Quantum Computing by Simulations.

Michail Zak.
Jet Propulsion Laboratory California Institute of Technology.

Chapter 1.
Quantum recurrent nets.

Abstract.

Quantum computing by simulations is based upon similarity between mathematical formalism of quantum mechanics and phenomena to be computed. It exploits a dynamical convergence of several competing phenomena to an attractor which can represent an extrenum of a function, an image, a solution to a system of ODE, or a stochastic process. In this chapter, a quantum version of recurrent nets (QRN) as an analog computing device is discussed. This concept is introduced by incorporating classical feedback loops into conventional quantum networks. It is shown that the dynamical evolution of such networks, which interleave quantum evolution with measurement and reset operations, exhibit novel dynamical properties. Moreover, decoherence in quantum recurrent networks is less problematic than in conventional quantum network architectures due to the modest phase coherence times needed for network operation. It is proven that a hypothetical quantum computer can implement an exponentially larger number of the degrees of freedom within the same
size.

1. 1.1. Introduction.

Natural computing is based upon similarity between mathematical formalism of a physical phenomenon to be simulated and phenomena to be computed. Usually it exploits a dynamical convergence of a physical process to a certain state, or attractor, so that the measured parameters characterizing this attractor can be uniquely identified. Thus, unlike digital computers which operate via manipulations with numbers, in analog computers numbers appear as a result of measurement of physical parameters. That is why the criteria of computational complexity developed for digital algorithms, strictly speaking, are not applicable to analog algorithms. At the same time, analog algorithms have their own criteria of "complexity" such as: the time of convergence to an attractor subject to a prescribed error, the degree of stability of the attractor, the pattern of convergence (asymptotic or oscillatory), type of the attractor (static, periodic, chaotic, or stochastic), etc.

The competition between digital and analog computers, i.e., between computations and simulations, has a long history. During the last fifty years, the theory of computation has been based, implicitly, upon classical physics as idealized in the deterministic Turing machine model. However, despite the many successes of digital computers, the existence of so called hard problems has revealed limitations of their capabilities, since the computational time for solving such problems grows exponentially with the size of the problem.

It was well understood that one possible way to fight the "curse" of the combinatorial explosion is to enrich digital computers with analog devices. In contradistinction to a digital computer, which performs operations on numbers symbolizing an underlying physical process, an analog computer processes information by exploiting physical phenomena directly. It is this problem solving via direct simulation that allows an analog approach to reduce the complexity of the computations significantly. This idea was stressed by Feynman, who demonstrated that the problem of exponential complexity in terms of calculated probabilities can be reduced to a problem of polynomial complexity in terms of simulated probabilities. Conceptually, a similar approach can be applied to the whole class of NP-complete problems. Indeed, the theory of computational complexity is an attribute of digital approach to computations. At the same time, in principle, one can find such a physical phenomenon whose mathematical description is equivalent to those of a particular NP-complete problem. Then, incorporating this phenomenon into an appropriate analog device, one can simulate the corresponding NP-complete problem. But is it possible, in general, to find a new mathematical formulation for any intractable problem in such a way that it becomes tractable? Some experts in computational complexity believe that, in the spirit of the Godel theorem, there always exist computational problems such that every mathematical formulation that captures the essence of the problem is intractable,. At this step, we cannot prove or disprove this statement.

There is another class of problems for which simulations are superior over computations. In contradistinction to NP-complete problems whose complexity is in an exponentially large number of simple computations, these problems have complex and sometimes, partially unknown analytical structure. Simulations of solutions of such problems are based upon a black-box approach when unknown components of the model are found in the course of a trial-and-error learning process. A typical representative of a corresponding analog device implementing black-box based simulations is a neurocomputer where unknown (learnable) parameters are incorporated in the form of synaptic interconnections between dynamical units called “neurons”. However, usually analog computers are associated with certain limitations such as the lack of universality, slow performance, and low accuracy, and this is the price to be paid for certain advantages of simulations. A partial success in development of a universal analog device is associated with neurocomputers which are based upon massively parallel adaptive dynamical systems modeled on the general features of biological neural networks that are intended to interact with the object of the real world in the same way the biological systems do. However, the capacity of the neurocomputers is roughly proportional to the size of the apparatus, and that limits actual power significantly.

A second way to fight a curse of dimension is to utilize a non-deterministic approach to computations. This approach is associated with the Monte Carlo method introduced by N.C. Metropolis and S.M. Ulam in 1940. The idea of this method is based upon the relationships between the probabilistic characteristics of certain stochastic processes and solutions to some deterministic problems such as values of integrals, solutions to differential equations, etc. The strength of the method is that its error does not depend on the number of variables in the problem, and therefore, if applicable, it breaks the curse of dimension. The effectiveness of the Monte-Carlo approach is inversely proportional to the smoothness parameter that characterizes the degree of correlation within the input data. However, the Monte-Carlo method is not the only way to apply nondeterminism for computations. There is a class of so-called randomized algorithms that are effective for combinatorial problems. In general, a randomized strategy for this kind of problem is useful when there are many ways in which an algorithm can proceed, but it is difficult to determine a way that is guaranteed to be good. In particular, if the benefits of good choices outweigh the costs of bad choices, a random selection of good and bad choices can yield a good algorithm.

In general, the theory of computational complexity proves that polynomial time nondeterministic algorithms are more powerful than polynomial time deterministic ones. However, the main limitation of the whole non-deterministic approach is in the generation of random numbers: the generators are slow and not always reliable (i.e., the sequence of numbers that they produce may harbor hidden correlations that no truly random sequence would possess). That is why the concept of a quantum computer became so attractive: its analog nature is based upon physical simulations of quantum probabilities, and at the same time, it is universal (at least for modeling physical world).

Although the development of the quantum-mechanical device is still in progress, a new quantum theory of computations has been founded, C.Williams, 1997. This theory suggests that there is a second fundamental advantage of the hypothetical quantum computer which is based upon the wave properties of quantum probabilities: a single quantum computer can follow many distinct computational paths all at the same time and produce a final output depending on the interference of all of them. This particular property opened up a new chain of algorithms that solve in polynomial time such hard problems as factorization and discrete log, i.e., the problems that are believed to be intractable on any classical computer.

In order to clarify the connection between quantum algorithms and combinatorial optimization, consider n binary variables $x = x_1, x_2, .. x_n; x_i ... = 0,1;$ and combine them into larger number of variables as all possible products of n old variables:

$$y = x_1 \otimes x_2 \otimes ... \otimes x_n = x_1 x_2 ... x_m, x_2, x_1, .. x_m, \ \ etc \qquad \text{(I.1.1)}$$

The number of these new variables is $N = 2^n$. In many practical applications, a function to be optimized is defined at a set of the new variables. For instance, in the course of a spacecraft design, the optimal placement of sensors requires to minimize the cost function which depends upon 2^n values of y, since each of n assigned places can be equipped or not equipped by a sensor. Since the number of possible assignments grows exponentially with the number of placements, it appears that the time required to solve this problem must also grow exponentially (in the worst case) even if a single computation of each of 2^n values of the cost function is trivial. Actually it is this property which makes the problems of combinatorial optimization intractable by classical computing.

There is a striking similarity between the structure of combinatorial problems and some special properties of quantum evolution, namely, the property of the direct-product-decomposability. This property follows from the fact that if two unitary matrices U_1 and U_2 are solutions to the Schrödinger equation, then their tensor product $U_1 \otimes U_2$ will be also the solution to it. Therefore, with an input of *n* binary variables of the type x, one can obtain 2^n variables y as an output in one computational run. In other words, the transition from n basic variables x to 2^n combinatorial variables y is carried out by the laws of nature, and that is the *natural* foundation of quantum computing.

Actually the transition from x to y is carried out by n of 2 by 2 identity matrices $I^{(i)}$ as following:

$$y_{j_1 \cdots j_n} = \sum I_{j_1 i_1}{}^{(1)} \ldots I_{j_n i_n}{}^{(n)} x_{i_1 \ldots i_n} \qquad \text{(I.1.1.2)}$$

Replacing identity matrices by non-identical unitary matrices $S^{(i)}$ one finds a new variable $z_{j_1 \cdots j_n}$ which is combined of weighted sums of all the components of the variable *y,* and that is due to another fundamental property of quantum mechanics: the interference of probabilities that follows from the Schrödinger equation.

If the matrices $S^{(i)}$ are chosen such that the variable z is equal to the cost function, then the computation is accomplished: the output contains all the 2^n values of the cost function. However, in order to find the optimal value of the combinatorial variable $y = y_0$, one has to impose an additional constraint upon the matrices $S^{(i)}$, namely: the weight coefficient of y_0 must dominate over other weight coefficients in order to detect this optimal value in a few number of measurements, and this constraint is, probably, the toughest. Hence quantum computing does not allow iterations, feedbacks, or any other types of control over the computational process: one must get the solution at once, or he does not get it at all.

Thus, there are at least two areas where the quantum computer is expected to be superior over the classical one: quantum mechanics (due to simulation of quantum probabilities), and some specific combinatorial problems linked to operation research (due to interference of probabilities, ditect-product-decomposability, and entanglement.

In this chapter an attempt is made to combine the power of quantum computing and the dynamical complexity of neural nets. There are at least three reasons for such combinations. Firstly, it will represent a universal analog device with a built-in random number generator. Secondly, its capacity will be exponentially larger than those of a classical neurocomputer due to the superposition and entanglement effects. Thirdly, it will introduce iterations in quantum computing. The main challenge of the approach is in reconciliation of linear reversible quantum evolution and nonlinear irreversible dynamics of neural nets.

This chapter introduces a new dynamical paradigm: quantum recurrent nets (QRN). There are remarkably few papers in which quantum natural computing is discussed. The first one, Cerny 1993, introduces a hypothetical quantum device (a slot machine) for solving a traveling salesman problem. As shown by the author, such a device, although intellectually appealing, requires an exponentially large number of measurements to get the right answer. Another one, M.Zak 2005, represents an attempt to exploit combinatorial properties of tensor product decomposability of unitary evolution of many-particle quantum systems for simulating solutions to NP-complete problems, the reinforcement and selection of the desired solution being executed by quantum resonance; although the implementability of the approach is still in question, the potential difficulties are not associated with the NP-completeness of the problem. It will be discussed in the next chapter of this book.

1.1.2. Neural net as a dynamical system.

A neural net as a nonlinear dissipative dynamical system can be represented by the following set of ODE:

$$\tau_i \dot{x}_i = -x_i + \sigma\left(\sum_j T_{ij} x_j\right), \qquad \tau_i > 0 \qquad \text{(I.1.2.1)}$$

where x_i are state variables, or mean soma potentials, characterizing the neuron activities,

T_{ij} are constant control parameters representing the weights of synaptic interconnections, τ_i are suitable time constants, and $\sigma(\cdot)$ is a sigmoid function having a saturated non linearity (usually $\sigma(x) = \tanh \beta x$ where $\beta = const > 0$ is an additional control parameter).

An invariant characterizing the local dissipativity of the system (1) is expressed explicitly via its parameters:

$$divx = \sum_i \frac{1}{\tau_i}(-1 + \frac{\beta T_{ii}}{\cosh^2 \sum T_{ij} x_j}) \qquad \text{(I.1.2.2)}$$

A necessary (but not sufficient) condition that the system (1) has attractors is that there are some domains in phase space where the invariant (2) is negative.

If the matrix T_{ij} is symmetric

$$T_{ij} = T_{ji} \qquad \text{(I.1.2.3)}$$

then equation (1) can be presented in the form of a gradient system, and therefore it can have only static attractors. In the basin of a static attractor, the invariant (2) must be negative.

Since the system (1) is nonlinear, it can have more than one attractor; consequently, in some domains of phase space, the invariant (2) may be positive or zero. Equations (1) present the neural net in its "natural" form in the sense that x_i and T_{ij} correspond to physical parameters: neuron potentials and synaptic interconnections, respectively. However, it is important to emphasize that the relationship between the invariants of the "vector" u_i and the "tensor" T_{ij} are not preserved by the coordinate transformation, i.e., equation (1) does not possess an invariant tensor structure. Consequently, the column u_i and the matrix T_{ij} cannot be treated as a vector and tensor, respectively.

In most applications, the neural nets performance is associated with convergence to attractors (pattern recognition, optimization, decision making, control, associative memory, generalization, etc.). The locations of attractors and their basins in phase space can be prescribed by an appropriate choice of the synaptic weights T_{ij} , i.e., by solving inverse dynamical problems. However, since dimensionality of neural nets is usually very high (in biological systems it is of order of 10^{11} with the number of synaptic interconnections of the order of 10^{15}), the straightforward analytical approach can be very expensive and time consuming. An alternative way to select synaptic weights in order to do specific tasks was borrowed from biological systems. It is based upon iterative adjustments of T_{ij} as a result of comparison of the net output with known correct answers (supervised learning) or as a result of creating of new categories from the correlations of the input data when correct answers are not known (unsupervised learning). Actually the procedure of learning is implemented by another dynamical system with the state variables T_{ij} which converges to certain attractors representing the desired synaptic weights.

Equation (1) represents a so called continuously updated neural net. Its discrete version is modeled by a corresponding contracting nonlinear map whose dynamical behavior, in principle, is similar to those of Eq. (1). In the simplest form such a map can be written in a McCulloch-Pitts form
(J. Hertz,1991):

$$x_i(t+1) = \operatorname{sgn} \sum T_{ij} x_j(t) \qquad \text{(I.1.2.4)}$$

where the sign function plays the role of the sigmoid function.

