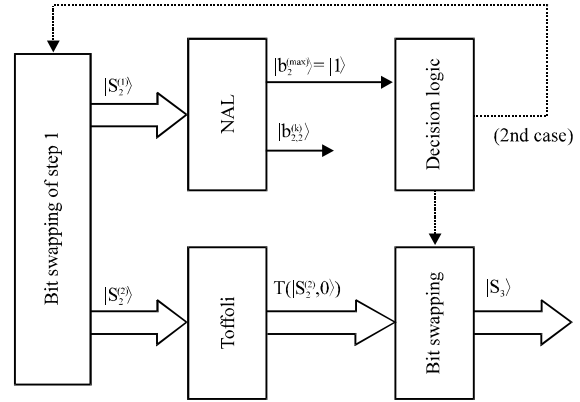


Fig. 2: Quantum architecture for the toy case: the figure refers to its operation in the step 2, Wide arrows: A set of entangled qubits, Tiny arrows: Single disentangled qubits, Dashed lines: Signals of control logic

able to find the maximum in a set of positive integers is illustrated by the following simple example. Let $\{4, 6, 2, 0\}$ be the set of integers under consideration. Hence, being $M = 4$ and $n = 3$, the superposition of Eq. 6 will be:

$$|\Omega\rangle = \frac{1}{2}|100\rangle + \frac{1}{2}|110\rangle + \frac{1}{2}|010\rangle + \frac{1}{2}|000\rangle$$

which will be also used for the initialization of superposition $|S_1\rangle$ as in Eq. 7. Three identical and independent superpositions, i.e., $|S_1^{(1)}\rangle$, $|S_1^{(2)}\rangle$ and $|S_1^{(3)}\rangle$ will be generated.

The operations involved in the step 1 are clearly illustrated in Fig. 1. By using a suited generator of superpositions, $|S_1^{(1)}\rangle$ will feed the quantum array implementing the NAL algorithm. It should be outlined that qubits $|b_{1,1}^{(k)}\rangle$, $|b_{1,2}^{(k)}\rangle$ and $|b_{1,3}^{(k)}\rangle$ are all entangled for any k th element, $k = 1 \dots 4$, of the superposition. Only the first two qubits will be used as inputs of the NAL gate; i.e., $|c^{(k)}\rangle = |b_{1,1}^{(k)}\rangle$ and $|\Phi^{(k)}\rangle = |b_{1,2}^{(k)}\rangle$. Since, there are some components of $|S_1^{(1)}\rangle$ whose the first and most significant qubit is $|1\rangle$, then $|b_1^{(max)}\rangle = |1\rangle$ and hence, the current situation is the “second case”. This result can be clearly stored and ascertained by a block that performs some branch instructions on the basis of the actual case. In fact, the two output qubits of the NAL gate are disentangled.

Successively, by applying the Toffoli gate to another copy $|S_1^{(2)}\rangle$ as specified in Eq. 10, the following superposition is obtained:

$$T(|S_1^{(2)}, 0\rangle) = \frac{1}{2}|1000\rangle + \frac{1}{2}|1101\rangle + \frac{1}{2}|0100\rangle + \frac{1}{2}|0000\rangle$$

Bearing in mind that this is the “second case”, a suited array will perform the bit swapping and selection defined in Eq. 11, in order to obtain the superposition $|S_2^{(1)}\rangle$ for the next step:

$$|S_2^{(1)}\rangle = \frac{1}{2}|00\rangle + \frac{1}{2}|10\rangle + \frac{1}{2}|00\rangle + \frac{1}{2}|00\rangle$$

More precisely, since in the next step 2 two copies of the latter superposition are necessary, i.e., $|S_2^{(1)}\rangle$ and $|S_2^{(2)}\rangle$, they will be obtained by repeating the latter branch twofold and by using $|S_1^{(2)}\rangle$ and $|S_1^{(3)}\rangle$, respectively.

