Int J Theor Phys (2008) 47: 2181–2194
DOI 10.1007/s10773-008-9698-x

The Mechanism of Quantum Computation

Giuseppe Castagnoli

Received: 31 January 2008 / Accepted: 27 February 2008 / Published online: 8 March 2008

Abstract I provide an alternative way of seeing quantum computation. First, I describe an idealized classical problem solving machine whose coordinates are submitted to a nonfunctional relation representing all the problem constraints; moving an input part, reversibly and nondeterministically produces a solution through a many body interaction. The machine can be considered the many body generalization of another perfect machine, the bouncing ball model of reversible computation. The mathematical description of the machine's motion, as it is, is applicable to quantum problem solving, an extension of the quantum algorithms that comprises the physical representation of the interdependence between the problem and the solution. The configuration space of the classical machine is replaced by the phase space of the quantum machine. The relation between the coordinates of the machine parts now applies to the populations of the reduced density operators of the parts of the computer register throughout state vector reduction. Thus, reduction produces the solution of the problem under a nonfunctional relation representing the problem-solution interdependence. At the light of this finding, the quantum speed up turns out to be "precognition" of the solution, namely the reduction of the initial ignorance of the solution due to backdating, to before running the algorithm, a part of the state vector reduction on the solution (a time-symmetric part in the case of unstructured problems); as such, it is bounded by state vector reduction through an entropic inequality. The computation mechanism under discussion might also explain the wholeness appearing in the introspective analysis of perception.

Keywords Quantum computation · Quantum measurement · Speed up · Nondeterminism · Reversibility

1 Introduction

I have been invited to write about the early history of quantum computation as seen from my special point of view, which I think hinges on the idea that computation is a reproduction of

G. Castagnoli (✉)
Via San Bernardo 9A, Pieve Ligure Alta, Genoa 16030, Italy
e-mail: giuseppe.castagnoli@gmail.com

the workings of the mind. This might reflect my professional experience in artificial intelligence, where the introspective analysis of our way of perceiving a pattern or associating concepts is essential for the development of a pattern recognition algorithm or a cognitive network. I got to quantum computation by shifting the interplay between introspection and computation from the logical to the physical level. Here below I provide my line of thinking. It starts from the wholeness (or unity) appearing in the introspective analysis of perception, to get to a fundamental computation mechanism that, in my judgement, stands at the basis of quantum computation.

For wholeness of perception, I mean the following. For example, in this moment, I see the room in which I am working, an armchair, the window, the garden, and the Mediterranean Sea on the background. In my visual perception, besides some aspects that are addressed by artificial intelligence, like the recognition of patterns, there is another thing that should be addressed by a physical information theory, the both obvious and striking fact that I see so many things together at the same time. What I see is close to a digital picture whose specification would require a significant amount of information. And apparently we can perceive a significant amount of information simultaneously all together, in the so called "present". Another example is our capability of grasping the solution of a problem. Reasonably, when we grasp the solution, we should take into account at the same time the statement of the problem, the solution, and the logical connection in between.

The idea that many things interact, are processed, at the same time can be formalized by resorting to a notion of the Gestalt theory. Wholeness (in fact, Gestalt) is simultaneous dependence between quantitative variables (e.g. [28]). Applied to a physical situation, this definition becomes: the wholeness of a physical situation implies that there is simultaneous dependence between all the quantitative variables describing it. An example is second Newton's law "force equal mass times acceleration" in the case of a point mass. It establishes a simultaneous dependence between the three quantitative variables that describe the physical situation. For reasons that will become clear, it is important to note that this dependence should be considered objectively perfect. If we see it as a mechanism whose degrees of freedom are the variables related by the law, this mechanism should be perfectly accurate, rigid, and reversible—it is not the case that Newton's law gets deformed because of flexibility or jams because of friction or irregularities. Another important feature of the simultaneous dependences that we find in Nature is that they can be nonfunctional, which is also the case of Newton's law. The change of one variable is correlated with an identical change of the product or ratio of the other two variables, but does not determine their individual changes. Correspondingly Newton's law can host nondeterminism, in the form of the many body problem. As we will see, a perfect nonfunctional simultaneous dependence enables a nondeterministic form of computation isomorphic with many body interaction. Of this form of computation one can say that any amount of information is processed at the same time.

We examine the relation between simultaneous dependence and computation. We can start by checking that there is no simultaneous dependence in classical computation. Let us consider the idealized bouncing ball model of reversible computation [22]. The variables at stake are ball positions and momenta. Outside collisions, there is no simultaneous dependence between the variables of different balls, which are independent of each other. In the instant of (idealized) collision, there is simultaneous dependence between the variables of the colliding balls, but this is limited to ball pairs (there can be several collisions at the same time, but involving independent ball pairs, with no simultaneous dependence between the variables of different pairs). The simultaneous collision between more than two balls is avoided since it would introduce the many body problem, namely an undetermined dynamics. No matter what computation size is, simultaneous dependence remains confined to ball pairs, it does not scale.

By assuming a perfect, nonfunctional simultaneous dependence between all computational variables, one can devise an idealized classical machine that nondeterministically produces the solution of a system of Boolean equations under the simultaneous influence of all equations. Boolean variables are mapped on real variables—the coordinates of the machine parts. These are related by a perfect nonfunctional simultaneous dependence representing all the Boolean equations. Moving the "input" part of the machine instantly produces a solution through a many body interaction (Sect. 2). This form of computation is essentially different from classical computation, namely from the causal propagation of the input into the output. For example, a many body interaction of this kind can produce two inputs such that their product is a preassigned output; if this were an input-output propagation (which is not), one should say that the input is produced with precognition of the output. I call the nondeterministic production of the solution under a nonfunctional simultaneous dependence representing problem-solution interdependence, *simultaneous computation*.