By replacing sgn in Eq.(4) with a stochastic rule:

$$x_i(t+1) = S \sum T_{ij} x_j(t) \qquad \text{(I.1.2.5)}$$

$$S = +1 \quad \text{with probability } f(\sum T_{ij} x_j) \qquad \text{(I.1.2.6)}$$

$$S = -1 \quad \text{with probability } 1 - f(\sum T_{ij} x_j)$$

one arrives at a stochastic version of neural nets, while the actual implementation of the stochastic rule (6) is still to be based upon a random number generator.

The basic limitation of deterministic or stochastic classical neurocomputers is in their restricted capacity which is proportional to the size of the computer. This limitation becomes obvious when neurocomputer is compared with a human brain: there are 10^{11} of parallel units in a human brain while neural chips made so far contain of the order of 10^4 units, which is too few for most practical applications, (J. Hertz,1991).

I.1.3. Quantum model of evolution.

A state of a quantum system is described by a special kind of time dependent vector $|\psi>$ with complex components called amplitudes:

$$\{a_0 a_1 ... a_n\} = |\psi> \qquad (I.1.3.1)$$

If unobserved, the state evolution is governed by the Schrödinger equation:

$$i\hbar \frac{da_k}{dt} = \sum_l H_{kl} a_l \qquad (I.1.3.2)$$

which is linear and reversible.

Here H_{kl} is the Hamiltonian of the system, $i = \sqrt{-1}$, $\hbar = 1.0545 \times 10^{-34} JS$.

The solution of Eq. (2) can be written in the following form:

$$\{a_0(t),..a_n(t)\} = \{a_0(0),..a_n(0)\}U^* \qquad (I.1.3.3)$$

where U is a unitary matrix uniquely defined by the Hamiltonian:

$$U = e^{-iHt/\hbar}, \qquad UU^* = I \qquad (I.1.3.4)$$

After m equal time steps Δt

$$\{a_0(m\Delta t),..a_n(m\Delta t)\} = \{a_0(0),..a_n(0)\}U^{*m} \qquad (I.1.3.5)$$

the transformation of the amplitudes formally looks like those of the transition probabilities in Markov chains. However, there is a fundamental difference between these two processes: in Eq. (5) the probabilities are represented not by the amplitudes, but by squares of their modules:

$$p = \{|a_0|^2,...|a_n|^2\} \qquad (I.1.3.6)$$

and therefore, the unitary matrix U is not a transition probability matrix.

It turns out that this difference is the source of so called quantum interference which makes quantum computing so attractive. Indeed, due to interference of quantum probabilities:

$$p = |a_1 + a_2|^2 \neq p_1 + p_2 \qquad (I.1.3.7)$$

each element of a new vector $a_i(m\Delta t)$ in Eq. (5) will appear with the probability $|a_i|^2$ which includes all the combinations of the amplitudes of the previous vector.

I.1.4. Quantum Collapse and Sigmoid Function.

As mentioned above, neural nets have two universal features: dissipativity and nonlinearity. Due to dissipativity, a neural net can converge to an attractor and this convergence is accompanied by a loss of information. But such a loss is healthy: because of it, a neural net filters out insignificant features of a pattern vector while preserving only the invariants which characterizes its belonging to a certain class of patterns. These invariants are stored in the attractor, and therefore, the process of convergence performs generalization: two different patterns which have the same invariants will converge to the same attractor. Obviously, this convergence is irreversible. The nonlinearity increases the neural net capacity: it provides many different attractors including static, periodic, chaotic and erogdic, and that allows one to store simultaneously many different patterns. Both dissipativity and nonlinearity are implemented in neural nets by the sigmoid (or squashing) function discussed in Section 2. It is important to emphasize that the only qualitative properties of the sigmoid function are those which are important for the neural net performance, but not any specific forms of this function**.** Can we find a qualitative analog of a sigmoid function in quantum mechanics? Fortunately, yes: it is so called quantum collapse which occurs as a result of quantum measurements. Indeed, the result of any quantum measurement is always one of the eigenvalues of the operator corresponding to the observable being measured. In other words, a measurement maps a state vector of the amplitudes (3.1) into an eigenstate vector

$$\{a_0 a_1 .. a_n\} \to \{00...\underset{\uparrow_i}{1}...00\} \qquad (I.1.4.1)$$

while the probability that this will be the i^{th} eigenvector is:

$$p_i = | a_i |^2 \qquad \text{(I.1.4.2)}$$

The operation (2) is nonlinear, dissipative, and irreversible, and it can play the role of a natural "quantum" sigmoid function.

I.1.5. QRN Architectures.

Let us introduce the following sequence of transformations for the state vector (3.1):

$$| \psi(0) > \to U | \psi(0) > \to \sigma_1 \{U | \psi(0) | \} = | \psi(t+1) > \qquad \text{(I.1.5.1)}$$

which is a formal representation of Eq.(4.1)) with σ_1 denoting a "quantum" sigmoid function.

In order to continue this sequence, we have to reset the quantum device considering the resulting eigenstate as a new input. Then one arrives at the following neural net:

$$a_i(t+1) = \sigma_1 \{ \sum U_{ij} a_j(t) \}, \qquad i = 1.2...n \qquad \text{(I.1.5.2)}$$

which has the form similar to Eq.(2.5). The curly brackets are intended to emphasize that σ_1 is to be taken as a measurement operation with the effect similar to those of a sigmoid function in classical neural networks (Fig. 1).

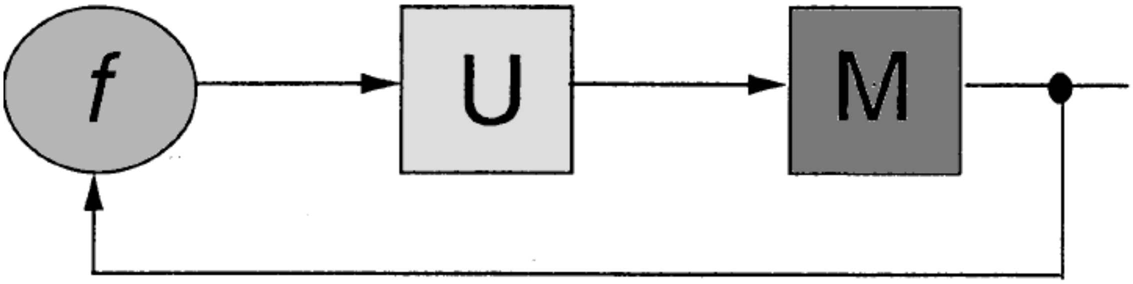

Figure 1. The simplest architecture of quantum neural net.

However, there are two significant differences between the quantum (5.2) and classical (2.5) neural nets. Firstly, in Eq. (5.2) the randomness appears in the form of quantum measurements as a result of the probabilistic nature of the quantum mechanics, while in (2.5) a special device generating random numbers is required. Secondly, if the dimension of the classical matrix T_{ij} is $N \times N$, then within the same space one can arrange the unitary matrix U (or the Hamiltonian H) of dimension $2^N \times 2^N$ exploiting the quantum entanglement and direct product decomposability of the Schrödinger equation (see Eq. (1.1). One should notice that each non-diagonal element of the matrix H may consist of two independent components: real and imaginary. The only constraint imposed upon these elements is that H is the Hermitian matrix, i.e.,

$$H_{ij} = \overline{H}_{ji} \qquad \text{(I.1.5.3)}$$

and therefore, the $n \times n$ Hermitian matrix has n^2 independent components.

So far the architecture of the neural net (5.2) was based upon one measurement per each run of the quantum device. However, in general, one can repeat each run for l times $l \le n$ collecting l independent measurements. Then, instead of the mapping (4.1), one arrives at the following best estimate of the new state vector:

$$\{a_0 ... a_n\} \to \{0 ... \underset{\uparrow_{i_1}}{\frac{1}{\sqrt{l}}} ... 0 ... \underset{\uparrow_{i_l}}{\frac{1}{\sqrt{l}}} ...\} \qquad \text{(I.1.5.4)}$$

while the probability that the new state vector has non-zero i_k^{th} component is

$$p_{ik} = | a_{ik} |^2 \qquad \text{(I.1.5.5)}$$

Denoting the sigmoid function corresponding to the mapping (5.4) as σ_l, one can rewrite Eq. (5.2) in the following form:

$$a_i(t+1) = \sigma_l\{\sum U_{ij} a_j(t)\}, \qquad i = 1.2...n \qquad \text{(I.1.5.6)}$$

The next step in complexity of the ORN architecture can be obtained if one introduces several quantum devices with synchronized measurements and resets:

$$a_i^{(1)}(t+1) = \sigma_{l_1 l_2}\{\sum U_{ij}^{(1)} a_j^{(1)}(t)\}, \quad i = 1.2...n_1 \qquad \text{(I.1.5.7)}$$

$$a_i^{(2)}(t+1) = \sigma_{l_2 l_1}\{\sum U_{ij}^{(2)} a_j^{(2)}(t)\}, \qquad i \neq 1.2...n_2 \qquad \text{(I.1.5.8)}$$

Here the sigmoid functions $\sigma_{l_1 l_2}$ and $\sigma_{l_2 l_1}$, map the state vectors into a weighted mixtures of the measurements:

$$\{a_1^{(1)}...a_n^{(1)}\} \to \frac{a_{11} a_{l_1}^{(1)} + a_{12} a_{l_2}^{(2)}}{|a_{11} a_{l_1}^{(1)} + a_{12} a_{l_2}^{(2)}|} \qquad \text{(I.1.5.9)}$$

$$\{a_1^{(2)}...a_n^{(2)}\} \to \frac{a_{21} a_{l_1}^{(1)} + a_{22} a_{l_2}^{(2)}}{|a_{21} a_{l_1}^{(1)} + a_{22} a_{l_2}^{(2)}|} \qquad \text{(I.1.5.10)}$$

where $a_{l_1}^{(1)}$ and $a_{l_2}^{(2)}$ are the result of measurements presented in the form (4), and a_{11}, a_{12}, a_{21} and a_{22} are constants.

Thus, Eqs. (7) and (8) evolve independently during the quantum regime, i.e., in between two consequtive measurements; however, during the measurements and resets they are coupled via the Eqs. (9) and (10). It is easy to calculate that the neural nets (2), (6) and (7), (8) operate with patterns whose dimensions are $n, n(n-1)(n-l), n_1(n_1-1)(n_1-l), n_2(n_2-1)(n_2-l)$, respectively.

In a more general architecture, one can have K-parallel quantum devices U_i with l_i consequtive measurements M_i for each of them *(i=1,2...k)*, see Fig. 2.

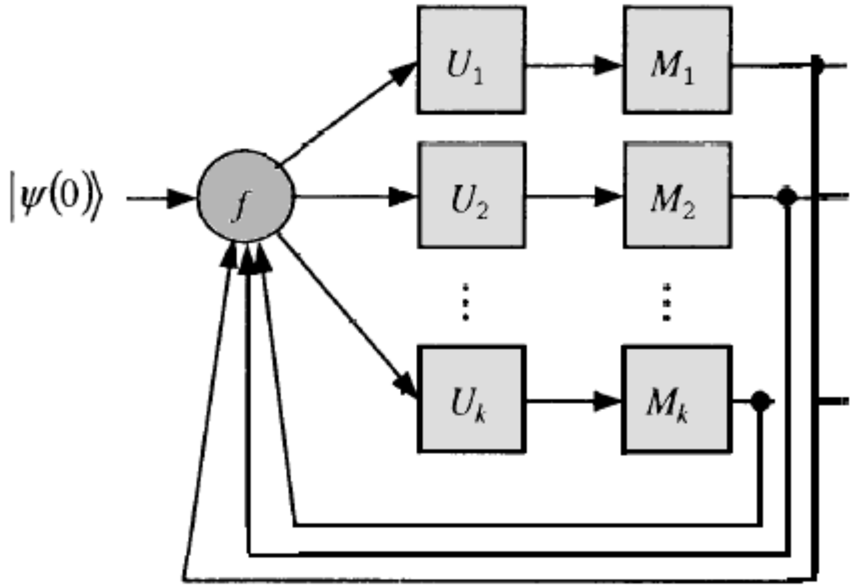

Figure 2. ***The k-Parallel Quantum Neural Network Architecture***

I.1.6. Maximum Likelihood Dynamics.