In the step 2 the same operations are performed, as illustrated in Fig. 2, where the NAL array is fed by the superposition $|S_2^{(1)}\rangle$. Similarly to the previous step, there is one component having its first qubit equal to $|1\rangle$ and hence $|b_2^{(max)}\rangle = |1\rangle$. The application of the Toffoli gate to the other instance $|S_2^{(2)}\rangle$ of the superposition will yield by Eq. 12:

$$T(|S_2^{(2)}, 0\rangle) = \frac{1}{2}|000\rangle + \frac{1}{2}|100\rangle + \frac{1}{2}|000\rangle + \frac{1}{2}|000\rangle$$

Being this the penultimate step, the sole exemplar of the last superposition $|S_3\rangle$ will be still formed by two qubits instead of only one. More precisely, using Eq. 13:

$$|S_3\rangle = \frac{1}{2}|00\rangle + \frac{1}{2}|01\rangle + \frac{1}{2}|00\rangle + \frac{1}{2}|00\rangle$$

Finally, in step 3 the NAL algorithm is applied to $|S_3\rangle$ and so $|b_3^{(max)}\rangle = |0\rangle$ is obtained, since all the components

have their first qubit equal to $|0\rangle$. As a result, the greatest integer in the considered set is correctly identified to be 6 that is:

$$|\xi^{(max)}\rangle = |b_1^{(max)} b_2^{(max)} b_3^{(max)}\rangle = |110\rangle$$

CASE STUDY: DETERMINATION OF THE SHORTEST TOUR LENGTH IN THE TSP

The TSP regards the following NP-complete problem. A salesman must visit N cities. In order to optimize his job, the length of his tour should be as short as possible. The tour should include all the cities but only one time in order to be optimal. The distances between the cities are stored in a matrix D such that $D(i,j)$ is the distance from city i to city j . Matrix D is not necessarily symmetric and has the terms of its main diagonal equal to zero by convention. The present objective is the determination of the length L_{min} of the shortest tour.

Quantum representation of a solution: The preliminary operation to be carried out is the representation of solutions by means of strings of qubits and then to store all the solutions in a suitable superposition. It is not important to include in the superposition also strings that are insignificant, in the sense that they do not represent a valid solution. They will be easily disregarded by a suitable computation of the performance of interest. What is important is the simplicity of the representation.

Each city is represented by a string $|\eta\rangle$ of ρ qubits:

$$|\eta\rangle = |\eta_1 \eta_2 \dots \eta_\rho\rangle \tag{14}$$

where, ρ is greater than or equal to $\log_2 N$. In the case ρ is greater than $\log_2 N$, then there are strings that are not valid since they do not correspond to cities. They will be denoted as $|\sigma^{(m)}\rangle$, $m = 1 \dots \mu$, $\mu = (2^\rho - N)$.

All the solutions of the problem are stored in a superposition of strings whose length is equal to ρN qubits:

$$|\Gamma\rangle = \frac{1}{\sqrt{M}} \sum_{k_1, k_2, \dots, k_N} |\eta^{(k_1)} \eta^{(k_2)} \dots \eta^{(k_N)}\rangle \tag{15}$$

where, $M = 2^{\rho N}$; $|\eta^{(k_i)}\rangle$, $i = 1 \dots N$, is a string of ρ qubits that should represent the city visited by the salesman in the i th step of his tour; the sum is extended over any combination of qubits (i.e., M addends). The components of the superposition $|\Gamma\rangle$ are valid solutions only if the N strings $|\eta^{(k_i)}\rangle$ correspond to different cities and to valid representations. Hence, in the computation of the

performance of the solution this constraint must be taken into account introducing a suitable penalty in order to disregard the components that are not valid.

Computation of the tour length and determination of its minimum value: The length L of the tour is obtained by adding the distances of the successive cities visited by the salesman. The value so obtained is then modified for taking into account the validity of the tour: a penalty term D_0 is added to it when it is not valid. The introduction of this term is controlled by a suitable flag qubit.

When the components of $|\Gamma\rangle$ do not represent a valid solution, a qubit $|d\rangle$ must be introduced for marking them. As said, a solution is not valid either if two or more than two cities $|\eta^{(k_i)}\rangle$ coincide between them or if anyone of them does not represent a city.