Noticeably, the mathematical description of the motion of the idealized classical machine can represent a realistic quantum computation. We should replace the configuration space of the classical machine by the phase space of the quantum machine. The nonfunctional simultaneous dependence between the coordinates of the machine parts now applies to the populations of the reduced density operators of the parts of the computer register before and after state vector reduction. Reduction conserves this dependence, it occurs under it. The populations before and after reduction are analogous to the coordinates of the classical machine, both perform a computation by changing under a nonfunctional simultaneous dependence (representing state vector reduction in the quantum case, a perfect many body interaction in the classical case, problem-solution interdependence in either case). That infinite classical precision can be dispensed for in the quantum framework was already noted by Finkelstein [21].

It should be noted that the nonfunctional simultaneous dependence between the populations before and after reduction, functionally extends to all the amplitudes of the quantum process throughout preparation, unitary development, and measurement. Under the extended dependence, state vector reduction changes the forward development into the backward development, i.e. the same unitary transformation but ending with the outcome of measurement (Sect. 3).

Simultaneous computation is almost evident in the algorithms of Simon [32] and Shor [31]. On the contrary, it is completely hidden in Deutsch's [16] and Grover's [23] algorithms, which yield their speed ups through unitary evolutions (disregarding the probability of error). This can be ascribed to the fact that these algorithms physically represent only the procedure that leads to the solution, and are oblivious of the representation of the problem. However, it suffices to add the physical representation of the problem to see simultaneous computation (that state vector reduction produces the solution through a nonfunctional simultaneous dependence representing problem-solution interdependence). At the light of this finding, the speed up becomes "precognition of the solution", namely the reduction of the initial ignorance of the solution due to backdating, to before running the algorithm, a time-symmetric part of the state vector reduction on the solution. This holds in the case of Deutsch's and Grover's algorithms, which address unstructured problems. More in general, the speed up is bounded by state vector reduction through an entropic inequality (Sect. 4).

From the one side, the notion of simultaneous computation shows the existence of a fundamental—reversible and nondeterministic—computation mechanism implicit in the quantum algorithms and explains the speed up. From the other, it provides a possible meaning to the common sense statement that we can perceive a lot of information all together and simultaneously in the so called "present". A quantum state can hold any amount of

information, which is simultaneously processed by the sequence: preparation, unitary transformation, and measurement. The entire processing is simultaneous in the sense that there is simultaneous dependence between all amplitudes at any pair of times along the process. I conjecture that the time interval spanned by backdated state vector reduction, characterized by simultaneous dependence across time, corresponds to the introspective notion of "present".

The observation that, in visual perception, we take into account many things at the same time acquires a literal meaning. Taking into account many things at the same time is exactly what many body interaction—nondeterministic simultaneous computation—does. In this perspective, the wholeness/unity/Gestalt of conscious perception is not the epiphenomenon of an independent deterministic activity but an essential feature of the present form of nondeterministic computation.

It might be interesting to compare these notions with the idealized bouncing ball model of classical computation, where simultaneous dependence is limited to the variables of ball pairs in the instant of their collision. The amount of information simultaneously processed cannot scale up, not to bring in many ball collisions and an uncontrollable form of nondeterminism. The deterministic, two-body character of classical computation prevents taking into account many (so to speak, more than two) things at the same time, or (in present assumptions) hosting perception either.

The present identification between the notions of simultaneous dependence, physical law, and perception has a precedent in Plato's notion of Form (the Greek word Eidos translates into Form, Idea, or Vision): "Ideas are objective perfections that exist in themselves and for themselves, at the same time they are the cause of natural phenomena, they keep phenomena bound together and constitute their unity." In this quotation from Phaedo, the Ideas of our mind are clearly identified with physical laws; as well known, Platonic Ideas are also perfect mathematical objects. The usual Platonist interpretation of this ambivalence is that the mind can access an autonomous and objective world of perfect mathematical ideas. A more physical interpretation is the other way around, the ideas in our head—our perceptions—could be represented as physical laws, namely as objectively perfect simultaneous dependences.

The present idea that "grasping the solution of a problem" implies a simultaneous dependence representing the problem-solution interdependence, is parallel to another statement of the theory of Forms: "To know the Form of X is to understand the nature of X; so the philosopher who, for example, grasps the Form of justice, knows not merely what acts are just, but also why they are just."

The connection between the world of Ideas, the physical world, and the mental world is also central to Penrose and Hameroff search for a quantum theory of consciousness—the very existence of consciousness is ascribed to our capability of accessing the Platonic world of perfect mathematical Forms [25, 29]. See also [33].

The following Sects. 2 through 4 provide a detailed development of the model of simultaneous computation. In Sect. 5, I try to position the present approach within the development of the notion of quantum computation. Section 6 provides an account of the Turin workshops and a commented bibliography of my publications on the subject of quantum computation.