Let us turn to the simplest version of a quantum neural net (5.2), Fig. 1. As pointed out above, its performance is non-deterministic in a sense that each independent run of Eqs. (5.2) may lead to a different trajectory. However, in order to understand better the nonlinear structure of Eq. (5.2), we will introduce the best estimate, or the maximum likelihood trajectory by replacing the highest probability term in the output state by one. Choosing, for simplicity, a unitary matrix with real components:

$$U = \begin{pmatrix} 0.858726 & 0.387195 & -0.17004 & 0.289405 \\ 0.179855 & -0.801066 & 0.0518717 & 0.568555 \\ -0.362639 & 0.144341 & -0.832118 & 0.394003 \\ -0.314219 & 0.433058 & 0.525334 & 0.661628 \end{pmatrix} \qquad \text{I.1.6.1)}$$

one can verify that any initial state which is sufficiently close to the state $\{0001\}$ will be attracted to it, and therefore, the eigenstate $\{0001\}$ is a static attractor. In the same way one can find other static attractors, for instance

$$\{1000\},\{0100\},\{0010\},\frac{1}{\sqrt{2}}\{1100\},etc \qquad \text{(I.1.6.2)}$$

Another unitary matrix

$$U = \begin{pmatrix} -0.377565 & 0.554112 & -0.741892 \\ -0.70484 & 0.347627 & 0.618349 \\ -0.600537 & -0.756383 & -0.259309 \end{pmatrix} \qquad \text{(I.1.6.3)}$$

produces periodic attractors:

$$\{100\} \to \{010\} \to \{001\} \to \{100\} \quad etc \qquad \text{(I.1.6.4)}$$

Thus, relatively simple unitary matrices (1) and (3) within the framework of the quantum neural net (5.2) or (5.6), allows one to store several different patterns, namely: static patterns and periodically oscillating patterns. This means that in terms of the maximum likelihood dynamics, the quantum neural net behaves as a typical nonlinear system. However, the maximum likelihood dynamics cannot be identified with a deterministic dynamics. Indeed, if one runs Eq. (5.6) several times, all the solutions may be different from each other, so that with a small probability a pattern may converge to a "wrong" attractor; moreover, a pattern may wander between all five attractors performing a new stochastic paradigm. Strictly speaking such a "leak" from the deterministic performance of the maximum likelihood dynamics is a source of errors in the performance of a neural net. However, in many cases when neural net is expected to display certain flexibility by escaping a prescribed paradigm, this leak may create a useful emerging behavior.

In order to evaluate deviations from the maximum likelihood solution, one has to turn to the probabilistic description of solutions to Eqs. (5.2) and (5.6).

I.1.7. Evolution of probabilities.

Let us take another look at Eq. (5.2). Actually it performs a mapping of an i^{th} eigenvector into an j^{th} eigenvector:

$$\underset{\uparrow_i}{\{00...010...0\}} \to \underset{\uparrow_j}{\{00...010...0\}} \qquad \text{(I.1.7.1)}$$

The probability of the transition (1) is uniquely defined by the unitary matrix *U*:

$$p_{ij} = |U_{ji}|^2, \qquad \sum_{i=1}^{n} p_{ij} = 1 \qquad \text{(I.1.7.2)}$$

and therefore the matrix $\| p_{ij} \|$ plays the role of the transition matrix in a generalized random walk which is represented by the chain of mapping (1).

Thus, the probabilistic performance of Eq. (5.2) has remarkable features: it is quantum (in a sense of the interference of probabilities) in between two consecutive measurements, and it is classical in description of the sequence of mapping (1). Applying the transition probability matrix (2) and starting, for example, with eigenstate $\{10...0\}$, one obtains the following sequence of the probability vectors:

$$\pi_0 = \{10...0\}; \quad \pi_1 = \{10...0\}\begin{pmatrix} p_{11}\cdots p_{11} \\ .... \\ p_{n1}\cdots p_{nn} \end{pmatrix} = \{\pi_1{}^1 ... \pi_n{}^1\}; etc \qquad \text{(I.1.7.3)}$$

An i^{th} component of the vector $\pi_m, i.e \quad \pi_m{}^i$ expresses the probability that the system is in the i^{th} eigenstate after m steps. As follows from Eqs. (3), the evolution of probabilities is a linear stochastic process, although each particular realization of the solution to Eq. (5.2) evolves nonlinearly, and one of such realization is the maximum likelihood solution considered in the previous section. In this context, the probability distribution over different particular realizations can be taken as a measure of possible deviations from the best estimate solution. However, the stochastic process (3) as an ensemble of particular realizations, has its own invariant characteristics which can be expressed independently on these realizations. One of such characteristics is the probability $f_{ij}{}^{(m)}$ that the transition from the eigenstate *i* to the eigenstate *j* is performed in *m* steps. This characteristic is expressed via the following recursive relationships, M. Bartlett, 1956:

$$f_{ij}{}^{(1)} = p_{ij}{}^{(1)} = p_{ij}, f_{ij}{}^{(2)} = p_{ij}{}^{(2)} - f_{ij}{}^{(1)} p_{ij}$$
$$f_{ij}{}^{(m)} = p_{ij}{}^{(m)} - f_{ij}{}^{(1)} p_{jj}{}^{(n-1)} - f_{ij}{}^{(3)} p_{jj}{}^{(n-2)} ... - f_{ij}{}^{(n-1)} p_{jj}. \qquad \text{(I.1.7.4)}$$

If

$$\sum_{m=1}^{\infty} f_{ij}{}^{(m)} < 1 \qquad \text{(I.1.7.5)}$$

then the process initially in the eigenstate *i* may never reach the eigenstate *j*.

If $\sum_{m=1}^{\infty} f_{ij}{}^{(m)} = 1$ (I.1.7.6)

then the i^{th} eigenstate is a recurrent state, i.e., it can be visited more than once. In partiicular, if

$$p_{ii} = 1 \qquad \text{(I.1.7.7)}$$

this recurrent state is an absorbing one: the process will never leave it once it enters.

From the viewpoint of neural net performance, any absorbing state represents a deterministic static attractor without a possibility of "leaks." In this context, a recurrent, but not absorbing state can be associated with a periodic or an aperiodic (chaotic) attractor. To be more precise, an eigenstate *I* has a period τ $(\tau > 1)$ *if* $p_{ii}{}^{(m)} = 0$ whenever *m* is not divisible by τ, and τ is the largest integer with this property. The eigenstate is aperiodic

if $\tau = 1$ (I.1,7.8)

Another invariant characteristic which can be exploited for categorization and generalization is reducibility, i.e., partitioning of the states of a Markov chain into several disjoint classes in which motion is trapped. Indeed, each hierarchy of such classes can be used as a set of filters which are passed by a pattern before it arrives at the smallest irreducible class whose all states are recurrent. For the purpose of evaluation of deviations (or "leaks") from the maximum likelihood solution, long-run properties of the evolution of probabilities (3) are important. Some of these properties are known from theory of Markov chains, namely: for any irreducible erogodic Markov chain the limit $p_{ij}{}^{(m)}$ exists and it is independent of *I*, .e.,

$$\lim p_{ij}{}^{(m)} = \pi_i \quad at \quad m \to \infty \qquad \text{(I.1.7.9)}$$

while

$$\pi_j > 0, \quad \pi_j = \sum_{i=0}^{k} \pi_i p_{ij}, \quad j = 0,1,..k, \qquad \sum_{j=}^{k} \pi_j = 1, \qquad \pi_j = \frac{1}{\mu_{ii}} \qquad \text{(I.1.7.10)}$$

Here μ_{ii} is the expected recurrence time

$$\mu_{ii} = 1 + \sum_{l \neq j} p_{ij}\mu_{li} < \infty \qquad \text{(I.1.7.11)}$$

The definition of ergodicity of a Markov chain is based upon the conditions for aperiodicity (8) and positive recurrence (9), while the condition for irreducibility requires existence of a value of *m* not dependent upon *i* and *j* for which $p_{ij}^{(m)}$>0 for *i* and *j*. The convergence of the evolution (3) to a stationary stochastic process suggests additional tools for information processing. Indeed, such a process for n-dimensional eigenstates can be uniquely defined by *n* statistical invariants (for instance, by first *n* moments) which are calculated by summations over time rather than over the ensemble, and for that a single run of the quantum net (5.2) is sufficient. Hence, triggered by a simple eigenstate, a prescribed by n-invariants stochastic process can be retrieved and displayed for the purposes of Monte-Carlo computations, for modelling and prediction of behavior of stochastic systems, etc.
Continuing analysis of evolution of probability, let us introduce the following difference equation

$$\pi_i(t+\tau) = \sum_{j=1}^{n} \pi_j(t)p_{ij}, \qquad \sum_{i=1}^{n} \pi_i = 1, \qquad \pi_i \geq 0, \quad i = 1,2,...n \qquad \text{(I.1.7.12)}$$

It should be noticed that the vector $\pi = (\pi_1,..\pi_n)$ as well as the stochastic matrix p_{ij} exist only in an abstract Euclidean space: they never appear explicitly in physical space. The evolution (12) is also irreversible, but it is linear and deterministic.

The only way to reconstruct the probability vector $\pi(t)$ is to utilize the measurement results for the vector a(t). In general case, many different realizations of Eq. (1) are required for that purpose. However, if the condition (5) holds, the ergodic attractor $\pi = \pi^{\infty}$ can be found from the only one realization due to the ergodicity of the stochastic process. The ergodic attractor π^{∞} can be found analytically from the steady-state equations:

$$\pi_i^{\infty} = \sum_{j=1}^{n} p_{ij}\pi_j^{\infty}, \ \sum_{i=1}^{n} \pi_i^{\infty} = 1, \qquad \sum_{j=1}^{n} p_{ij} = 1, \qquad \pi_i = 1, \quad p_{ij} = 0 \qquad \text{(I.1.7.13)}$$

This system of *n+1* equations with respect to *n* unknowns $\pi_i^{\infty}(=1,2,..n$ has a unique solution.

As an example, consider a two-state case (*n=2):*

$$p_{11}\pi_1^{\infty} + p_{21}\pi_2^{\infty} = \pi_1^{\infty}, \qquad p_{12}\pi_1^{\infty} + p_{22}\pi_2^{\infty} = \pi_2^{\infty} \qquad \text{(I.1.7.14)}$$

Utilizing the constraints in Eqs. (13) one obtains:

$$\pi_1^{\infty} = \frac{1-p_{22}}{2-(p_{11}+p_{22})}, \qquad \pi_2^{\infty} = \frac{1-p_{11}}{2-(p_{11}+p_{22})} \qquad \text{(I.1.7.15)}$$

Hence on the first sight, there are infinite numbers of unitary matrices u_{ij} which provide the same ergodic attractor. However, such a redundancy is illusive since the fact that the stochastic matrix p_{ij} has been derived from the unitary matrix u_{ij} impose a very severe restriction upon p_{ij}: not only the sum of each row, but also the sum of each column is equal to one, i.e., now in addition to the constrain in Eqs. (13), an additional constraint

$$\sum_{i=1}^{n} p_{ij} = 1 \qquad \text{(I.1.7.16)}$$

is imposed upon the stochastic matrix. This makes this matrix doubly stochastic that always leads to an ergodic attractor with uniform distribution of probabilities. Obviously such a property significantly reduces the usefulness of the Quantum recurrent nen (QRN). However, as will be shown below, by slight change of the QRN architecture, the restriction (16) can be removed.

I.1.8. Multivariate ONR.

In the previous section we have analyzed the simplest quantum neural net whose probabilistic performance was represented by a single-variable stochastic process equivalent to generalized random walk. In this section we will turn to multi-variable stochastic process and start with the two-measurement architecture. Instead of Eq.(14) now we have the following mapping:

$$\frac{1}{\sqrt{2}}\{00...1_{i_1}0...1_{i_2}0...0\} \to \frac{1}{\sqrt{2}}\{00...1_{j_1}0...1_{j_2}0...0\} \qquad \text{(I.1.8.1)}$$

i.e., $$I_1 + I_2 \to J_1 + J_2 \qquad \text{(I.1.8.2)}$$

where I_1, I_2, J_1 and J_2 are the eigenstates with the unit *1* is at the $i_1^{th}, i_2^{th}, j_1^{th}$ and j_2^{th} places, respectively. Then the transitional probability of the mappings:

$$p_{i_1 i_2}{}^{j_1}(I_1 + I_2 \to J_1) = \frac{1}{2}|U_{j_1 i_1} + U_{j_1 i_2}|^2 \qquad \text{(I.1.8.3)}$$

$$p_{i_1 i_2}{}^{j_2}(I_1 + I_2 \to J_2) = \frac{1}{2}|U_{j_1 i_1} + U_{j_1 i_2}|^2 \qquad \text{(I.1.8.4}$$

Since these mapping result from two independent measurements, the joint transitional probability for the mapping (1) is

$$p_{i_1 i_2}{}^{j_1 j_2}(I_1 + I_2 \to J_1 + J_2) = \frac{1}{2}|U_{j_1 i_1} + U_{j_1 i_2}|^2 |U_{j_2 i_1} + U_{j_2 i_2}|^2 \qquad \text{(I.1.8.5)}$$

One can verify that

$$\sum_{j=1}^{n} p_{i_1 i_2}{}^{j} = 1, \qquad \sum_{j_1 j_2 = j}^{n} p_{i_1 i_2}{}^{j_1 j_2} = 1, \qquad \text{(I.1.8.6)}$$

It should be emphasized that the input patterns *I* interfere, i.e., their probabilities are added according to the quantum laws since they are subjected to a unitary transformation in the quantum device. On the contrary, the output patterns *J* do not interfere because they are obtained as a result of two independent measurements. As mentioned above, Eq. (5) expresses the joint transition probabilities for two stochastic processes

$$I_1 \to J_1 \text{ and } I_2 \to J_2 \qquad \text{(I.1.8.7)}$$

which are coupled via the quantum interference. At the same time, each of the stochastic processes (5) considered separately has the transition probabilities following from Eq. (7.2), and by comparing Eqs. (7.2) and Eq. (5), one can see the effect of quantum interference for input patterns.