The first situation is ascertained by the following Boolean function $g(\cdot)$:

$$g(\eta^{(i_1)}, \dots, \eta^{(i_N)}) = \begin{cases} 1, & \text{if } \eta^{(i_1)} = \eta^{(i_j)} \text{ for some couple } i, j \\ 0, & \text{if } \eta^{(i_1)} \neq \eta^{(i_j)} \text{ for all couples } i, j \end{cases} \tag{16}$$

The function $g(\cdot)$ can be implemented as follows:

$$\begin{cases} \alpha_h^{(i_1, i_2)} = \text{XOR}(\eta_h^{(i_1)}, \eta_h^{(i_2)}), h = 1 \dots \rho \\ \alpha_{k_i, k_j} = \text{OR}(\alpha_1^{(i_1, i_2)}, \alpha_2^{(i_1, i_2)}, \dots, \alpha_\rho^{(i_1, i_2)}) \\ \alpha_u = \text{AND}(\alpha_{k_u, k_{u+1}}, \alpha_{k_u, k_{u+2}}, \alpha_{k_u, k_{u+3}}, \dots), u = 1 \dots N - 1 \\ g(\eta^{(i_1)}, \dots, \eta^{(i_N)}) = \text{NAND}(\alpha_1, \alpha_2, \dots, \alpha_{N-1}) \end{cases} \tag{17}$$

The second situation is ascertained by the following Boolean function $q(\cdot)$:

$$q(\eta^{(i_1)}, \dots, \eta^{(i_N)}) = \begin{cases} 1, & \text{if } \eta^{(i_1)} = \sigma^{(m)} \text{ for some couple } k_i, m \\ 0, & \text{otherwise} \end{cases} \tag{18}$$

The function $q(\cdot)$ can be implemented as follows:

$$\begin{cases} \beta_h^{(i_1, m)} = \text{XOR}(\eta_h^{(i_1)}, \sigma_h^{(m)}), h = 1 \dots \rho \\ \beta_{k_i, m} = \text{OR}(\beta_1^{(i_1, m)}, \beta_2^{(i_1, m)}, \dots, \beta_\rho^{(i_1, m)}) \\ \beta_i = \text{AND}(\beta_{k_i, 1}, \beta_{k_i, 2}, \dots, \beta_{k_i, \mu}), i = 1 \dots N \\ q(\eta^{(i_1)}, \dots, \eta^{(i_N)}) = \text{NAND}(\beta_1, \beta_2, \dots, \beta_N) \end{cases} \tag{19}$$

The two situations can be taken into account simultaneously by the Boolean function $p(\cdot)$:

$$p(\eta^{(i_1)}, \dots, \eta^{(i_N)}) = \text{OR}[g(\eta^{(i_1)}, \dots, \eta^{(i_N)}), q(\eta^{(i_1)}, \dots, \eta^{(i_N)})] \tag{20}$$

The Boolean function $p(\cdot)$ is then converted into a quantum operator π which is applied to the superposition $|\Gamma\rangle$ so as to obtain the final superposition $|\Theta\rangle$ where the flag qubit $|d\rangle$ is present and entangled with $|\Gamma\rangle$:

$$\begin{cases} \pi : |\Gamma, 0\rangle = |\Gamma, d\rangle \\ |\Theta\rangle = |\Gamma, d\rangle = \frac{1}{\sqrt{M}} \sum_{k_1, \dots, k_N} |\eta^{(k_1)} \eta^{(k_2)} \dots \eta^{(k_N)} d^{(k_1 \dots k_N)}\rangle \\ d^{(k_1 \dots k_N)} = p(\eta^{(k_1)}, \dots, \eta^{(k_N)}) \end{cases} \quad (21)$$