2 Simultaneous Computation in an Idealized Classical Framework

By assuming a perfect nonfunctional simultaneous dependence between all computational variables, one can devise an idealized classical machine that—thanks to a many body interaction—nondeterministically produces the solution of a system of Boolean equations

under the simultaneous influence of all equations. We can start with the simple problem of finding the solutions of the equation $y = \overline{x}$, i.e. $x = 0$, $y = 1$ and $x = 1$, $y = 0$. Let X, Y, Q be real non-negative variables. The Boolean problem can be transformed into the problem of finding the solutions, for $Q > 0$, of the simultaneous equations

$$\frac{X}{Q} + \frac{Y}{Q} = 1, \tag{1}$$

$$\left(\frac{X}{Q}\right)^2 + \left(\frac{Y}{Q}\right)^2 = 1. \tag{2}$$

$Q = 0$ implies $X = Y = 0$, while $\frac{X}{Q}$ and $\frac{Y}{Q}$ are individually undetermined. With $Q > 0$, $\frac{X}{Q} \equiv x$ and $\frac{Y}{Q} \equiv y$: one can see that $X = 0$, $Y = Q > 0$ corresponds to the Boolean values $x = 0$, $y = 1$ and $X = Q > 0$, $Y = 0$ to $x = 1$, $y = 0$.

In this real variable representation, the solutions can be computed by a many body interaction, as follows. Equation (1) can be represented by an idealized hydraulic circuit where Q is the coordinate of a piston feeding in parallel (through an incompressible fluid) two pistons of even section and mass, and coordinates respectively X and Y. Equation (2) is represented by a differential mechanism with non-linear (parabolic) cams applying to pistons X, Y, and Q (I use the same symbol to denote the piston and its coordinate). The initial configuration of the machine is $X = Y = Q = 0$; it can be argued that any movement of piston Q from $Q = 0$ to $Q > 0$ instantly produces a solution in a nondeterministic way. This motion could be obtained by applying a force to piston Q, then there would be no reason that either X or Y (in a mutually exclusive way) move with Q, as either movement offers zero static resistance to the force (there is only the inertia of the pistons). This reversible, nondeterministic many body interaction should be postulated in the present idealized classical framework, in the quantum framework it becomes a representation of measurement.

Unlike deterministic reversible processes, the present process is not invertible—in general one cannot go back and forth along the same process. For example, we can think of connecting the input piston to an ideal spring charged when $Q = 0$. On the one side, there would be oscillations without dissipation. On the other, at each oscillation, the movement of the input piston from $Q = 0$ to $Q > 0$ would randomly drag either X or Y in a mutually exclusive way.

This idealized computation mechanism can solve any system of Boolean equations, namely of N NAND equations $x_{i,3} = \mathrm{NAND}(x_{i,1}, x_{i,2})$, with $i = 1, \ldots, N$ and $x_{i,j} = x_{h,k}$ for some assignments of i, j, h, k. The hydraulic circuit becomes the series of an input branch/piston Q and N quadruples of parallel branches/pistons $X_{i,j}$, $j = 1, \ldots, 4$. The four branches/pistons of each quadruple are labeled by the Boolean values that satisfy the corresponding NAND equation. For example, branches/pistons $X_{i,1}$; $X_{i,2}$; $X_{i,3}$; $X_{i,4}$ are labeled by, respectively, $x_{i,1} = 0$, $x_{i,2} = 0$, $x_{i,3} = 1$; $x_{i,1} = 0$, $x_{i,2} = 1$, $x_{i,3} = 1$; $x_{i,1} = 1$, $x_{i,2} = 0$, $x_{i,3} = 1$; $x_{i,1} = 1$, $x_{i,2} = 1$, $x_{i,3} = 0$. "Fluxes" $X_{i,j}$ in the branches of the same quadruple are made to be mutually exclusive with one another by nonlinear transmissions between the corresponding pistons and the total flux across branches labeled by the same value of the same Boolean variable is made to be conserved across different quadruples by linear transmissions between the corresponding pistons. By applying a force to the input piston Q, the machine's motion from $Q = 0$ to $Q > 0$ instantly produces a solution under the simultaneous influence of all the problem constraints (in each quadruple, there is only one branch with flux > 0, the series of all these branches is labeled by a Boolean assignment that solves

the system). By using the partial OR (POR) gate instead of the NAND gate, quadruples can be replaced by triples.

This idealized machine has the only purpose of introducing the idea of simultaneous computation, namely of a computation mechanism that, thanks to a perfect nonfunctional simultaneous dependence between all its degrees of freedom, nondeterministically produces the solution of a problem under the simultaneous influence of all the problem constraints.

3 Quantum Computation as Simultaneous Computation

To see that quantum computation is simultaneous computation, we should replace the configuration space of the idealized classical machine by the phase space of the quantum machine. Simultaneous dependence between the coordinates of the machine parts becomes simultaneous dependence between the populations of the reduced density operators of the parts of the computer register. Let us consider the previous example of solving the Boolean equation $y = \overline{x}$. The motion of the idealized machine from $Q = 0$ to $Q > 0$ is analogous to measuring two qubits in the entangled state $|0\rangle_X|1\rangle_Y + |1\rangle_X|0\rangle_Y$. Simultaneous dependence is represented by a relation between symbolic variables. Let us represent the populations of the reduced density operator of the first (second) qubit by the variables x_{11}, x_{22} (y_{11}, y_{22}). The state before measurement corresponds to the assignment $x_{11} = x_{22} = y_{11} = y_{22} = \frac{1}{2}$, the state after measurement to $x_{11} = 1$, $x_{22} = 0$, $y_{11} = 0$, $y_{22} = 1$ or, in a mutually exclusive way, $x_{11} = 0$, $x_{22} = 1$, $y_{11} = 1$, $y_{22} = 0$. The transition imposed by the quantum principle is isomorphic with the transition from $Q = 0$ to $Q > 0$ of the idealized classical machine and can be represented in exactly the same way. The correspondence between coordinates and populations is:

$$\frac{X}{Q} = x_{11}(t_r) = 1 - x_{22}(t_r), \qquad \frac{Y}{Q} = y_{11}(t_r) = 1 - y_{22}(t_r). \tag{3}$$