It is interesting to notice that although the probabilities in Eqs. (5) have a tensor structure, strictly speaking they are not tensors. Indeed, if one refers the Hamiltonian *H*, and therefore the unitary matrix *U* to a different coordinate system, the transformations of the probabilities (5) will be different from those required for tensors. Nevertheless, one can still formally apply the chain rule for evolution of transitional probabilities, for instance:

$$p_{i_1 i_2 q_1 q_2}(I_1 + I_2 \to J_1 + J_2 \to Q_1 + Q_2) = p_{i_1 i_2 j_1 j_2} p_{j_1 j_2 q_1 q_2} etc \qquad \text{(I.1.8.8)}$$

Eqs. (5) is easily generalized to the case of *l* measurements $l \le n$:

$$p_{i_1 \ldots i_l q_1 \ldots q_l} = p_{i_1 \ldots i_l j_1 \ldots j_l} p_{j_1 \ldots j_l q_1 \ldots q_l} \ etc$$

(I.1.8.9)

$$p_{i_1 \ldots i_l j_1 \ldots j_l} = \frac{1}{l^l} \prod_{\alpha=1}^{l} | \sum_{\beta=1}^{l} U_{j_\alpha i_\beta} |^2$$

(I.1.8.10)

Now the evolution in physical space, instead of Eq. (I.5.2)), is described by the following:

$$a_i(t+\tau) = \sigma_l\{\sum U_{ij} a_j(t)\}, \qquad i = 1.2 \ldots n \qquad \text{(I.1.8.11)}$$

where σ_l is the *l*-measurements operator.

Obviously, the evolution of the state vector a_i is more "random" than those of Eq. (I.5.2) since the corresponding probability distribution depends upon *l* variables.

Eq. (11) can be included in a system with interference inputs and independent outputs as a generalization of the system (5.7),(5.8).

I.1.9. QRN with input interference.

In order to remove the restriction (7.16), let us turn to the architecture shown in Fig. 1 and assume that the result of the measurement, i.e., a unit vector $a_m(t) = \{00...010...0\}$ is combined with an arbitrary complex (interference) vector, Fig. 3.

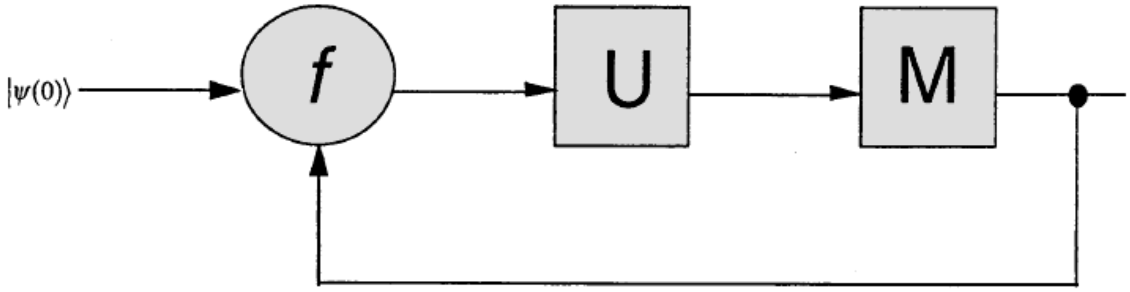

Figure 3. QRN with input interference.

is combined with an arbitrary complex vector *m*:

$$m = \{m_1, ..m_n\} \qquad \text{(I.1.9.1)}$$

such that

$$a(t) = [a_m(t) + m]c, \qquad c = \frac{1}{m_1^2 + ...(m_i + 1)^2 + ..m_n^2} \qquad \text{(I.1.9.2)}$$

Then the transition probability matrix becomes

$$p_{ij} = \frac{|U_{j1}m_1 + ...U_{ji}(m_i + 1) + ..U_{jn}m_n|^2}{|m_1^2 + ...(m_i + 1)^2 + ..m_n^2|} \qquad \text{(I.1.9.3)}$$

Thus, now the structure of the transition probability matrix p_{ij} can be controlled by the interference vector m.

Eq. (3) is derived for a one-dimensional stochastic process, but its generalization to *l*-dimensional case is straight-forward.

I.1.10. More complex archtectures.

In order to clarify the more complex architectures of (QRN), for instance, such as those given by Eqs. (5.9) and (5.10), turn to Eq. (8.5), and consider the tensor $p_{i_1 i_2 j_1 j_2}$. By simple manipulation of indices one obtains:

$$p_{i_1 i_2 j_1 j_2} \pi_{j_1} \pi_{j_2} = \pi_{i_1 i_2} \qquad \text{(I.1.10.1)}$$

The products $\pi_{j_1} \pi_{j_2}$ and $\pi_{i_1} \pi_{i_2}$ represent the components of the direct product of two vectors $\pi_1 \otimes \pi_2$ and therefore, Eq. (1) can be rewritten as:

$$[\pi_1 \otimes \pi_2]_{t+\tau} = p_{12}[\pi_1 \otimes \pi_2]_t \qquad \text{(I.1.10.2)}$$

where p_{12} is the tensor with the components $p_{i_1 i_2 j_1 j_2}$, and $\pi_1 \pi_2$ represents probability vectors for two different stochastic processes coupled via quantum interference (see Eq. (8.5)).

In order to understand the physical meaning of Eq. (2), start with a simpler case when two stochastic processes $\pi_1 \pi_2$ are considered separately. Then each vector evolves according to the following equation:

$$\pi_i(t+\tau) = p_i^{\,j}\pi_j(t) \qquad \text{(I.1.10.3)}$$

Moreover, if these processes are coupled in a “quantum” sense, one arrives at a simultaneous nonlinear system:

$$\pi_i^{(1)}(t+\tau) = p_{i_1 j_\cdot}\pi_{j_1}(t) \qquad \text{(I.1.10.4)}$$

$$\pi_i^{(2)}(t+\tau) = p_{i_2 j_2}\pi_{j_2}(t) \qquad \text{(I.1.10.5)}$$

It should be emphasized that the matrices $p_{i_1 j_1}$ $p_{i_2 j_2}$ are not constant: they depend both upon the components of π_1 and π_2 (see Eq. (8.5)). (This dependence is expressed by the mixed terms $U_{j_1 j_2}$ and $U_{j_2 j_1}$). Obviously, the vectors π_1 and π_2 now represent the conditional probabilities.

Thus, due to the quantum interference, the stochastic vectors π_1 (given π_2) and π_2 (given π_1) evolve nonlinearly. However, the direct product $\pi_1 \otimes \pi_2$ which can be associated with the joint probability, evolve linearly according to Eq. (8.5). (We should notice again that for more complex architectures of the type (8.10), the joint probability may not exist). Such a result is not surprising: as we saw above, even for the simplest architecture (5.2), the maximum likelihood (deterministic) solution evolve nonlinearly, while the probability of the process evolve linearly. Analogously, in complex architectures, both maximum liklihood and conditional probability processes evolve nonlinearly, but the joint probabilities evolve linearly.

Eq. (2) can be generalized for *l*-measurement architectures as following

$$[\pi_1 \otimes \pi_2 \otimes \ldots \otimes \pi_l]_{t+\tau} = p_{1,2,\ldots l}[\pi_1 \otimes \pi_2 \otimes \ldots \otimes \pi_l]_t \qquad \text{(I.1.10.6)}$$

I.1.11. Non-Markovian processes.

The quantum neural nets (5.2) or (5.6), with a slight modification, can generate non-Markovian processes that are “more deterministic” because of higher correlations between values of the vector a_i at different times, i.e., between $a_i(t), a_i(t-\tau), a_i(t-2\tau)$, etc. Indeed, let us assume that each new measurement is combined with the *l* previous measurements (instead of *l* repeated measurements). Then Eq. (8.10) will express the joint distribution of, $a_i(t), a_i(t-\tau), a_i(t-2\tau)$ etc. The evolution of these probabilities is described by the equation following from (10.6):

$$\begin{aligned} &[\pi(t)\otimes\pi(t-\tau\otimes\ldots\otimes\pi(t-l\tau]_{t+\tau} = \\ &= p_{1,2,\ldots l}[\pi(t-\tau)\otimes\pi(t-2\tau)\otimes\ldots\otimes\pi[t-(l+1)\tau]_t \end{aligned} \qquad \text{(I.1.11.1)}$$

Thus, instead of *l*-dimensional Markov process in (10.9), now we have a one-dimensional non-Markovian process of the *l*th order.

By combining l_1 new measurements with l_2 previous measurements, one can generate an l_1-dimensional non-Markovian process of the $l_2^{\,th}$ order.

I.1.12. Nonlinear QRN.

So far all the stochastic processes considered above were linear. Now let us assume that along with the Eq. (5.2) that is implemented by quantum device, we implement (in a classical way) the associated probability equation (10.6). At this point these two equations are not coupled yet. Now turning to Eqs. (9.1)-(9.3), assume that the role of the interference vector *m* is played by the probability vector π. Then Eqs. (5.2) and (10.6) take the form:

$$a_i(t+1) = \sigma_1\{\sum U_{ij}a_j(t)\}, \qquad i = 1.2\ldots n \qquad \text{(I.1.12.1)}$$

$$\pi_i(t+\tau) = \sum p_{ij}\pi_j(t), \quad i{=}1,2,\ldots n \qquad \text{(I.1.12.2)}$$

where $a_i(t) = [\{00\ldots010\ldots0\} + \{\pi_1\pi_2\ldots\pi_n\}]C$ (I.1.12.3)

$$C = \frac{1}{\pi_1^{\ 2} + ...(\pi_i + 1)^2 + ... + \pi_n^{\ 2}} \qquad (I.1.12.4)$$

$$p_{ij} = \frac{|U_{j_1}\pi_1 + ...U_{j_i}(\pi_i + 1) + ...U_{j_n}\pi_n|^2}{|\pi_1^{\ 2} + ...(\pi_i + 1)^2 + ... + \pi_n^{\ 2}|} \qquad (I.1.12.5)$$

and they are coupled. Moreover, the probability evolution (2) becomes nonlinear since the matrix p_{ij} depends upon the probability vector π.

Remark. One can associate Eq. (1) with the equation of motion in physical space, and Eq. (2) – with the Liouville equation describing the evolution of an initial randomness in a probability (virtual) space. In QNR architecture, Eq. (1) is always nonlinear (due to quantum collapse), while Eq. (2) is linear unless it is coupled with Eq. (1) via the feedback (3). Therefore, one arrives at two fundamentally different dynamical topologies of QRN: the first one is linked to Newtonian physics where equation of motion is never coupled with the corresponding Liouville equation, and the second one can be linked to quantum physics (in the Madelung version of the Schrödinger equation) where the Hamilton-Jacobi equation is coupled with the corresponding Liouville equation by the quantum potential. In the next part of this book we will exploit the second type of dynamical topology, and we will call it quantum-inspired (iQ) since, in general, the feedback between the equation of motion and the Liouville equation will be different from the quantum potential.

I.1.13. Spontaneous self-organization.

In this section we will demonstrate a relation of non-linear QRN considered above to a concept of a spontaneous self-organization as a component of life and intelligence. As shown in section 7, a linear QRN eventually approaches an attractor in probability space (see Eq. (I.1.7.15) that represents a stationary stochastic process, and this attractor does not depends upon initial conditions. Therefore, from the viewpoint of information processing, this attractor performs generalization by placing all possible entry patterns in the same class. Let us ask now the following question: can the system (7.14) change its evolution, and consequently, its limit distribution, without any external "help"? The formal answer is definitely positive. Indeed, if the transition matrix depends upon the current probability distribution

$$p = p(\pi) \qquad (I.1.13.1)$$

then the evolution (7.14) becomes nonlinear, and it may have many different scenarios depending upon the initial state π^0. In particular case (7.12), it can "overcome" the second law of thermodynamics decreasing its final entropy by using only the "internal" resources. The last conclusion illuminates the Schrödinger's statement that 'life is to create order in the disordered environment against the second law of thermodynamics". Indeed, suppose that the selected unitary matrix is

$$U = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} \qquad (I.1.13.2)$$

Then the corresponding transition probability matrix in Eq. (7.12), according to Eq. (7.2) will be doubly-stochastic:

$$p = \begin{pmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{pmatrix} \qquad (I.1.13.3)$$

and the stochastic process (7.12) is already in its thermodynamics limit (3), i.e.,

$$\pi_1 = \pi_2 = 1/2 \qquad (I.1.13.4)$$

Let us assume that the objective of the system is to approach the deterministic state

$$\pi_1 = 1, \quad \pi_2 = 0 \qquad (I.1.13.5)$$

without help from outside. In order to do that, the system should adapt a feedback (1) in the form:

$$a = (a_1, a_2), \qquad a_1 = -2\pi_1, \qquad a_2 = 1 \qquad (I.1.13.6)$$

Then, according to Eqs. (7.2), the new transition probability matrix □ will be:

$$p_{11} = \frac{\pi_1^{\ 2}}{2\pi_1^{\ 2} - 2\pi_1 + 1} \qquad p_{12} = \frac{(1-\pi_1)^2}{2\pi_1^{\ 2} - 2\pi_1 + 1}$$

$$p_{21} = \frac{(1+\pi_1)^2}{2\pi_1^{\ 2} + 2} \qquad p_{22} = \frac{(1-\pi_1)^2}{2\pi_1^{\ 2} + 2} \qquad (I.1.13.7)$$

Hence, the evolution of the probability π_1 now can be presented as:

$$\pi_1^{(n+1)} = \pi_1^{(n)} p_{11} + (1 - \pi^{(n)}) p_{21} \qquad \text{(I.1.13.8)}$$

in which $p_{11}$and p_{22} are substituted from Eqs. (7).