The matrix D of distances between cities, each measured by δ bits assuming a suitable normalization and quantization of distances, is determined by a Boolean function $t(\cdot)$ whose output is defined by δ bits and the inputs are two cities, each defined by ρ bits:

$$t(\eta^{(i)}, \eta^{(j)}) = D(i, j), i = 1 \dots N, j = 1 \dots N \quad (22)$$

When ρ is greater than $\log_2 N$, there are strings $|\eta^{(k)}\rangle$ that do not represent cities and they are denoted as $|\sigma^{(m)}\rangle$. The value of $t(\cdot)$ is set to zero when any of the $|\eta^{(k)}\rangle$ of Eq. 22 coincides with some $|\sigma^{(m)}\rangle$. The value zero is arbitrarily chosen, since this case is successively disregarded by introducing a suitable penalty term. The Boolean function $t(\cdot)$ is mapped into a quantum operator Δ :

$$\Delta : |\eta^{(i)}, \eta^{(j)}, 0_s\rangle = |\eta^{(i)}, \eta^{(j)}, t(\eta^{(i)}, \eta^{(j)})\rangle \quad (23)$$

where, the RHS of Eq. 23 is an entangled pair representing the result characterized by a string of $(2\rho + \delta)$ qubits.

The length L of the tour is obtained by summing the distances between successive cities. It is represented by λ bits, included the penalty. The said distances are added each other, starting from the first one and following the procedure suggested by Vedral *et al.* (1996) which is based on the use of linear quantum gates only. With the notations of Vedral *et al.* (1996):

$$\begin{aligned} |Z_1\rangle &= |\Theta, t(\eta^{(k_1)}, \eta^{(k_1)})\rangle \\ |Z_2\rangle &= |\Theta, t(\eta^{(k_1)}, \eta^{(k_2)})\rangle \\ &|t(\eta^{(k_1)}, \eta^{(k_1)}), t(\eta^{(k_1)}, \eta^{(k_2)})\rangle \Rightarrow \\ &|t(\eta^{(k_1)}, \eta^{(k_1)}); t(\eta^{(k_1)}, \eta^{(k_1)}) + t(\eta^{(k_1)}, \eta^{(k_2)})\rangle \end{aligned} \quad (24a)$$

$$\begin{aligned} |Z_3\rangle &= |\Theta, t(\eta^{(k_2)}, \eta^{(k_3)})\rangle \\ &|t(\eta^{(k_2)}, \eta^{(k_3)}); t(\eta^{(k_1)}, \eta^{(k_1)}) + t(\eta^{(k_1)}, \eta^{(k_2)})\rangle \Rightarrow \\ &|t(\eta^{(k_2)}, \eta^{(k_3)}); t(\eta^{(k_1)}, \eta^{(k_1)}) + t(\eta^{(k_1)}, \eta^{(k_2)}) + t(\eta^{(k_2)}, \eta^{(k_3)})\rangle \end{aligned} \quad (24b)$$

$$\begin{aligned} |Z_N\rangle &= |\Theta, t(\eta^{(k_{N-1})}, \eta^{(k_N)})\rangle \\ &|t(\eta^{(k_{N-1})}, \eta^{(k_N)}); t(\eta^{(k_1)}, \eta^{(k_1)}) + \sum_{i=1}^{N-2} t(\eta^{(k_i)}, \eta^{(k_{i+1})})\rangle \Rightarrow \\ &|t(\eta^{(k_{N-1})}, \eta^{(k_N)}); t(\eta^{(k_1)}, \eta^{(k_1)}) + \sum_{i=1}^{N-1} t(\eta^{(k_i)}, \eta^{(k_{i+1})})\rangle \end{aligned} \quad (24c)$$

$$\begin{aligned} |Z_{\text{opt}}\rangle &= |\Theta, d \cdot D_0\rangle \\ &|d \cdot D_0; t(\eta^{(k_1)}, \eta^{(k_1)}) + \sum_{i=1}^{N-1} t(\eta^{(k_i)}, \eta^{(k_{i+1})})\rangle \Rightarrow \\ &|d \cdot D_0; t(\eta^{(k_1)}, \eta^{(k_1)}) + \sum_{i=1}^{N-1} t(\eta^{(k_i)}, \eta^{(k_{i+1})}) + d \cdot D_0\rangle = |d \cdot D_0; L\rangle \end{aligned} \quad (24d)$$

where, D_0 is the penalty term introduced in the case the component of the superposition $|\Gamma\rangle$ does not represent a valid solution (i.e., the corresponding flag qubit $|d\rangle$ is $|1\rangle$).