Reduction occurs under the same non functional simultaneous dependence, repeated here for convenience:

$$\frac{X}{Q} + \frac{Y}{Q} = 1, \tag{4}$$

$$\left(\frac{X}{Q}\right)^2 + \left(\frac{Y}{Q}\right)^2 = 1. \tag{5}$$

The transition from $Q = 0$ to $Q > 0$ produces state vector reduction, which is thus isomorphic with a many body interaction that performs a nondeterministic computation. The infinite precision required by the classical machine to produce the solution through a many body interaction is thus replaced by state vector reduction. That infinite classical precision can be dispensed for because of quantization was already noted by Finkelstein [21].

Summing up, the solution of the problem is reversibly and nondeterministically produced under a nonfunctional simultaneous dependence representing all the problem constraints and conserved throughout state vector reduction. This is an alternative way of formulating the quantum principle (stating that the state before measurement is projected on the subspace of an eigenvalue of the measured observable), closer to the notion of physical law—of transformation under a perfect nonfunctional simultaneous dependence—and more suited to the present computational context.

I should note that simultaneous dependence is not limited to the populations immediately before and after measurement, it extends to all the amplitudes of the basis vectors $\alpha_j(t)$ at any time t of the process: preparation, unitary development, and measurement. In fact the transition from $Q = 0$ to $Q > 0$ changes any $\alpha_j(t)$ from the value before reduction to the value after reduction, changing the forward development into the backward development (the same unitary transformation but ending with the state after reduction). We should think to supplement equations (3), (4), and (5), representing the nonfunctional part of the simultaneous dependence, with the infinite system of equations representing (for any time t) the functional dependence of the $\alpha_j(t)$ on the $x_{i,i}(t_r)$.

4 An Explanation of the Speed up

The fact that, in the quantum context, simultaneous computation is the nondeterministic production of the solution under a nonfunctional simultaneous dependence representing the problem-solution interdependence, is almost evident in the algorithms of Simon [32] and Shor [31]. Here the computation of the periodic function $f(x)$produces an entangled state of the form

$$\sum_{x=1}^{N} |x\rangle_X |f(x)\rangle_K; \tag{6}$$

this is the unitary development of an initial state where register $X$is in an even superposition of all the possible values of x and register K is in a sharp value. The final measurement is equivalent to measuring the content of register K—the value of $f(x)$—in state (6), for the retroactivity of state vector reduction in a reversible evolution. Say that the outcome of measurement is $f(x_0)$. This measurement induces a state vector reduction from state (6) to

$$\left(\sum_{m=1}^{M} |x_0 + mT\rangle_X\right) |f(x_0)\rangle_K, \tag{7}$$

where T is the period. By applying the quantum Fourier transform to this superposition of arguments (another unitary transformation), one extracts the period of the function (disregarding the probability of error); therefore one can say that (7) is the solution of the problem up to a unitary transformation of the computational basis. We can see that the amplitudes of the basis vectors throughout preparation/unitary evolution/measurement are submitted to a nonfunctional simultaneous dependence representing problem-solution interdependence, exactly as illustrated in Sect. 3. State vector reduction produces the solution under this simultaneous dependence. This is of course simultaneous computation.

On the contrary, simultaneous computation is completely hidden in Deutsch's [16] and Grover's [23] algorithms, which yield their speed ups through unitary evolutions—I am presently considering Cleve's et al. [15] revisitation of Deutsch's algorithm and Grover's algorithm for a database size that provides no probability of error. However, this can be ascribed to the fact that these algorithms physically represent only the procedure that leads to the solution, whereas the interdependence between the problem and the solution (an essential feature of simultaneous computation) is disregarded.

We are dealing with quantum games. One player chooses at random one of the four functions in Deutsch's problem, or a data base location in Grover's problem, the other player must find out the choice of the first player (a character thereof in Deutsch's problem), but

the physical representation does not comprise the random generation of the move of the first player. Simultaneous computation (problem-solution interdependence) does not appear since the problem is not represented physically.

Let us focus on Grover's algorithm and let the size of the database be N. In the conventional algorithm, the quantum database is represented by a quantum computer that, given an input x, computes $\delta(k,x)$, where δ is the Kronecker function and k is the database location randomly chosen by the first player. For each input x provided by the second player, the computation of $\delta(k,x)$ tells whether it is the database location chosen by the first player. The second player prepares the input register X in an even superposition of all the possible values of x. To find out the choice of the first player, the algorithm has to compute $\delta(k,x)$ the order of $\sqrt{N}$ times, instead of N like in the classical case.