It is easily verifiable that

$$\pi_1^{\infty} = 1, \ \pi_2^{\infty} = 0 \qquad \text{(I.1.13.9)}$$

i.e., the objective is achieved due to the "internal" feedback (6).

As follows from Eqs.(5) and (9), due to the built-in feedback (6) and without any external effort, the system moved from the state of maximum entropy to the state of minimum entropy, and that violates the second law of thermodynamics. This means that such a system does not belong to physical world: it is neither a Newtonian nor a quantum one. But to what world does it belong? Let us recall again the Schrödinger statement (Schrödinger. 1945): "life is to create order in the disordered environment *against the second law of thermodynamics*". That gives a hint for exploiting the effect of self-organization for modeling some aspects of life, and we return to this subject in the second part of this book in connection with quantum-inspired computing. The application of QRN-based self-organization model to common sense decision making process has been introduced in Zak,M., 2000. In the second part of this book, based upon the paradigm of spontaneous self-organization, a new class of dynamical systems called quantum-inspired (iQ), or ***self-supervised*** will be introduced. It will be shown that these systems are neither Newtonian nor quantum (although they have a quantum topology), and that they demonstrate properties similar to those of intelligent systems.

I.1.14. Summary.

There are two broad areas in which classical recurrent nets become very effective: The associative memory and optimization. In this section we will present a brief description of additional advantages that can be expected from QRN as a generalization of classical neural nets.

The problem of associative memory is formulated as following: store a set of *q n*-dimensional patterns $\xi_i^{\eta} (\eta = 1,2,...q; \quad i = 1,2,...n$ as a dynamical attractor; if a new pattern ζ_i presented as an input is sufficiently close to a particular pattern $\tilde{\xi}_i^{\eta}$, i.e., it belongs to the basin of the corresponding attractor, it will trigger a dynamical process which eventually converges to the sample pattern $\tilde{\xi}_i^{\eta}$. From the viewpoint of information processing, such a convergence can be interpreted not only as an associative memory, but also as a pattern recognition, identification, classification etc. However, the most important part of this process which distinguishes neural nets from other computational tools is generalization. Indeed, the convergence of the solution to the attractor is a dissipative process: it is accompanied by the loss of unnecessary information. Only invariants that characterize the belonging of a pattern to a certain class survives this loss, and they are represented by the attractor.

The fundamental problem in associative memory is to find such a synaptic interconnections T_{ij} (see Eq.(2.1)), or, in case of a quantum implementation, the Hamiltonian *H*, that provides a prescribed number of attractors of certain type and at certain locations.

In optimization performance the problem is inverse: the matrix T_{ij} (or H_{ij}) is given, and the neural net must converge to an attractor which represent a minimum to a certain function (or functional) formulated in terms of the matrices T_{ij} or H_{ij}.

There are several advantages that can be expected from quantum implementation of recurrent nets. Firstly, since the dimension of the unitary matrix *n* can be exponentially larger within the same space had it been implemented by a quantum device, the capacity of quantum neural nets in terms of the number of patterns stored as well as their dimensions can be exponentially larger too.

Secondly, QRN have a new class of attractors representing different stochastic processes, which in terms of associated memory, can store complex behaviors of biological and engineering systems, or in terms of optimization, to minimize a functional whose formulation includes statistical invariants.

The details of ORN performance in learning, optimization, associative memory, as well as in generation of stochastic processes can be found in Zak, M.,1998, 1999.

In this chapter the attention was focused on the most remarkable property of nonlinear QRN that is associated with the spontaneous self-organization as a possible bridge to model intelligent behavior. It is important to emphasize that the architecture of that ORN includes a built-in feedback from the probability evolution to the evolution of the state vector, and that leads to such a non-Newtonian property as transition from a disorder to the order without any external interference. In the next part of the book, this property will play a fundamental role in quantum-inspired models.

Another unique property of QRN is based upon quantum interference of probabilities. Due to this interference, the stored patterns acquire a logical structure in a sense that each combination of patterns has a qualitatively new meaning in the same way in which combinations of letters forming words do. This property has a very interesting philosophical consequence. Indeed, it was always difficult to understand how biological neural nets can learn patterns of external world without any preliminary structure built-in to their synaptic interconnections. The experience with artificial neural nets shows that training without a preliminary structure is exponentially longer than those with a structure, and that poses the following question: who created the "first" structure in biological neural nets which provides the ability to learn and select useful properties in polynomial time? In other words, can natural selection act without a "creator"? The QRN may give a positive answer to this question: the logical structure of synaptic interconnections can be imposed by natural laws of physics, and in particular, by quantum mechanics. Hence, if biological neural nets utilize quantum effects in their performance, they can learn the model of the external world, including its logical structure, in polynomial time without any preliminary structure. More details on that subject are presented in the next chapter.

Chapter I.2.
Quantum model of emerging grammars

Give me a laundry list and I'll set it to music.
G.A.Rossini.

Abstract.

In this Chapter, as an application of the theory discussed in the previous Chapter, a special class of QRN simulating Markov chains with absorbing states is introduced. The absorbing states are exploited for pattern recognition: each class of patterns is attracted to a unique absorbing state. Due to quantum interference of patterns, each combination of patterns acquires its own meaning: it is attracted to a certain combination of absorbing states which is different from those of individual attractions. This fundamentally new effect can be interpreted as formation of a grammar, i.e., a set of rules assigning certain meaning to different combinations of patterns. It appears that there exists a class of unitary operators in which each member gives rise to a different artificial language with associated grammar.

I.2.1. General remarks.

One of the oldest and most challenging problems is to understand the process of language formation. In this section we will introduce a model of grammar formation based upon a unique property of QRN: the pattern interference. Let us assume that we store letters of the alphabet in the form of the corresponding stochastic attractors ξ_η . Then if some of these letters, say $\xi_{\eta_1}...\xi_{\eta_l}$., are presented to the QRN simultaneously, their processing will be accompanied by quantum interference in such a way that they will converge to a new attractor, say $\xi_{1,2,...l}$. This new attractor preserves the identities of the letters $\xi_{\eta_1}...\xi_{\eta_l}$, but at the same time, it is not a simple sum of these letters. Moreover, any additonal letter $\xi_{\eta_{l+1}}$ may create a totally different new attractor $\xi_{1,2,...l,l+1}$. Actually this phenomenon is similar to formation of words from letters, sentences from words, etc. In other words, the pattern interference creates a grammar by giving different meaning to different combinations of letters. However, although this grammar is imposed by natural laws of quantum mechanics, it can be changed. Indeed, by changing phases of the components H_{ij} of the Hamiltonian, one changes the way in which the patterns interfere and therefore, the "English" grammar can be transformed into "French" grammar etc.

It should be recalled that the ability to create and understand language is the fundamental property of intelligence that distinguishes human from other livings. At this stage, we do not have any evidence that Nature exploits this particular quantum phenomenon for emerging grammars, but we do not yet observe any alternative ways either. Therefore it is safe to apply QRN for modeling artificial intelligent agents like robots rather than human.

In this section, based upon a concept of QRN, a new phenomenological formalism for pattern recognition and grammar formation is described.

I.2.2. Emerging grammar formalism.

We will start with a QRN that augmented with a classical measurement and quantum reset operation. The design of the one-dimensional version of this network is shown in Fig. 3. An initial state, $|\psi(0)>$, is fed into the network, transformed under the action of a unitary operator, *U*, subjected to a measurement indicated by the measurement operator M{ }, and the result of the measurement is used to control the new state fed back into the network at the next iteration. One is free to record, duplicate or even monitor the sequence of measurement outcomes, as they are all merely bits and hence constitute classical information. Moreover, one is free to choose the function used during the reset phase, including the possibility of adding no offset state whatsoever. Such flexibility makes the QRN architecture remarkably versatile. To simulate a Markov process, it is sufficient to return just the last output state to the next input at each iteration. For a proof-of-concept, we will start with the following unitary N-dimensional operator

$$U = \begin{pmatrix} U_{11}U_{12}...U_{1n}0...0 \\ U_{12}U_{22}...U_{2n}0...0 \\ .... \\ U_{n1}U_{n2}...U_{nn}0...0 \\ 0..........010...0 \\ .... \\ 0................01 \end{pmatrix} \qquad n < N \qquad \text{(I.2.2.1)}$$

that maps the i^{th} eigenvector into a j^{th} eigenvector

$$\{00...\underset{\uparrow_i}{0}10...0\} \rightarrow \{00...0\underset{\uparrow_j}{1}0...0\} \qquad \text{(I.2.2.2)}$$

with the probability

$$p_{ij} = |U_{ji}|^2 \qquad \text{(I.2.2.3)}$$

(See Eq. (1.7.12))

Eq. (3) is modified to the following (see Eq. (1.9.3)

$$p_{ij} = \frac{|\sum_{k=1}^{n} U_{jk}a_k + U_{ji}|^2}{|\sum_{k \neq i}^{n} a_k{}^2 + (a_i + 1)^2|} \qquad \text{(I.2.2.4)}$$

if each result of the measurement is combined with an arbitrary offset vector

$$|\psi' >= \{a_1...a_n\} \qquad \text{(I.2.2.5)}$$

It should be emphasized that the sum of the output vector in (2) and the offset vector (5) is first calculated, normalized, and then the corresponding quantum re-entering state is prepared.

For the purpose of pattern recognition, the offset vector will be chosen as follows:

$$|\psi'_0 >= \begin{cases} \{a_1, a_2, ...a_N\} & if \quad i \leq n \\ 0 & if \quad i > n \end{cases} \qquad \text{(I.2.2.6)}$$

where *i* is defned by Eq. (2).

Now the probability of the mapping (2) performed by the unitary operator U and the offset vector (6) can be obtained by combining Eqs. (3) and (4), and the transition matrix for the corresponding Markov chain is

$$p = \begin{pmatrix} p_1^{\,1} \cdots p_1^{\,n} p_1^{\,n+1} \cdots \quad p_1^{\,N} \\ p_2^{\,1} \cdots p_2^{\,n} p_2^{\,n+1} \cdots \quad p_2^{\,N} \\ \cdots\cdots\cdots\cdots\cdots\cdots \\ p_n^{\,1} \cdots p_n^{\,n} p_n^{\,n+1} \ldots\ldots p_n^{\,N} \\ 0\ldots\ldots0 \quad 10\ldots\ldots\ldots0 \\ \cdots\cdots\cdots\cdots\cdots\cdots \\ 0\ldots\ldots\ldots\ldots\ldots\ldots\ldots1 \\ \end{pmatrix}, \qquad 0 \le p_i^{\,j} < 1, \qquad \sum_{j=1}^{N} p_i^{\,j} = 1 \qquad \text{(I.2.2.7)}$$

This chain has *n* transient states $T_q(q = 1.2,..n)$ and *N-n* absorbing states $A_\gamma(\gamma = n+1, n+2,...N)$ and therefore, regardless of an initial state, the stochastic process eventually will be trapped in one of the absorbing states A_k . However, the probability that it will be a prescribed state A_γ depends upon the initial state. Indeed, as follows from theory of Markov chains, the probability $f_q^{\,k}$ of absorption into A_k from T_q satisfies the system of equations, Feller, W., 1957,

$$f_q^{\,k} = \sum_{j=1}^{n} p_q^{\,j} f_j^{\,k} \quad for \qquad q = 0,1...n; \qquad k = n+1, n+2,...N \qquad \text{(I.2.2.8)}$$

Consequently, by appropriate choice of U and $|\psi_0' >$ in Eqs. (1) and (6), one can divide all the initial states into *N-n* groups such that each state of the group is absorbed (with a sufficiently high probability) into the same prescribed state. Such a performance can be interpreted as pattern classification if each eigenvector introduced to the QRN is associated with the corresponding patterns.

We will not go into more mathematical details here in order to focus attention upon formation of an artificial language instead. For that purpose, suppose that each run of the quantum device is repeated *l* times while $l \le n$ independent measurements are collected and fed back into QRN. Then, instead of mapping (2), one arrives at the following:

$$\frac{1}{\sqrt{l}}\{0...010...010...0\} \to \{00...010...0010...0\}, \qquad m \le l \qquad \text{(I.2.2.9)}$$

$$\uparrow_{i_1} \quad \uparrow_{i_l} \qquad\qquad \uparrow_{j_1} \quad \uparrow_{j_m}$$

This corresponds to evolution of *k* different patterns introduced to QRN simultaneously.