The choice of D_0 and λ can be reasonably based on the maximum entry d_{max} of matrix D. Since, the length of the tour is less than Nd_{max} and $d_{\text{max}} < 2^\delta$, λ can be chosen as the smallest integer greater than or equal to $\delta + \log_2 N$. Moreover, D_0 should be chosen equal to Nd_{max} in order to cause the overflow of the previous sum when the solution is not valid. The final superposition is then:

$$|\Psi\rangle = |\Theta, L\rangle = |\Gamma, d, L\rangle = \frac{1}{\sqrt{M}} \sum_{i=1}^M |\Gamma_i, d_i, L_i\rangle \quad (25)$$

The minimum length L_{min} of the tour is determined by applying the quantum architecture obtained as previously illustrated (considering the search of a minimum rather than a maximum) with reference to the superposition of Eq. 25 and in particular, to the strings $|L_i\rangle$.

Computational cost of the exhaustive optimization: The complexity of the TSP is related to the number N of cities to be visited. For this reason, the dependence on N of the computational cost of the exhaustive optimization will be investigated in the following, either applying the previous quantum architecture or when a direct procedure based on the NAL algorithm is followed.

The computational cost in the first case is $O(\lambda)$, since λ is the number of applications of the NAL algorithm and of the quantum measurements as well. As said, λ is conveniently chosen on the order of $\delta + \log_2 N$. Reminding the discussion at the beginning of this paper, a coarse resolution (i.e., relatively few δ bits) for representing the distances between cities is necessary to reduce and control the overall computational complexity. Usually this does not prevent the minimum length of the tour from being represented among all the considered solutions of the problem. Hence, the computational complexity can be considered on the order of $O(\log_2 N)$.

In the second case, by following for instance the procedure described in the Appendix and using a similar notation, it is necessary firstly to derive a superposition $|\Omega\rangle$ constituted by all and only all the M' valid solutions of the TSP, where $M' = N!$. By directly applying the NAL

algorithm, the resulting computational cost will be on the order of $O(\log_2(N!))$. In this case, in order to reduce the complexity, the exhaustive optimization must be abandoned by shrinking the number of explored solutions without considering if the remaining ones may all correspond to poor results (high lengths of the tour).

Finally, it is important to point out that M' is much lower than the number M of elements in the superposition $|\Gamma\rangle$ reported in Eq. 15. In spite of this inconvenience, the dependence on N of the computational cost attainable by the architecture proposed in this paper is much smaller than the one associated with the direct application of the NAL algorithm to $|\Omega\rangle$. In fact, the former can be considered as $O(\log_2 N)$ while the latter is $O(\log_2(N!))$; for instance, in the case $N = 300$ it is $\log_2 N = 8.2$ and $\log_2(N!) = 2041$, respectively.

DISCUSSION

Exhaustive search is obtained efficiently in this paper exploiting the properties of quantum processing and using a suitable interconnection of linear and nonlinear quantum gates. In this regard, actual problems can be reduced to the determination of the maximum/minimum in a set of integers by using the strict relation that exists between Boolean algebra and quantum processing. In both cases, in fact, strings of binary quantities (bits in the Boolean case and qubits in quantum processing) are needed to represent what is of interest. Moreover, valid Boolean functions can be converted into quantum linear operators. What is completely different in the two cases is the possibility offered by the quantum processing to handle simultaneously all the possible alternatives or solutions of a problem. The difficult step of extracting the optimal solution from the superposition is solved by using a suitable quantum architecture constituted by linear and nonlinear quantum operators.