To physically represent the problem, it suffices to represent the random generation of k on the part of the first player. To this end, we add an ancillary register K prepared in a superposition of all the possible values of k. The extended algorithm repeatedly computes $\delta(k,x)$ as before but now for a superposition of all the possible combinations of values of k and x. This entangles each possible value of k with the corresponding solution (the same value of k) found by the second player at the end of the algorithm. For example, with database size $N=4$, the state before measurement is:

$$\frac{1}{2\sqrt{2}}\big(|00\rangle_K|00\rangle_X+|01\rangle_K|01\rangle_X+|10\rangle_K|10\rangle_X+|11\rangle_K|11\rangle_X\big)\big(|0\rangle_F-|1\rangle_F\big). \tag{8}$$

Measuring the content of registers K and X determines the moves of both players—also representing the random choice of the value of k on the part of the first player. The state vector reduction induced by measuring the content of register K can be backdated to before running the algorithm. This leaves the initial preparation of register X—a superposition of all the possible values of x—unaltered (because of the unitary transformations in between) and brings that of register K to a sharp value, thus representing exactly the original Grover's algorithm.

Thus, by completing the physical representation of Grover's algorithm, one finds again a succession of entanglement and disentanglement and the nondeterministic production of the solution under a nonfunctional simultaneous dependence representing the problem-solution interdependence. The nondeterministic production of the contents of registers K and X can be seen as mutual determination between these contents, which justifies the square root speed up with respect to a classical database search, where the content of the former register determines that of the latter and not vice-versa. By ascribing the speed up to mutual determination between register contents, one finds that it is bounded by state vector reduction through an entropic inequality, as follows.

Mutual determination does not mean that the choice of the first player determines the solution found by the second player at the end of the algorithm, which would be the classical database search. Neither it means that the solution found by the second player at the end of the algorithm creates the choice of the first player, which would also be unilateral determination.

Mutual determination is symmetrical, it can be represented by saying that the contents of the two registers are determined by the measurement of the first (second) bit of register K and the second (first) bit of register X. Thus Grover's algorithm is equivalent to the following game. We should think to arrange the N database locations in a matrix of $\sqrt{N}$ columns and $\sqrt{N}$ rows—with $N=4$ the row can be identified by the first bit of either register, the column by the second bit. At the end of Grover's algorithm, the first player determines, say, the row by measuring the first bit of register K in state (8)—this is equivalent to determining

the row before running the algorithm, for what said before. The second player determines the column by measuring the second bit of register X. The related state vector reduction can be backdated to before running the algorithm, namely to the initial preparation of the two registers K and X, each in an even superposition of all the possible values of, respectively, k and x. This leaves the initial preparation of register X unaltered and reduces that of register K to the superposition of all the values of k ending by that bit (determining the column before running the algorithm). In this picture, Grover's algorithm searches just the row randomly chosen by the first player, which justifies the $O(\sqrt{N})$ computations of $\delta(k, x)$, i.e. the square root speed up (of course the picture should be symmetrized for the exchange of columns and rows).

The same justification holds in the case that the value of k is already determined before running the algorithm, like in virtual database search; this situation is indistinguishable from the random generation of k at the end of the algorithm, since state vector reduction can be backdated so that k is already determined before running the algorithm. With k predetermined, the preparation of register K in an even superposition of all the possible values of k represents the initial ignorance of the value of k on the part of the second player. Since there is no more determination of the column on the part of the second player, mutual determination between the contents of registers K and X becomes "precognition" of the column on the part of the second player. "Precognition" corresponds to backdating, to before running the algorithm, the state vector reduction induced by the measurement of (say) the second bit of register X, which leaves (as said before) the initial preparation of register X unaltered and determines the second bit in the initial preparation of register K (determines the column), reducing the initial ignorance of the second player about the value of k. The related information gain is

$$\Delta S = \frac{1}{2} \lg N, \tag{9}$$

one bit with $N = 4$. Besides database size, N is the ratio between the size of the superposition before measurement (8 terms with amplitudes even in modulus—see (8)) and the size of the subspace on which the superposition is projected by quantum measurement (the 2 dimensions of the Hilbert space of register F). It is thus a measure of state vector reduction.

I should like to quote the question raised by Grover in his paper [24]: "*What is the reason that one would expect that a quantum mechanical scheme could accomplish the search in* $O(\sqrt{N})$ *steps*? *It would be insightful to have a simple two line argument for this without having to describe the details of the search algorithm.*" The "precognition" explanation might provide this argument. Casting it in two lines: "*the speed up is the reduction of the initial ignorance of the solution due to backdating, to before running the algorithm, a time-symmetric part of the state vector reduction on the solution.*"

A similar extension of Deutsch's algorithm yields the state before measurement:

$$\frac{1}{2\sqrt{2}}\big[\big(|00\rangle_K + |11\rangle_K\big)|0\rangle_X + \big(|01\rangle_K + |10\rangle_K\big)|1\rangle_X\big]\big(|0\rangle_F - |1\rangle_F\big) \tag{10}$$

where $k = 00, 01, 10, 11$ specifies the function randomly chosen by the first player and x is the answer provided by the second player (whether the function is balanced or constant). State (10) is reached by invoking the computation of the function only once instead of the two times required in the classical case. Although things are less symmetrical than in database search, as the two registers have different length, there is still a succession of entanglement and disentanglement, and mutual determination between the contents of the two

registers K and X. The information gain $\Delta S = \frac{1}{2} \lg N$, associated to backdating a time symmetric part of the state vector reduction on the solution, is one bit—the ratio of Hilbert space sizes before and after measurement is still $N = 4$. This is consistent with the fact that the speed up of Deutsch's algorithm consists in having to check the value of one bit rather than the two required in the classical case.