One can generalize Eq. (4) to the following

$$p_{i_1..i_l j_1...j_m} = \frac{\prod_{\alpha=1}^{m} |\sum_{k=1}^{n} U_{j_\alpha k} a_k + \sum_{\beta=1}^{l} U_{j_\alpha i_\beta}|^2}{|\sum_{k \ne i}^{n} a_k^{\,2} + \sum_{\beta=1}^{l} (a_{i_\beta} + 1)^2|^m}, \quad m \le l \qquad \text{(I.2.2.10)}$$

by considering how each of the recurrent states combined with the offset vector (5) evolves under the action of the unitary operator U . Eq. (10) defines the probability of transition from the set of inputs $i_1...i_l$ to the set of outputs $j_1...j_m$. If *m=l*, and the offset vector is expressed by Eq. (6), the transition probability matrix p_l can be presented in the form similar to p_1 in Eq.(7)

$$p_l = \begin{pmatrix} p_{11\ldots 1}{}^{11\ldots 1} \cdots p_{11\ldots 1}{}^{nn\ldots n} \cdots p_{11\ldots 1}{}^{N^l N^l \ldots N^l} \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ p_{nn\ldots n}{}^{11\ldots 1} \cdots p_{nn\ldots n}{}^{nn\ldots n} \cdots p_{nn\ldots n}{}^{N^l N^l \ldots N^l} \\ 0\ldots\ldots\ldots\ldots 010\ldots\ldots\ldots\ldots\ldots\ldots 0 \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ 0\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots 01 \end{pmatrix}, \qquad \text{(I.2.2.11)}$$

This means that the corresponding l-variate stochastic process has n^l transient states $T_q(q = 1,2,\ldots n^l)$ and $N^l - n^l$ absorbing states A_γ, and therefore,

$\begin{pmatrix} n \\ l \end{pmatrix}$combinations of different patterns (in the form of normalized sums of different eigenstates) are mapped onto $N_c \le \begin{pmatrix} n \\ l \end{pmatrix}$different classes. Hence, the total number of pattern combinations that can be classified by the QRN is

$$S = \sum_{l=1}^{n} \begin{pmatrix} n \\ l \end{pmatrix} = 2^n \qquad \text{(I.2.2.12)}$$

Now the performance of the QRN can be given the following interpretation. As soon as the unitary matrix U and the offset vector $|\psi' >$ are chosen (see Eqs. (1) and (6)), all the transition matrices $p_k(k = 1,2,..l)$ are uniquely defined (see Eqs. (4), (7), (10) and (11)). It should be noticed that these matrices do not have to be implemented: they exist in an abstract mathematical space being induced by the operator U and offset vector$|\psi' >$.

If the only one measurement is fed back (l=1), then the transition matrix (7) manipulates by basic patterns-eigenstates that can be identified with ``letters" of an alphabet: by mapping each eigenvector into a corresponding class, it assigns a certain meaning to the letter. If $l$ independent measurements are fed back, $(1 < l \le n)$then the transition matrix (11) assigns certain meanings to combinations of letters, i.e., to $l$-letter ``words". In order to understand the rules of these assignments, i.e., the ``grammar", let us turn to Eq. (10). As follows from there, in general

$$p_{i_1 \ldots i_l}{}^{j_1 \ldots j_m} \ne p_{i_1}{}^{j_1} \otimes \ldots \otimes p_{i_l}{}^{j_m} \qquad \text{(I.2.2.13)}$$

i.e., an l-variate stochastic process is not simply the product of l underlying one-dimensional stochastic processes, and the difference

$$\Delta_{i_1 \ldots i_l}{}^{j_1 \ldots j_m} = | p_{i_1 \ldots i_l}{}^{j_1 \ldots j_m} - p_{i_1}{}^{j_1} \otimes \ldots \otimes p_{i_l}{}^{j_m} | \qquad \text{(I.2.2.14)}$$

expresses the amount of ``novelty", or new information created by interaction between different patterns via quantum interference. Formally Eq. (14) resembles quantum entanglement that is also responsible for creation of new information; however, actually this entanglement is not quantum: it is a correlation between several classical stochastic processes generated by quantum interference. It should be recalled that classical neural nets where patterns are stored at dynamical attractors, do not have a grammar: any combination of patterns is meaningless unless their storage is specially arranged, and that would require actual implementation of an exponential number of new attractors (see Eq. (12)).

I.2.3. Dynamical Complexity.

In this section, we will discuss Shannon and algorithmic complexity of QRN for emerging grammars. Although the concept of complexity is well understood intuitively, its strict definition remains an enigma since there are many different aspects which can be associated with complexity (the number of interacting variables, the degree of instability, the degree of determinism, etc.). Here we will associate dynamical complexity with the degree of unpredictability of the underlying motion. Then the Shannon entropy becomes the most natural measure of dynamical complexity of QRNs:

$$\overline{H} = -\sum_{i=0}^{n-1} \pi_i \log_2 \pi_i \qquad \text{and} \qquad \overline{H}_{\max} \propto \log_2 n \qquad (I.2.3.1)$$

Let us assume now that the unitary matrix in Eq. (1.1) is composed of a direct product of n 2×2 unitary matrices:

$$U = U_1 \otimes U_2 \otimes ... \otimes U_n, \qquad N = 2^n = 2^{q/4} \qquad (I.2.3.2)$$

where the number of independent components in U_i

$$q = 4n \qquad (I.2.3.3)$$

Then the dynamical complexity of QRN becomes exponentially larger (see Eq. (2):

$$\overline{H}_{\max} \propto \log_2 n \propto n \qquad (I.2.3.4)$$

although the algorithmic complexity is still expressed by Eq. (3). Thus, QRNs based upon representation (1) generate "complexity" in an exponential rate, and therefore the underlying stochastic processes attain structure of fractals. Indeed, as shown in Shroeder, M, 1991, a continuous version of a Markov process exhibits self-similar structure down to infinitesimal scales of observation. Although the Markov processes generated by QRNs are finite-dimensional, their scales approaches zero exponentially fast when the number of the variables n (see Eq. (5) grows only linearly. This means that QRN generate "quantum fractals" which can be applied to image compression, animation, or for a finite-dimensional representation of Weierstrass-type functions which are continuous, but non-differentiable. In contradistinction to classical fractals, quantum fractals are more controllable since their probabilistic structure can be prescribed.

Now suppose that we are interested in generating a stochastic process with prescribed probability distribution. Then the algorithmic complexity becomes important: it will allow us to preserve only $q=4n$ (out of $N = 2^n$) independent characteristics of the distribution (although the stochastic process will be still N-dimensional, and its Shannon complexity will be of order of n).

The difference between the Shannon and the algorithmic complexities affects the design of the l measurements architecture in the following way. Indeed, the input-output relationships require the number of mapping (i.e., quantum circuits) which is polynomial in N, i.e., exponential in n. However, if the unitary matrix U has a direct-product representation (4) then, as follows from the identity:

$$(U_1 \otimes U_2)(a_1 \otimes a_2) = (U_1 a_1) \otimes (U_2 a_2) = Ua \qquad (I.1.2.3.5)$$

i.e., $a = a_1 \otimes a_2$ (I.2.3.6)

and therefore, not only the size of the unitary matrix U and the state vector a, but also the number of mapping circuits for l-measurement architectures become polynomial in n as far as their actual implementation is concerned. In addition to that, in the case (4), n out of l measurements can be performed in parallel.

Eq. (4) is not the only representation of a unitary matrix which preserves its exponential size while utilizing only polynomial resources. Indeed, consider the following combination of matrix products:

$$U = (U_1^{(1)} \otimes ... \otimes U_n^{(1)})(U_1^{(2)} \otimes ... \otimes U_n^{(2)})...(U_1^{(m)} \otimes ... \otimes U_n^{(m)}) \qquad (I.2.3.7)$$

Here the number of independent components is:

$$q = 4mn \qquad (I.2.3.8)$$

while the dimensionality

$$N = 2^n = 2^{q/4m} \qquad (I.2.3.9)$$

In Eq. (9), N and q are associated with the Shannon and the algorithmic complexity, respectively.

Thus, each unitary operator having the structure (1) and supplied with an offset vector of the type (6) generates a new grammar. Since the structure (1) is preserved under matrix products, new operators of the type (9) represent new grammar. In particular, if the time period of each run of the QRN is increased in q times, then the effective unitary operator will be different from the original one and thereby a set of new languages can be generated by the same quantum ``hardware''. In addition to that, Eq. (7) opens up a possibility to build a high-dimensional operator U from low-dimensional components of the same structure.

It is worth mentioning that not every language of the possible set of languages is useful. Indeed, the performance of the QRN, and in particular, the assignments of pattern combinations to specific absorbing states is probabilistic. It is reasonable to require that for each selected patterns combination, the corresponding absorbing probability distribution over all possible states has a well-pronounced preference for a certain state; otherwise a word would lose its stable meaning. (It should be noticed that small overlapping of absorbing states is acceptable: it makes the language more colorful by incorporating double-meaning to some words.) As mentioned earlier, stability of the meaning of the basic patterns, i.e., letters, can be achieved by an appropriate choice of the unitary operator (1) and the offset vector based upon solutions of Eq. (8). However, as soon as U and $|\psi'_0>$ are fixed, there is no further control over stability of words' meaning since all the transition matrices p_i are already predetermined (see Eqs. (10) and (11)). In this situation, one can characterize the effectiveness of the language by the ratio ζ of the number W of useful words to the total number of words S

$$\zeta = \frac{W}{S}, \quad S \approx O(2^n) \qquad \text{(I.2.3.10)}$$

Hence, in order to maximize ζ , one has to identify such a solution to Eq. (2.8) which simultaneously stabilizes the meanings of all the letters as well as most of the words. Obviously, in general, this problem is hard.

4. Examples.

In order to demonstrate the existence of effective emerging grammars, consider the following example. Suppose that in Eqs. (2.1) and (2.6) are chosen as follows

$$U = \begin{pmatrix} \cos\varphi & \sin\varphi & 0 & 0 \\ -\sin\varphi & \cos\varphi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} a = (0,0,a_3,a_4) \qquad \text{(I.2.4.1)}$$

where φ, a_3, a_4 are real.

Then, applying Eq. (2.4) one finds the elements of the transition matrix p (see Eq. (1.7)):

$$p_1^{\ 1} = p_2^{\ 2} = \gamma_1 \cos^2\varphi, \qquad p_1^{\ 2} = p_2^{\ 1} = \gamma_1 \cos^2\varphi,$$

$$p_1^{\ 3} = p_2^{\ 3} = \gamma_1 a_3^{\ 2}, \qquad p_1^{\ 4} = p_2^{\ 4} = \gamma_1 a_4^{\ 2}$$

$$p_3^{\ 1} = p_3^{\ 2} = p_3^{\ 4} = p_4^{\ 1} = p_4^{\ 2} = p_4^{\ 3} = 0, \qquad p_3^{\ 3} = p_4^{\ 4} = 1 \qquad \text{(I.2.4.2)}$$

$$\gamma_1 = \frac{1}{a_3^{\ 2} + a_4^{\ 2} + 1}$$

As follows from Eq. (2), there are two transient states T_1 and T_2 , and two absorbing states, A_3 and A_4 .

Introducing four input patterns

$$|\psi_1 >= \{1000\}, \quad |\psi_2 >= \{0100\}, |\psi_3 >= \{0010\}, |\psi_4 >= \{0001\} \qquad \text{(I.2.4.4)}$$

as well as their images in the probabilistic space

$$|\pi_1 = \{1000\}, \quad \pi_2 = \{0100\}, \pi_3 = \{0010\}, |\pi_4 = \{0001\} \qquad \text{(I.2.4.5)}$$

first one can write down trivial mapping

$$|\psi_3 > \to \pi_3 \to A_3, \qquad f_3^{\ 3} = 1, \text{and } |\psi_4 > \to \pi_4 \to A_4, \qquad f_4^{\ 4} = 1, \qquad \text{(I.2.4.6)}$$

Other transitions

$$|\psi_1 > \to \pi_1 \to A_3, |\psi_1 > \to \pi_4 \to A_4, \qquad \text{(I.2.4.7)}$$

$$|\psi_2 > \to \pi_2 \to A_3, \text{and } |\psi_2 > \to \pi_2 \to A_4, \qquad \text{(I.2.4.8)}$$

are more complex, and they can be found from Eq. (2.8):

$$f_1^{\ 2} = p_1^{\ 1} f_1^{\ 3} + p_1^{\ 2} f_2^{\ 3} + p_1^{\ 3}, \qquad f_2^{\ 3} = p_2^{\ 1} f_1^{\ 3} + p_2^{\ 2} f_2^{\ 3} + p_2^{\ 3}, \qquad \text{(I.2.4.9)}$$

whence

$$f_1^{\ 3} = \frac{a_3^{\ 2}}{a_3^{\ 2} + a_4^{\ 2}}, \ f_2^{\ 3} = \frac{a_4^{\ 2}}{a_3^{\ 2} + a_4^{\ 2}}, \qquad \text{(I.2.4.10)}$$

Similarly one finds

$$f_1^{\ 4} = f_1^{\ 3}, \qquad f_2^{\ 4} = f_2^{\ 3} \qquad \text{(I.2.4.11)}$$

Thus, if

$$a_3 \cong a_4 \qquad \text{(I.2.4.12)}$$

the patterns $|\psi_1>$ and $|\psi_2>$ do not have any meaning; with the same probability they can be absorbed by the states A_3 or A_4 . However, if

$$a_3 >> a_4 \text{ or } a_3 << a_4 \tag{I.2.4.13}$$

the same patterns are absorbed by only one state A_3, A_4 and that assigns certain meaning to each of them.
For mapping combinations of patterns (4), one has to repeat twice each measurement before feeding it back. Now the input pattern's combinations will be the following:

$$|\psi_{12}>=|\psi_{21}>=\frac{1}{\sqrt{2}}\{1100\}, \qquad |\psi_{13}>=|\psi_{31}>=\frac{1}{\sqrt{2}}\{1010\} \tag{I.2.4.14}$$

but their image in the probabilistic space will be di€erent from (4)

$$\pi_{12}=\pi_1\otimes\pi_2 \qquad \pi_{13}=\pi_1\otimes\pi_3 \tag{I.2.4.15}$$

Instead of listing all the 64 elements of the matrix p_2 (see Eqs. (3.11) and (3.12)), we will concentrate upon those that will be used in our analysis. First of all

$$\begin{aligned}
&p_{ii}^{\alpha\beta}=0\,if && \alpha,\beta\neq i \quad i=3,4\\
&p_{ii}^{\alpha\beta}=1 && \text{otherwise.}\\
&p_{ij}^{\alpha\beta}=0\,if && \alpha,\beta\neq i \quad i=3,4\\
&p_{ij}^{\alpha\beta}=1 && \text{otherwise.}
\end{aligned} \tag{I.2.4.16}$$