Nonetheless, it should be noted that it is not known to be true and is generally suspected to be false that quantum computers can solve NP-complete problems in polynomial time (Bernstein and Vazirani, 1997). In fact, the assumption of nonlinearity in the quantum effects is very important in this paper and so the applicability of the proposed approach is strictly dependent on the feasibility to implement the designed quantum gates by using the forthcoming technologies.

CONCLUSIONS

A significant contribution to the imitation of natural mechanisms of optimization is represented by evolutionary algorithms. They rely on the efficacy of these natural mechanisms. What is imitated belongs to the

macroscopic world while nature operates with the same efficacy at the microscopic level as well, where the governing laws are those of quantum mechanics. Hence, it is quite reasonable to investigate at this level the possibility of obtaining efficient optimization algorithms.

In the present study a procedure is proposed for synthesizing a quantum hardware architecture able to solve with a reasonable computational cost the basic problems of operational research and optimization, in particular the determination of the maximum/minimum in a set of integers. By means of this architecture it is possible to pursue an exhaustive search of the optimal solution and to control the overall computational cost without the necessity for iterative time-consuming procedures, as clearly proved by the example illustrated in this paper with relation to the well-known TSP problem.

APPENDIX

Direct application of the NAL algorithm to find the maximum in a set of positive integers: The direct application of the NAL algorithm allows the determination of the maximum in a string of positive integers. This is based on the use of specific Boolean functions successively transformed into linear quantum operators. Each of them must be elaborated by eventually arriving to a suitable quantum architecture including the said linear operator. The latter is usually much more complex than the simple Toffoli gate that is used in the architecture proposed in this paper, as clearly proved by considering the function $f_c(\cdot)$ required in the procedure described in the following.

Using the same notations previously introduced in the study, first the maximum in the set of M positive integers $\xi^{(k)}$, $k = 1 \dots M$, is found. To do this, the following Boolean function is constructed:

$$f_c(x) = \begin{cases} 0, & \text{if } x < c \\ 1, & \text{if } x \geq c \end{cases} \quad (26)$$

Then the problem is mapped into the quantum field and the Abrams-Lloyd algorithm is applied to the composed function $f_c(|\Omega\rangle)$, where $|\Omega\rangle$ is the quantum superposition of the previous integers. If the maximum is greater than or equal to c then Abrams-Lloyd will return YES, otherwise it will return NO. If the result was YES, then double c and try again. If the result was NO, then halve c and try again. In this manner the maximum $\xi^{(max)}$ is determined by a computational cost on the order of $\log_2(\xi^{(max)})$.

Once $\xi^{(max)}$ is known, the value of c is set to $c = \xi^{(max)}$ and then $f_c(|\Omega\rangle) = 1$ only at the maximum. To find this maximum, the method of Abrams-Lloyd is used to search

only on the domain $\{1, 2, \dots, M/2\}$. Depending on whether the answer is YES or NO, it is known whether the maximum occurs in the first or second half of the range $\{1, 2, \dots, M\}$. Then, the first half of that part can be searched and so on, each time narrowing down the potential location of the maximum by a factor of two. After $\log_2(M)$ repetitions the maximum is obtained and hence, the whole procedure to find the maximum has a $O[\log_2(\zeta^{(m\infty)}) + \log_2(M)]$ complexity.