Equation (9) can be rewritten by noting that $\lg N$, the logarithm of the squeeze of Hilbert space size, is the von Neumann entropy of the reduced density operator of register K in state (8):

$$\Delta S = \frac{1}{2} \Delta R. \tag{11}$$

I call this entropy ΔR since it is also the decrease of entropy of register K during state vector reduction—before reduction K is maximally entangled, after reduction it is in a sharp state and its entropy is zero. ΔR can be used as an entropic measure of state vector reduction. It is more general than the logarithm of the ratio of Hilbert space sizes, with which it coincides in the case of even modulus amplitudes.

Similarly, the information gain ΔS associated with partial backdated state vector reduction can be used as a measure of the speed up. This means defining the speed up—when applicable—as the reduction of the logarithmic size of the problem such that the time taken by the quantum algorithm to solve the problem is the same as the time taken by the classical algorithm to solve the reduced problem.

For example, in the case of Grover's algorithm, if database size is $N = 4$, the logarithmic size of the problem is $\lg 4 = 2$ (the number of bits of register K), the logarithmic size of the reduced problem is $\lg 2 = 1$. The time taken by the quantum algorithm to solve the problem of 2 bits is the same as the time taken by the classical algorithm to solve the problem of 1 bit—in both cases $\delta(k, x)$ is computed once.

Equation (11) states that this measure of the speed up is 50% of the entropic measure of state vector reduction in both Grover's and Deutsch's algorithms. These algorithms concern unstructured problems. More in general, the notion that the speed up is partial backdated state vector reduction implies:

$$\Delta S \leq \Delta R, \tag{12}$$

where ΔR, the entropic measure of state vector reduction, can be defined in general as the entropy of the reduced density operator of the observable being measured. We can do without the details of the quantum algorithm by considering the state immediately before the measurement projection, when the observable is maximally entangled with the pointer of the measurement apparatus. In particular, inequality (12) states that, when the problem-solution interdependence is physically represented, there is no speed up without state vector reduction [9].

5 Positioning

I try to position the present approach within the early development of the notion that computation is a physical process. This development can be segmented as follows: (I) Investigation of the thermodynamic aspect of classical computation [4, 27]. (II) Introduction of the computational notion of chronon in quantum relativity [19]. (III) Introduction of the notion of quantum bit and identification of computation in the quantum framework [20]. (IV) Discovery of the logical reversibility of computation and that reversible classical computation

dissipates no energy in the limit of zero speed [2, 3]. (V) Discovery of an idealized classical model of reversible computation [22]. (VI) Development of quantum algorithms with no speed up [1, 18]. (VII) Discovery of the quantum speed up, namely that there are quantum algorithms essentially quicker than their classical counterparts [16]. Since then, research has focused on finding more speed ups, fostering the robustness of the computation process against decoherence, and developing laboratory implementations.

One can distinguish two phases in the above development. In the first phase, (II) through (VI), quantum computation is a unitary evolution and there is no expectation of a speed up. The fundamental computation model is deterministic reversible computation. The second phase starts with the first speed up discovered by Deutsch (VII). The fact that it is reached through a unitary evolution is here ascribed to the incompleteness of the physical representation. By completing the representation (by representing the problem), one can see that the solution is nondeterministically produced under a nonfunctional simultaneous dependence representing problem-solution interdependence. This shows that quantum computation is simultaneous computation and that the speed up is bounded by state vector reduction through an entropic inequality. Thus, in the second phase, the deterministic reversible computation model of the first phase should be replaced by the nondeterministic reversible model of simultaneous computation. One goes from the former to the latter model by replacing functional simultaneous dependence by nonfunctional simultaneous dependence.

6 Toward the Turin Workshops

The origin of my special interest in a simultaneous form of computation goes back to an intuition on the nature of the mind I had at the age of 16. I thought to have seen that our mind is generated by the dialog between a limited number of "fundamental feelings"; each feeling, while keeping its identity, was becoming something new and richer in the dialog, while the entire dialog was simultaneously perceived. An account of that experience can be found in the on-line Archives of Scientists' Transcendent Experiences, under the title "From an altered state of consciousness to a life long quest of a model of mind" [8].

When I studied electronic engineering at the Politecnico of Turin, I went in depth into the theory of formal languages, artificial intelligence, and self organizing systems, looking for a correspondence with some aspects of the intuition. After my university degree, compromising with practical needs, I started working to avionics in the aviation division of Fiat, then moved to Seattle where I worked for three years at Boeing and experienced what a best practice at an international level is. Eventually I came back to Italy to work in Elsag, the company that later sponsored the international Elsag Bailey-ISI Turin workshops. I was attracted by a very advanced artificial intelligence program that was on its start. Under the direction of Luigi Stringa, we developed the first programmable parallel processor commercialized in the world, with industrial applications to pattern recognition. The system was used for address recognition by the postal services of Italy, the United States, and several European countries.

Still with Elsag, I used my familiarity with the US technology environment to organize joint projects with some of the best US information technology groups. Noticeably, we have been working with two of the creators of ARPANet, Vinton G. Cerf, 2004 Alan Turing Award for inventing with Robert E. Kahn the TCP/IP protocol, and Barry Wessler—and their groups. By using the internetworking architecture underlying Internet, we developed nationwide hybrid electronic mail systems and tracking and tracing systems for the Italian Post

office and, in subsequent years, many others. The large systems division I was responsible for was profitable, with its yield Elsag started an acquisition strategy, in particular through the friendly acquisition of the American Bailey, thus becoming Elsag Bailey and second in the world in the electronic control of continuous industrial processes, immediately after Honeywell.