This means that there are four absorbing states: A_{33}, A_{34}, A_{43} and A_{44} ; the rest 12 states (T_{12}, T_{13}, etc.) are transient. Here we will be interested only in the evolution of the pattern's combination $|\psi_{12}>$ (see Eq. (14)) since it is the only one which entangles the patterns $|\psi_1>$ and $|\psi_2>$ (see Eq. (4)). (Other combinations: $|\psi_{13}>|\psi_{23}>$ etc. are not entangled, and therefore, their evolution can be predicted from the previous analysis as a direct products $|\psi_{13}>\otimes|\psi_{13}>, |\psi_{13}>\otimes|\psi_{13}>$, i.e., it does not have any novelty element.)
Thus, one obtains

$$\begin{aligned}
&p_{12}^{11}=\gamma_2(\cos\varphi+\sin\varphi)^4, \quad p_{12}^{22}=\gamma_2(\cos\varphi-\sin\varphi)^4,\\
&p_{12}^{12}=\gamma_2(\cos^2\varphi-\sin^2\varphi)^2=p_{12}^{21} \qquad p_{12}^{13}=\gamma_2 a_3^2(\cos\varphi+\sin\varphi)^2\\
&p_{12}^{14}=\gamma_2 a_4^2(\cos\varphi+\sin\varphi)^2 \qquad p_{12}^{23}=\gamma_2 a_3^2(\cos\varphi-\sin\varphi)^2\\
&p_{12}^{24}=\gamma_2 a_4^2(\cos\varphi-\sin\varphi)^2 \qquad p_{12}^{34}=p_{12}^{43}=\gamma_2 a_3^2 a_4^2\\
&p_{12}^{33}=\gamma_2 a_3^4 \quad p_{12}^{44}=\gamma_2 a_4^4
\end{aligned} \tag{I.2.4.17}$$

where

$$\gamma_2=\frac{1}{(a_3^2+a_4^2+2)^2} \tag{I.2.4.18}$$

As follows from the last four equations in (17), there are direct transitions from the pattern combination $|\psi_{12}>$ to the absorbing states. However, in addition to that, there exist many indirect transitions to the same states, for instance, $T_{12}\to T_{13}\to T_{33}$, $T_{12}\to T_{14}\to T_{44}$, and these transitions include the entanglement effect that has maxima at $\varphi=\pm\frac{1}{\sqrt{2}}$.

As a result, the pattern combination $|\psi_{12}>$ acquires a new meaning since it cannot be reduced to the direct product of the patterns $|\psi_1>$ and $|\psi_2>$.
The performance of this simple QRN becomes more sophisticated if the elements of the unitary matrix U and the component of the offset vector a in Eq. (3.1) are complex numbers. Utilizing the properties (3.7), one can represent a unitary operator U in a compressed form gaining exponential dimensionality of U with linear resources.

I.2.5. Summary.

Thus, it has been demonstrated that QRN is capable of creating emerging grammars by assigning different meanings to different combinations of letters. The paradigm is based upon quantum interference of patterns which entangles the corresponding Markov processes, and thereby, creates a new meaning depending upon how different patterns interact. The capacity of the language, i.e., the total number of words in it is exponential in *n* where *n* is dimensionality of the basic unitary operator. However, if this operator is presented as a direct product, then the number of words can be made double-exponential in the dimensionality.

Chapter I.3
Quantum resonance for
Simulation NP-complete Problems.

Abstract.

The mathematical formalism of quantum resonance combined with tensor product decomposability of unitary evolutions is mapped onto a class of NP-complete combinatorial problems. It has been demonstrated that nature has polynomial resources for solving NP-complete problems and that may help to develop a new strategy for artificial intelligence, as well as to re-evaluate the role of natural selection in biological evolution.

I.3.1. Introduction.

The previous chapters were focused on combining the power of quantum physics and the dynamical complexity of neural nets. The main challenge of the approach was in reconciliation of linear reversible quantum evolution and nonlinear irreversible dynamics of neural nets. The attention was focused on those aspects of QRN that are related to modeling intelligence, namely: self-organization and formation of grammar. In this chapter an attempt is made to simulate another property related to intelligence: combinatorial optimization that is the main obstacle to artificial intelligence. It is a well established fact that nature exploits combinatorial optimization for natural selection. It is also known that even ants collectively solve combinatorial problems (such as the shortest path to food in a labyrinth) more efficiently than man-made artificial devices. That is why combinatorial problems are not only the obstacle, but is the greatest challenge to artificial intelligence. In this Chapter, a new approach to simulation of NP-complete problems is introduced: combinatorial properties of tensor product decomposability of unitary evolution of many-particle quantum systems are mapped to solutions of NP-complete problems, while the reinforcement and selection of a desired solution is executed by quantum resonance.

I.3.2. Quantum Resonance.

Consider a quantum system characterized by a discrete spectrum of energy eigenstates subject to a small perturbing interaction, and let the perturbation be switched on at zero time. The Hamiltonian of the system can be presented as a sum of the time-independent and oscillating components

$$\square H = H_0 + \varepsilon_0 H_1 \int_{\omega} \xi(\omega) \sin \omega t d\omega \qquad (I.3.2.1)$$

where H_0 and H_1 are constant Hermitian matrices, ω is the frequency of perturbations, and $\xi(\omega)$ is the spectral density.

The probability of the transition from state *k* to *q* in the first approximation is proportional to the product,

D. R. Bates, 1961: $$P_{kq} \propto | \varphi^{*}{}_{k} H_1 \varphi_q |^2 \frac{[\sin \frac{1}{2}(a_{qk} - \omega)t]^2}{(a_{qk} - \omega)} \qquad (I.3.2.2)$$

Here φ_i are the eigenstates of H_0:

$$H_0 \varphi_j = E_j \varphi_j, \qquad j = 1,2,..n. \qquad (I.3.2.3)$$

where E_i are the energy eigenvalues,

$$\hbar a_{kq} = E_k - E_{q,} \quad k, q = 1,2,..n \tag{I.3.2.4}$$

and □ $\hbar$ is the Planck constant.

The resonance, i.e., a time-proportional growth of the transition probability □ occurs when □ $\omega = a_{qk}$:

$$□ P_{kq} \propto |\varphi^*{}_k H_1 \varphi_q| \xi^2(\omega)t \tag{I.3.2.5}$$

I.3.3. Combinatorial problems.

Combinatorial problems are among the hardest in the theory of computations. They include a special class of so called NP-complete problems which are considered to be intractable by most theoretical computer scientists. A typical representative of this class is a famous traveling-salesman problem (TSP) of determining the shortest closed tour that connects a given set of *n* points on the plane. As for any of NP-complete problem, here the algorithm for solution is very simple: enumerate all the tours, compute their lengths, and select the shortest one. However, the number of tours is proportional to *n!* and that leads to exponential growth of computational time as a function of the dimensionality *n* of the problem, and therefore, to computational intractability.

It should be noticed that, in contradistinction to continuous optimization problems where the knowledge about the length of a trajectory is transferred to the neighboring trajectories through the gradient, here the gradient does not exist, and there is no alternative to a simple enumeration of tours.

The class of NP-complete problems has a very interesting property: if any single problem (including its worse case) can be solved in polynomial time, then every NP-complete problem can be solved in polynomial time as well. But despite that, there is no progress so far in removing a curse of combinatorial explosion: it turns out that if one manages to achieve a polynomial time of computation, then the space or energy grow exponentially, i.e., the effect of combinatorial explosion stubbornly reappears. That is why the intractability of NP-complete problems is being observed as a fundamental principle of theory of computations which plays the same role as the second law of thermodynamics in physics.

At the same time, one has to recognize that the theory of computational complexity is an attribute of a digital approach to computations, which means that the monster of NP-completeness is a creature of the Turing machine. As an alternative, one can turn to an analog device which replaces digital computations by physical simulations. Indeed, assume that one found such a physical phenomenon whose mathematical description is equivalent to that of a particular NP-complete problem. Then, incorporating this phenomenon into an appropriate analog device one can simulate the corresponding NP-complete problem. In this connection it is interesting to note that, at first sight, NP-complete problems are fundamentally different from natural phenomena: they look like man-made puzzles and their formal mathematical framework is mapped into decision problems with yes/no solutions. However, one should recall that physical laws can also be stated in a "man-made" form: The least time (Fermat), the least action (in modifications of Hamilton, Lagrange, or Jacobi), and the least constraints (Gauss).

In this Chapter we will describe how to map a combinatorial decision problem into the physical phenomenon of quantum resonance on a conceptual level, and propose a possible circuit implementation.

Let us turn to the property (5) that can be mapped into several computational problems, and, for the purpose of illustration, choose the following one: given *m* different items to be distributed over *n* places; the cost of a β^{th} item put in a γ^{th} place is $\lambda_\beta^{(\gamma)}$; in general, the costs (the number of which is *nm*) can be positive or negative (but not zero), and there are no restrictions to how many different items can be put at the same place. Find yes/no answer to the following question: is there at least one total cost whose absolute value falls into an arbitrarily given interval.

This problem is typical for optimal design. Since the cost of a particular distribution is expressed by the sum

$$□ E_j = \sum_{\beta=1}^{n} \lambda^{\gamma_\beta}{}_\beta, \qquad j = 1,2...N = m^n \tag{I.3.3.1}$$

classically one has to compute all the m^n sums (1) in order to find is there at least one E_q such that

$$a_1 \leq |E_q| \leq a_2; \qquad a_2 > a_1, \tag{I.3.3.2}$$

where a_1 and a_2 are arbitrarily prescribed positive numbers.

Since costs $\lambda_\beta{}^{\gamma_\beta}$ can be positive or negative, the absolute value in Eq. (2) represents a global constraint, and therefore our problem belongs to the class of so called constraint satisfaction problems that are the hardest among other optimization problems. The constraint (2) prevents one from decomposing the solution into smaller-size sub-problems. It can be shown that this problem is mapped into the partition problem, Garrey,M.,1979 , and therefore, it is NP-complete.

Now we will demonstrate how this problem can be solved by the quantum device described above in one computational step.

First, let us represent the unitary matrix U$_0$ corresponding to the time-independent Hamiltonian

$$\square U_0 = e^{iH_0 t} \qquad \text{(I.3.3.3)}$$

as a tensor product of *n* diagonal unitary matrices of the size $m \times m$

$$U_0 = U_1 \otimes U_2 \otimes ... \otimes U_n, \qquad \text{(I.3.3.4)}$$

This corresponds to the direct sum decomposition of *H$_0$*

$$H_0 = H^{(1)}{}_0 \otimes 1_m \otimes ... \otimes 1_m + 1_m \otimes H^{(2)}{}_0 \otimes ... \otimes 1_m + ...$$
$$+ 1_m \otimes ... \otimes 1_m \otimes H_0{}^{(n)} \qquad \text{(I.3.3.5)}$$

where 1_m is an $m \times m$ unit matrix.

Here

$$H_0{}^{(r)} = \begin{bmatrix} \lambda_1{}^{(r)} 0...0 \\ .... \\ 0.......0\lambda^{(r)}{}_m \end{bmatrix} \qquad U_r = \begin{bmatrix} e^{i\lambda^{(r)}{}_1{}^r}\ 0...0 \\ .... \\ 0........0e^{i\lambda_n{}^{(r)}} \end{bmatrix} \qquad \text{(I.3.3.6)}$$

Then the unitary matrix *U$_0$* in (4) will be also diagonal and

$$H_0 = \begin{bmatrix} E_1 0...0 \\ .... \\ 0.....0E_N \end{bmatrix}, N = m^n \qquad \text{(I.3.3.7)}$$

while E$_j$ is expressed by Eq. (2.1).

Hence, if one select $\lambda_\beta{}^{(\gamma)}$ in (3) as the costs of an β^{th} item put in a γ^{th} place, then the eigenstates E_j of the Hamiltonian H_0 represent costs of all $N=n^n$ possible distributions (3).

Without loss of generality, one can assume that $m=2$ since computation of 2^n different sums has also an exponential computational complexity. At the same time, this assumption will simplify the implementation of the simulation device.

Now we have to choose the perturbation of the Hamiltonian, (see Eq. (2.1)). For that purpose assume that initially the quantum device is in a certain base state *k*, whose energy E_k does not belong to the interval (2), i.e.,

$$|E_k| < a_1, \qquad or \qquad |E_k| > a_2, \qquad \text{(I.3.3.8)}$$

and select $\xi(\omega)$ as follows: □

$$\xi(\omega) = \begin{Bmatrix} \xi_0 = Const & if \dfrac{|E_k - a_2|}{\hbar} \leq \omega < \dfrac{|E_k - a_1|}{\hbar} \\ 0 & otherwise \end{Bmatrix} \qquad \text{(I.3.3.9)}$$

The only constraint imposed upon the Hamiltonian H_1 is that all its out-of-diagonal components are non-zeros. Indeed, in this case each element of the matrix (2.5) will have the form:

$$|\varphi_k{}^* H_1 \varphi_q|^2 = |\lambda_k \lambda_q H_1{}^{(kq)}|^2 \neq 0 \qquad if \qquad k \neq q. \qquad \text{(I.3.3.10)}$$

and no possible resonance transitions will be missed.