REFERENCES

- Abrams, D. and S. Lloyd, 1998. Nonlinear quantum mechanics implies polynomial-time solution for NP-complete and #P problems. *Phys. Rev. Lett.*, 81: 3992-3995.
- Abs da Cruz, A.V., M.M.B.R. Vellasco and M.A.C. Pacheco, 2007. Quantum-inspired evolutionary algorithm for numerical optimization. *Hybrid Evol. Algorithms*, 75: 19-37.
- Abs da Cruz, A.V., M.M.B.R. Vellasco and M.A.C. Pacheco, 2010. Quantum-inspired evolutionary algorithms applied to numerical optimization problems. *Proceedings of the IEEE Congress on Evolutionary Computation*, July 18-23, 2010, Barcelona, Spain, pp: 1-6.
- Bernstein, E. and U. Vazirani, 1997. Quantum complexity theory. *SIAM J. Comput.*, 26: 1411-1473.
- Czachor, M., 1998a. Local modification of the abrams-lloyd nonlinear algorithm. <http://arxiv.org/pdf/quant-ph/9803019.pdf>
- Czachor, M., 1998b. Notes on nonlinear quantum algorithms. <http://arxiv.org/pdf/quant-ph/9802051v2.pdf>
- Das, S.R., 2011a. A crowd of quantum entanglements. *IEEE Spectr.*, 48: 18-18.
- Das, S.R., 2011b. A silicon spintronic memory that lasts. Stores spin for minutes instead of microseconds. *IEEE Spectrum*. <http://qa.spectrum.ieee.org/semiconductors/memory/a-silicon-spintronic-memory-that-lasts>
- Dong, D., C. Chen, J. Chu and T.J. Tarn, 2012. Robust quantum-inspired reinforcement learning for robot navigation. *IEEE Trans. Mechatron.*, 17: 86-97.
- Durr, C. and P. Hoyer, 1998. A quantum algorithm for finding the minimum. *Proceedings of 30th Annual ACM Symposium on Theory of Computing*, May 24-26, 1998, Dallas, TX., USA.
- Eberhart, R.C., Y. Shi and J. Kennedy, 2001. *Swarm Intelligence*. Morgan Kaufmann, San Francisco, CA., USA., ISBN: 9781558605954, Pages: 512.
- Ezhov, A.A. and D. Ventura, 2000. Quantum Neural Networks. In: *Future Directions for Intelligent Systems and Information Sciences*, Kasabov, N. (Eds.). Springer-Verlag, Berlin, Germany, pp: 213-234.
- Fan, K., A. Brabazon, C. O'Sullivan and M. O'Neill, 2008. Quantum-inspired evolutionary algorithms for financial data analysis. *Appl. Evol. Comput.*, 4974: 133-143.
- Gershenfeld, N.A. and I.L. Chuang, 1997. Bulk spin-resonance quantum computation. *Sci.*, 275: 350-356.
- Giraldi, G.A., R. Portugal and R.N. Thess, 2004. Genetic algorithms and quantum computation. *Comput. Sci. Neural Evol. Comput.*, <http://arxiv.org/pdf/cs/0403003.pdf>
- Goldberg, D.E., 1989. *Genetic Algorithms in Search Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
- Gupta, S. and R.K.P. Zia, 2001. Quantum neural networks. *J. Comput. Syst. Sci.*, 63: 355-383.
- Han, K. and J. H. Kim, 2000. Genetic quantum algorithm and its application to combinatorial optimization problems. *Proc. Congr. Evol. Comput.*, 2: 1354-1360.
- Han, K.H. and J.H. Kim, 2002. Quantum-inspired evolutionary algorithm for a class of combinatorial optimization. *IEEE Trans. Evol. Comput.*, 6: 580-593.
- Hassan, R., B. Cohanin, O. de Weck and G. Venter, 2005. A comparison of particle swarm optimization and the genetic algorithm. *Proceedings of the 46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, April 18-21, 2005, Austin, TX, USA., pp: 1-13.
- Huo, H.H., V. Stojkovic and Q.L. Xie, 2010. A quantum-inspired genetic algorithm based on probabilistic coding for multiple sequence alignment. *J. Bioinf. Comput. Biol.*, 8: 59-75.
- Ibrahim, A.A., A. Mohamed, H. Shareef and S.P. Ghoshal, 2011. An effective power quality monitor placement method utilizing quantum-inspired particle swarm optimization. *Proceedings of the 2011 International Conference on Electrical Engineering and Informatics*, July 17-19, 2011, Bandung, Indonesia, pp: 1-6.
- Ibrahim, A.A., A. Mohamed and H. Shareef, 2012. A novel quantum-inspired binary gravitational search algorithm in obtaining optimal power quality monitor placement. *J. Applied Sci.*, 12: 822-830.
- Lu, T.C. and J.C. Juang, 2011. Quantum-inspired space search algorithm (QSSA) for global numerical optimization. *Applied Math Comput.*, 218: 2516-2532.
- Malossini, A., E. Blanzieri and T. Calarco, 2008. Quantum genetic optimization. *IEEE Trans. Evol. Comput.*, 12: 231-241.