In that period, my compromising with the wish to decipher the intuition consisted in comparing my recollection of it with my growing understanding of information processing. However, classical computation—even the parallel processing that attracted me to Elsag—did not match the recollection. This brought me to quantum computation along the lines already expounded.

I went to Mario Rasetti (we have been school-fellows at both high school and the Politecnico of Turin) who got interested in my ideas and soon discovered that there was a newly borne quantum information theory—we were in the late eighties. Our shared interest in the new theory, my habit to join the best IT practices at an international level, and the ISI bent to scientific interchange naturally brought us to think of an international workshop on quantum information.

I involved my company, obtaining from its chief executive officer Enrico Albareto—who was strongly oriented toward innovation and had already funded two chairs in US universities (MIT and Harvard) at the time of the acquisition of Bailey—the authorization to finance the workshop within my division's budget. Naturally I cared to give my company something in exchange. Today Elsag (the multinational Bailey part has been sold to Hartman & Brown by the holding company Finmeccanica) is still hosting a quantum lab and developing a quantum cryptography product centered on its market.

In 1992 we signed a cooperation protocol between Elsag Bailey and ISI and soon afterwards got in touch with Roger Penrose who helped us to organize the winter 1992 Oxford meeting (in view of the first Turin workshop). I would like to conclude the present account by providing a testimony of that meeting.

I remember the philosopher of science David Albert and the physicists Charles Bennett, David Deutsch, Artur Ekert, Roger Penrose (we were guests of his Oxford University office), Mario Rasetti, Tom Toffoli, and Wojcieck Zurek, each in turn laying out his own vision of the field of quantum information, with David Deutsch already addressing in a unified way quantum computing, the philosophy of science of Popper and the theory of evolution of Darwin, Artur Ekert speaking of the practical application of quantum cryptography, Charles Bennett talking about the theory of teleportation, and all together discussing the organization of the future Turin workshop. There was a curious debate about its title, with a preference of Charles Bennett for quantum communication, of David Deutsch and David Albert for quantum computation, and a simple trade off proposed by Tom Toffoli, "quantum communication and computation", which of course remained in time.

But what struck me more was the enthusiastic desire of exchanging with one another ideas and recent results in a long dinner evening in a nearby pub, with Artur Ekert explaining to me the possible commercial applications of quantum cryptography, Tom Toffoli illustrating to my former boss Giuseppe Cuneo and colleague Giorgio Musso his universal cellular automaton, David Deutsch and David Albert arguing pro and against Popper before an amused Tom Toffoli, and everybody writing down formulas on paper napkins and tablecloths. In that evening, all the factors that made the Turin workshops and the development of the field a success were already coming out with evidence. Then those workshops remained historically geared with an extraordinarily lucky crossover between computer science and quantum computing, between theoretical and experimental quantum information, and with the fast development of the field. By the way, in spite of the presence of many blackboards

in the ISI institute, in villa Gualino on the hills of Turin, writing on paper napkins and tablecloths remained the preferred means of communication between physicists throughout those many workshops.

Besides the Turin workshops, my activity in the field of quantum computation can be summarized as follows. Since the first Turin workshop, I put forward the concept of quantum ground state computation [5, 6, 12, 13], which is natural in the context of simultaneous computation. The linear relations between populations of different observables required by simultaneous computation are implemented through a series of relaxations on the ground state. This form of computation promised a quantum speed up because of tunneling through energy barriers. Today, quantum ground state computation is believed to yield a square root speed up in any NP problem, but in the early nineties it did not attract much attention. Because of its mathematical intractability, it could not compete with the well defined unitary evolutions of the quantum algorithms, which were at that time in their bloom. At the end of the nineties, quantum ground state computation started receiving more attention. I joined forces with Artur Ekert, who had the idea of replacing the heat bath interaction by the adiabatic deformation of an initial trivial Hamiltonian into the problem Hamiltonian. We were on the point of working out the idea when there was the publication of the paper by Farhi et al. [17], as it happens in times of scientific competition. In [11], we also tried to replace the dynamic constraint of minimum energy by kinematic constraints due to particle statistics; in [26], we replaced dynamics by kinematics—a search that I consider still in progress.

After the Turin workshops, I published, with others, a few papers that ascribe the quantum speed up to the non causal joint-determination of the measurement outcome by the state before measurement and the quantum principle [7, 10, 14]. We showed that joint-determination was responsible for all the speed-ups discovered until then. Soon afterwards [30] there was the first paper on cluster computing, where the use of entanglement and disentanglement by quantum measurement becomes explicit.

Acknowledgements Thanks are due to David Finkelstein and Artur Ekert for encouragement to write down my way of thinking about quantum computation and, extended to Shlomit Ritz Finkelstein, for stimulating discussions.