Here, for the sake of concreteness, the initial state E_k was selected such that:

$$|E_k - a_1| > |E_k - a_2| \qquad \text{(I.3.3.11)}$$

Turning to Eq. (2.1) and taking into account Eqs. (2.4) and (2), one concludes that Hamiltonian of the system is decomposable only at *t=0*. For *t>0*, the quantum system becomes correlated due to the perturbations.

Suppose that the given interval a_1,a_2 contains at least one total cost $|E_q|$ from the set (1), i.e., $|E_q|$ satisfies the inequality (2). Then, according to Eqs. (2.5) and (9), the resonance transition from the initial state E_k to the state E_q (or other states satisfying (2)) will occur with the probability one. Indeed, in the presence of a resonance, the probability for non-resonance transitions is vanishingly small if $\varepsilon_0 << 1$ (see Eq. (2.1)).

However, if the given interval a_1,a_2 does not contain any costs $|E_q|$ from the set (1), then according to Eqs. (2.5) and (9), there will be no resonance transitions at all, and therefore, with the probability one the quantum device will stay in the initial state.

A mapping of the combinatorial problems onto tensor decomposability of the Schrödinger equation is illustrated in Fig. 4

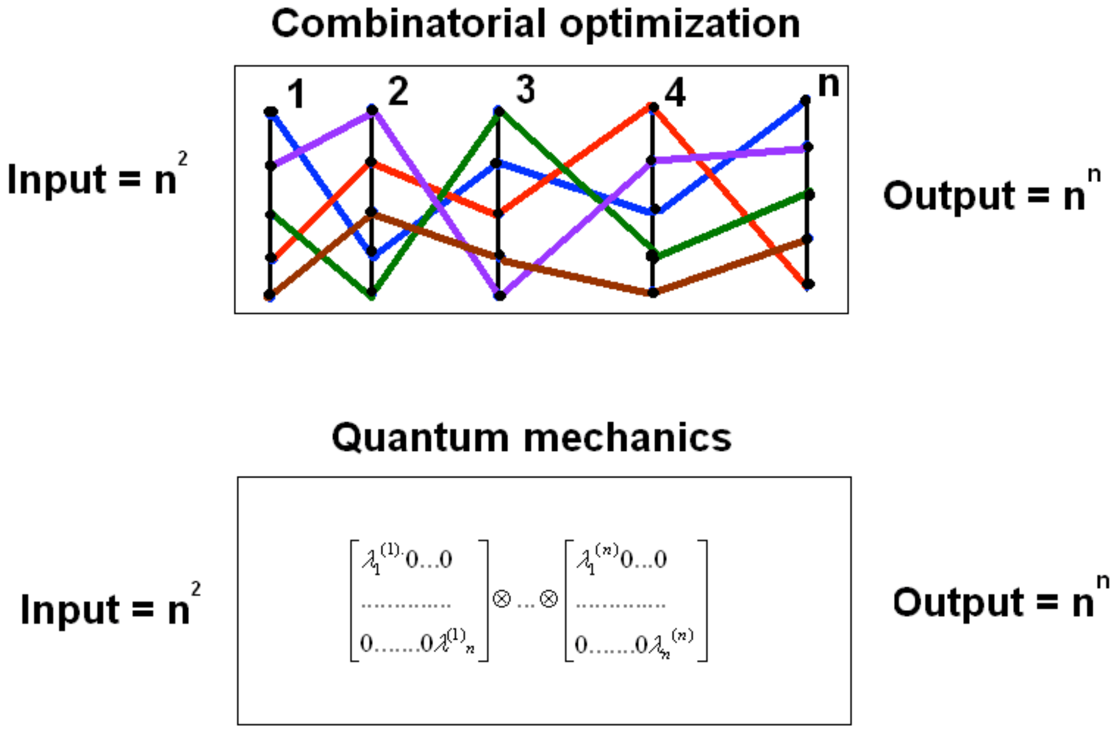
Combinatorial optimization
1
2
3
4
n
Input = n²
Output = nⁿ
Quantum mechanics
Input = n²
$\begin{bmatrix} \lambda_1^{(1)} 0 \ldots 0 \\ \ldots\ldots\ldots \\ 0 \ldots\ldots 0 \lambda^{(1)}_n \end{bmatrix} \otimes \ldots \otimes \begin{bmatrix} \lambda_1^{(n)} 0 \ldots 0 \\ \ldots\ldots\ldots \\ 0 \ldots\ldots 0 \lambda_n^{(n)} \end{bmatrix}$
Output = nⁿ

Figure 4. Mapping combinatorial optimization to quantum mechanics.

I.3.4. Circuit-based representation.

In this section we will discuss a transformation of the Hamiltonian (2.1) into the form that is suitable for a circuit-based implementation of the proposed algorithm.

Let us represent H_1 as a direct product of *n* identical 2×2 matrices:

$$H_1 = w \otimes w \otimes \ldots \otimes w \qquad \text{(I.3.4.1)}$$

where

$$w = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \qquad \text{(I.3.4.2)}$$

One should notice that the direct product (1) is different from the direct product of independent subspaces given by a direct sum (see Eq. (3.5)) so that the components (2) do not have a clear physical meaning. In other words, Eq. (1) expresses only a formal mathematical representation of the matrix H_1. Since the matrix *w* is normal, i.e., w*w=ww*, the following representations are available:

$$w = v\gamma v^{-1}, \qquad v^{-1} = v^T, \qquad \gamma = \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix}, \; v = \begin{bmatrix} \sqrt{2}/2 & -\sqrt{2}/2 \\ \sqrt{2}/2 & \sqrt{2}/2 \end{bmatrix},$$
$$w \otimes w \otimes \ldots \otimes w = ((v\gamma v^{-1}) \otimes (v\gamma v^{-1}) \otimes \ldots \otimes (v\gamma v^{-1}) =$$
$$(v \otimes v \otimes \ldots \otimes v) \bullet (\gamma \otimes \gamma \otimes \ldots \otimes \gamma) \bullet (v^{-1} \otimes v^{-1} \otimes \ldots \otimes v^{-1}) \tag{I.3.4.3}$$

where v and γ are the unitary and diagonal matrices, respectively.
Now the total Hamiltonian (1) can be rewritten in the following form:

$$H = H_0^{(1)} \otimes 1_m \otimes \ldots \otimes 1_m + 1_m \otimes H_0^{(2)} \otimes \ldots \otimes 1_m \otimes 1_m + \ldots$$
$$+ 1_m \otimes 1_m \otimes \ldots \otimes 1_m \otimes H_0^{(n)} + (w \otimes w \otimes \ldots \otimes w)\varepsilon_0 \int_\omega \xi(\omega) \sin \omega t d\omega \tag{I.3.4.4}$$

As follows from Eq. (4), at *t = 0* the system is decomposable into independent subspaces with the Hamiltonians $H_0^{(j)}$ since

$$H = H_0 \;\; at \qquad t = 0. \tag{I.3.4.5}$$

However, for *t>0* the system is entangled due to the action of the perturbations H_1.

One of the effective ways to build a physical system with the Hamiltonian (4) is via the corresponding unitary operator. However, because of time-dependent component of this Hamiltonian, the sought unitary matrix is not necessarily an exponent of the Hamiltonian. In order to circumvent this obstacle, we can approximate the time-dependent Hamiltonian

$$H = H_0 + \varepsilon_0 H_1 \int_\varpi \xi(\omega) \sin \omega t d\omega \tag{I.3.4.6}$$

with a sequence of piecewise constant Hamiltonians:

$$H_k = H_0 + \varepsilon_0 H_1 \int_\omega \xi(\omega) \sin \omega \Delta t d\omega, \qquad k = 0,1,2 \ldots p, \tag{I.3.4.7}$$

Then the solution to the corresponding Schrödinger equation can be presented as a linear superposition:

$$C = \prod_{k=0}^{p} \exp[H_0 + \varepsilon_0 H_1 \int_\omega \xi(\omega) \sin \omega \Delta t d\omega] \Delta_k t' \tag{I.3.4.8}$$

$$t_k{}' = -\frac{i}{\hbar} t \tag{I.3.4.9}$$

Selecting Δt_k sufficiently small

$$\Delta t_k << \omega^{-1} \tag{I.3.4.10}$$

one can rewrite Eq. (8) as

$$C = \prod_{k=0}^{p} \exp(H_0 \Delta t_k') \exp\{[\varepsilon_0 H_1 \int_\omega \xi(\omega) \sin \omega \Delta t_k d\omega] \Delta t_k'\} + o(\Delta t^2) \tag{I.3.4.11}$$

Here the last term (that is supposed to be small) compensates non-commutability of the matrices ||*H₀*|| and ||*H₁*||.

Substituting Eqs. (4) into Eq. (11) one finds

$$C = \prod_{k=0}^{p} \prod_{q=1}^{n} \exp(H_0{}^{(q)} \Delta t_k') \exp\{[(v \otimes \ldots \otimes v) \bullet (\gamma \otimes \ldots \otimes \gamma) \bullet (v^{-1} \otimes \ldots \otimes v^{-1}) \int_\omega \xi(\omega) \sin \omega \Delta t_k' d\omega] \varepsilon_0 \Delta t_k\} =$$
$$= \prod_{k=0}^{p} \prod_{q=1}^{n} \exp(H_0{}^{(q)} \Delta t_k')(v \otimes \ldots \otimes v) \exp\{[(\gamma \otimes \ldots \otimes \gamma) \int_\omega \xi(\omega) \sin \omega \Delta t_k' d\omega] \varepsilon_0 \Delta t_k'\}(v^{-1} \otimes \ldots \otimes v^{-1}) \tag{I.3.4.12}$$

where $$H_0^{(q)} = \begin{bmatrix}\lambda_{1q} & 0\\ 0 & \lambda_{2q}\end{bmatrix} \qquad \text{(I.3.4.13)}$$

Due to tensor-product-based representation (12) of the original Hamiltonian (2.1), the proposed algorithm can be implemented by a circuit of a polynomial complexity.

I.3.5. Summary.

Thus, it has been demonstrated how a "man-made" problems of exponential computational complexity which is hard to handle by algorithmic methods are solved by exploiting a strongly pronounced physical phenomena: quantum resonance.

The main advantage of the presented approach is in exponential speedup of solutions to NP-complete combinatorial problems. Three fundamental physical phenomena contribute to it: quantum resonance, entanglement, and tensor-product decomposability of the underlying unitary matrix.

Quantum resonance allows one to represent all the possible solutions to the problem as a set of competing dynamical processes: energy exchanges between pairs of quantum eigenstates. The mathematical formalism of quantum resonance provides storage for these processes: the transition matrix □P_{kq} (see Eq. (2.2)) where each process is labeled through the corresponding transition probability. Quantum entanglement implement the global constraint (2) and

tensor-product decomposability is a fundamental property of the Schrodinger equation for multi-particle systems. Due to its effect, the number of stored solutions, i.e., the number of transitions matrix P_{kq} is exponentially larger than the number of the input parameters (see Eq. (3.1)) and that is what directly contributes into exponential speedup and capacity.

In order to make these three physical phenomena work together, one has to choose the Hamiltonian of the quantum system such that the optimal solution is the winner in the competition with other solutions, i.e., that its transition probability is the largest. This is achieved by selecting the oscillating part of the Hamiltonian in the form of (3.9).

It should be emphasized that the solution of one NP-complete problem opens up a way to solve every NP-complete problem in polynomial time.

Possible implementation of the presented algorithm by a circuit of a polynomial complexity is demonstrated. However, one of the greatest challenges of building of quantum computer is controlling or removing quantum decoherence. This usually means isolating the system from its environment as the slightest interaction with the external world would cause the system to decohere. This effect is irreversible, as it is non-unitary, and is usually something that should be highly controlled, if not avoided.

References.

Zak, M., (1970), *Uniqueness and Stability of the Solution of The Small Perturbation Problem of a Flexible Filament with a Free End*, Journal of Applied Mathematics and Mechanics, Vol. 34, No. 6, Moscow, pp. 1048-1052, Transl. Engl. pp. 988-992.

Zak, M., "Terminal Model of Newtonian Dynamics," Int. J. of Theoretical Physics, No.32, 159-190, 1992

Zak, M., Quantum decision-maker, Information Sciences , 2000, 128, pp. 199-215

. Zak,M., 2003, Quantum algorithms in Hilbert Database. Int. J. of Theor. Phys.42, 2061-2068

Zak, M., Fijany,A., 2005, Quantum resonance for simulating combinatorial problems, Physics letters A, 345, 167-172.

Zak, M., 1994, *Postinstability models in Dynam*ics, Int. J. of Theor. Phys.vol. 33, No. 11, 2215-2280.

Zak , M., Quantum evolution as a nonlinear Markov process, Foundations of Physics Letters,
vol. 15. No. 3, 2002

Zak, M. 2005a, From Reversible Thermodynamics to Life. Chaos, Solitons &Fractals,
1019-1033,

Zak, M., Electronic Journal of Theoretical Physics **4**, No. 16(II) (2007) 11–96