- Mani, A. and C. Patvardhan, 2010. An adaptive quantum evolutionary algorithm for engineering optimization problems. *Int. J. Comput. Appl.*, 1: 43-48.
- Mukherjee, S.S., R. Chowdhury and S. Bhattacharyya, 2011. Image restoration using a multilayered quantum back propagation neural network. *Proceedings of 2011 International Conference on Computational Intelligence and Communication Systems*, October 7-9, 2011, Gwalior, India, pp: 426-430.
- Narayanan, A. and T. Menneer, 2000. Quantum artificial neural network architectures and components. *Inform. Sci.*, 128: 231-255.
- Nie, R., X. Xu and J. Yue, 2010. A novel quantum-inspired particle swarm algorithm and its application. *Proceedings of the 6th International Conference on Natural Computation*, Volume 5, August 10-12, 2010, Yantai, Shandong, China, pp: 2556-2560.
- Panella, M. and G. Martinelli, 2007. Binary neuro-fuzzy classifiers trained by nonlinear quantum circuits. *Appl. Fuzzy Sets Theory*, 4578: 237-244.
- Panella, M. and G. Martinelli, 2009. Neurofuzzy networks with nonlinear quantum learning. *IEEE Trans. Fuzzy Syst.*, 17: 698-710.
- Panella, M. and G. Martinelli, 2011. Neural networks with quantum architecture and quantum learning. *Int. J. Circuit Theory Appl.*, 39: 61-77.
- Panella, M., 2011. Advances in biological time series prediction by neural networks. *Biomed. Signal Process. Control*, 6: 112-120.
- Panella, M., 2012. A hierarchical procedure for the synthesis of ANFIS networks. *Adv. Fuzzy Syst.*
- Papageorgiou, A. and J.F. Traub, 2005. Qubit complexity of continuous problems. <http://arxiv.org/pdf/quant-ph/0512082v1.pdf>
- Platel, M.D., S. Schliebs and N. Kasabov, 2009. Quantum-inspired evolutionary algorithm: A multimodel EDA. *IEEE Trans. Evol. Comput.*, 13: 1218-1232.
- Ricks B. and D. Ventura, 2003. Training a quantum neural network. *Neural Inform. Process. Syst.*, 16: 1019-1026.
- Rieffel, E. and W. Polak, 2000. An introduction to quantum computing for non-physicists. *ACM Comput. Surv.*, 32: 300-335.
- Stick, D., W.K. Hensinger, S. Olmschenk, M.J. Madsen, K. Schwab and C. Monroe, 2006. Ion trap in a semiconductor chip. *Nat. Phys.*, 2: 36-39.
- Trugenberger, C.A., 2001. Probabilistic quantum memories. *Phys. Rev. Lett.*, Vol. 87.
- Vedral, V., A. Barenco and A. Ekert, 1996. Quantum networks for elementary arithmetic operations. *Phys. Rev. A*, 54: 147-153.
- Ventura, D., 1998. Quantum and evolutionary approaches to computational learning. Ph.D. Thesis, Brigham Young University, Department of Computer Science.
- Xiao, J., Y.P. Yan, J. Zhang and Y. Tang, 2010. A quantum-inspired genetic algorithm for k-means clustering. *Expert Syst. Appl.*, 37: 4966-4973.
- Yanguang, C. and C. Hao, 2010. A hybrid chaotic quantum evolutionary algorithm. *Proceedings of the IEEE International Conference on Intelligent Computing and Intelligent Systems*, October 29-31, 2010 Xiamen, Fujian, China, pp: 771-776.
- Zhang, G., 2011. Quantum-inspired evolutionary algorithms: A survey and empirical study. *J. Heuristics*, 17: 303-351.