References

1. Benioff, P.: Quantum mechanical Hamiltonian models of Turing machines. J. Stat. Phys. **29**, 515 (1982)
2. Bennett, C.H.: Logical reversibility of computation. IBM J. Res. Dev. **6**, 525 (1973)
3. Bennett, C.H.: The thermodynamics of computation—a review. Int. J. Theor. Phys. **21**, 905 (1982)
4. Brillouin, L.: Science and Information Theory. Academic Press, New York (1956)
5. Castagnoli, G.: Quantum steady computation. Int. J. Mod. Phys. B **5**(13), 2253 (1991)
6. Castagnoli, G.: Quantum nondeterministic computation based on fermionic antisymmetry. In: Toffoli, T., Biafore, M., Leao, J. (eds.) Proceedings of the Fourth Workshop on Physics and Computation. Boston University, 22–24 November 1996
7. Castagnoli, G.: Quantum computation based on retarded and advanced propagation. quant-ph/9706019 (1997)
8. Castagnoli, G.: From an altered state of consciousness to a life long quest of a model of mind. In: Tart, C.T. (ed.) TASTE Archives of Scientists' Transcendent Experiences, submission N 00088 (2002); http://www.issc-taste.org/arc/dbo.cgi?set=expom&id=00088&ss=1
9. Castagnoli, G.: Quantum problem solving as simultaneous computation. arXiv:0710.1744 (2007)
10. Castagnoli, G., Finkelstein, D.: Theory of the quantum speed up. Proc. Roy. Soc. Lond. A **457**, 1799 (2001); quant-ph/0010081
11. Castagnoli, G., Finkelstein, D.: Quantum ground state computation with kinematical gates. Proc. Roy. Soc. Lond. A **459**, 3099 (2003)
12. Castagnoli, G., Rasetti, M., Vincenzi, A.: Steady, simultaneous quantum computation: a paradigm for the investigation of nondeterministic and non-recursive computation. Int. J. Mod. Phys. C **3**(4), 661 (1992)

13. Castagnoli, G., Ekert, A., Macchiavello, C.: Quantum computation: from the sequential approach to simulated annealing. Int. J. Theor. Phys. **37**(1), 463 (1998)
14. Castagnoli, G., Monti, D., Sergienko, A.: Performing quantum measurement in suitably entangled states originates the quantum computation speed up. quant-ph/9908015 (1999)
15. Cleve, R., Ekert, A., Macchiavello, C., Mosca, M.: Quantum algorithms revisited. Proc. Roy. Soc. Lond. A **454**(1969), 339–354 (1997)
16. Deutsch, D.: Quantum theory, the Church–Turing principle and the universal quantum computer. Proc. Roy. Soc. Lond. A **400**, 97 (1985)
17. Farhi, E., Goldstone, J., Gutmann, S., Sipser, M.: Quantum computation by adiabatic evolution. arXiv:quant-ph/0001106 (2000)
18. Feynman, R.P.: Quantum mechanical Computers. Opt. News **11**, 11 (1985); reprinted in Found. Phys. **16**(6) (1986)
19. Finkelstein, D.R.: Space-time code. Phys. Rev. **184**, 1261 (1969)
20. Finkelstein, D.R.: Space-time structure in high energy interactions. In: Gudehus, T., Kaiser, G., Perlmutter, A. (eds.) Coral Gables Conference on Fundamental Interactions at High Energy, Center of Theoretical Studies, University of Miami, 22–24 January 1969, pp. 324–343. Gordon and Breach, New York (1969)
21. Finkelstein, D.R.: Generational quantum theory. Preprint, to become a Springer book (2008)
22. Fredkin, E., Toffoli, T.: Conservative logic. Int. J. Theor. Phys. **21**, 219 (1982)
23. Grover, L.K.: A fast quantum mechanical algorithm for database search. In: Proc. of the 28th Ann. ACM Symp. on Theory of Computing (1996)
24. Grover, L.K.: From Schrodinger equation to quantum search algorithm. Quant-ph/0109116 (2001)
25. Hameroff, S.R., Penrose, R.: Orchestrated reduction of quantum coherence. In: Hameroff, S.R., Kaszniak, A.W., Scott, A.C. (eds.) Brain Microtubules: A Model for Consciousness? Toward a Science of Consciousness—the First Tucson Discussions and Debates, pp. 507–540. MIT Press, Cambridge (1996)
26. Jones, J.A., Vedral, V., Ekert, A.K., Castagnoli, G.: Geometric quantum computation using nuclear magnetic resonance. Nature **403**, 869–871 (2000); quant-ph 9910052
27. Landauer, R.: Irreversibility and heath generation in the computing process. IBM J. Res. Dev. **3**, 113 (1961)
28. Mulligan, K., Smith, B.: Mach and Ehrenfels: The Foundations of Gestalt Theory. In: Smith, B. (ed.) Foundations of Gestalt Theory, p. 124. Philosophia, Munich/Vienna (1988); http://ontology.buffalo.edu/smith/articles/mach/mach.html - N_1_
29. Penrose, R.: Shadows of the Mind—a Search for the Missing Science of Consciousness. Oxford University Press, Oxford (1994)
30. Raussendorf, R., Briegel, H.J.: Quantum computing via measurements only. arXiv:quant-ph/0010033 v1 (7 October 2000)
31. Shor, P.: Algorithms for quantum computation: Discrete logarithms and factoring. In: Proc. 35th Ann. Symp. on Foundations of Comp. Sci., pp. 124–134 (1994)
32. Simon, D.: On the Power of Quantum Computation. In: Proc. 35th Ann. Symp. on Foundations of Comp. Sci., pp. 116–123 (1994)
33. Stapp, H.P.: Quantum theory and the role of mind in nature. arXiv:quant-ph/0103043 (2001)